The Applicaton of Knowledge-Based Techniques to Fault Diagnosis of 16 QAM Digital Microwave Radio Equipment.

by

K.E.Brown

Thesis submitted for the degree of Doctor of Philosophy to the Faculty of Science, University of Edinburgh. 1988



### Declaration of Originality

This thesis, composed entirely by myself, reports work conducted in the Department of Electrical Engineering at the University of Edinburgh and at Hewlett Packard QTD, South Queensferry exclusively by myself, with the exception of the section mentioned below.

The result presented in Section 6.2 is reproduced by kind permission of Dr T.M.Crawford.

signed

K.E.Brown

date: 20th April 1988

#### Acknowledgements

The author would like to express his gratitude, for their supervision, help and patience, to Professor P.M. Grant, Dr C.F.N. Cowan and Dr T.M. Crawford.

Thanks are also due to Mr G. Dobbie, Mr N. Carder and Mr D. Grieve of Hewlett Packard QTD. The finacial support of the Science and Engineering Research Council and Hewlett Packard QTD is greatly appreciated.

The help with the typing of Fiona, Gillian, Paula and Pat is gratefully acknowledged.

Finally, thanks are due to my parents for their encouragement, support and patience during my many years of education.

Contents

P	a	g	e	1	V	u	M	b	e	r

	m ! L ]		
	Title	page	(i)
	Abstra	act	(ii)
	Declar	ration of Originality	(iii)
	Acknow	wledgements	(iv)
	Conter	nts	(v)
	Abbre	viations	(viii)
Chapter	<u>r 1</u>	Introduction	1
	1.0	Introduction	1
	1.1	Thesis Outline	5
Chapter	r 2	Digital Radio	8
	2.0	Introduction	8
	2.1	Typical radio	10
	2.2	16 QAM modulation	14
	2.3	Fault sources in digital radio	19
	2.4	16 QAM signal constellation features	24
Chapter 3		Digital Radio Model	36
	3.0	Introduction	36
	3.1	Model requirements	37
	3.2	HP 4948A non-intrusive analyser	39
	3.3	Elements of the radio model	43
	3.4	Constellation feature variation	52
Chapte:	r 4	Expert Systems	61
	4.0	Introduction	61
	4.1	Advantages of knowledge-based systems	61
	4.2	Expert system structures	66
	4.3	Languages and shells for expert systems	72
	4.4	Present systems and their impact	76

(v)

page number

Chapter	-5	Rule-Based System	81		
0.001	5 0	Introduction	01		
	5 1	Pule generation	00		
	5.1	Rule generation	82		
	5.2	Prolog shell	91		
	5.3	C based system	98		
	5.4	Performance of the C based system	106		
	5.5	Summary	113		
Chapter	6	Machine Learning System	115		
	6.0	Introduction	115		
	6 1	Structure of the machine learning system	116		
	6.2	Performance of the machine learning system on a	120		
	0.2	real radio	149		
	6.3	Tests of the machine learning system on the	132		
		digital radio model			
	6.4	Hybrid system	138		
	6.5	Performance of the hybrid system used to	142		
		diagnose faults in the radio model			
Chapter	- 7	Conclusions	116		
chapter	7 0	Summary	140		
	7.0	Summary	140		
	/.1	Fulcher work	151		
	References Bibliography				
	Appendix A				
	elationships between the introduced impairments				

and the geometric features of the 16 QAM signal constellation.

(vi)

page number

185

193

197

202

Appendix B 181 The subroutine for the HP4948A to simulate TWT distortions and the routine for calling the various elements of the model transmitter.

# Appendix C

The source code for the prolog shell.

### Appendix D

The knowledge-base used with the prolog shell to perform the fault diagnoses.

Appendix E The source code for the adaptive combiner section of the machine learning system.

#### Appendix F

A copy of the paper entitled "The Application of Knowledge-Based Systems For Fault Diagnosis in Digital Microwave Radio Equipment", which was presented at the IEEE International Conference on Communications in June 1987. (viii)

# Abbreviations

Artificial intelligence	AI
Amplitude to amplitude conversion	AM-AM
Amplitude to phase conversion	AM-PM
Amplitude shift keying	ASK
Automatic gain control	AGC
Bit error rate	BER
Carrier to noise ratio	CNR
Frequency shift keying	FSK
Inphase	I
Intelligent knowledge-based systems	IKBS
Intermediate frequency	IF
Intersymbol interference	ISI
Phase lock loop	PLL
Phase shift keying	PSK
Pseudo random binary sequence	PRBS
Quadrature	Q
Quadrature amplitude modulation	QAM
Quadrature partial response signalling	QPRS
Quaternary phase shift keying	QPSK
Radio frequency	RF
Symbol timing recovery	STR
Travelling wave tube	TWT

#### CHAPTER 1

#### INTRODUCTION

### 1.0 Introduction

A knowledge-based [1,2,3,4] or expert system is a system which can replace, improve or aid a human "expert" in a given problem field. Such systems can be expected to increase productivity, improve reliability and reduce costs in a wide range of applications (production, planning, fault diagnosis, system control and others). Despite the fact that research in this area is still in its infancy, and the systems have yet to produce many of the benefits promised, vast amounts of money and effort are being invested in expert systems research. The Alvey and Esprit programmes are two cases in point. The Alvey programme is a U.K. government backed information technology initiative while Esprit is a collaborative European technology development programme.

Although the theoretical basis of many different types of expert system have been developed and reported in the literature, very few <u>working</u> systems have been detailed and implemented. The importance of producing an actual working system, and of evaluating the applicability of existing techniques to a specific task, provided the motivation for the work presented in this thesis. Once working expert systems have been developed for specific problems and they are shown to really free human experts to address themselves to other technical problems, then the knowledge gained in building these systems can be applied to other tasks to create a greater range of expert systems.

One of the subdivisions of expert systems consists of rulebased [5,6] systems which contain knowledge on how to solve a particular problem (fault diagnosis in this case). Such knowledge is embedded within the system as a set of rules. These rules are searched by a form of control to find the solution to the problem. The control can simply examine each rule in the order they are written or it can involve some other search technique, which looks for specific properties in a rule, before using the rule.

Another subdivision of knowledge-based systems is the machine learning system [7,8,9]; this kind of system obtains its information of the problem domain from a training sequence. Within the system there can be a variety of different algorithms which extract knowledge from the training sequence. This enables these systems to solve the problem during operation, when the algorithms are used in working mode instead of in training mode.

The area of application chosen to utilize the knowledge-based methods was the fault diagnosis of 16 state quadrature amplitude modulation (QAM) [10] digital microwave radio relay equipment. This task was chosen because it seemed a soluble problem and its successful completion would produce a system which would be of considerable assistance in the manufacture and maintenance of 16 QAM radios.

Fault diagnosis of 16 QAM digital radio equipment is required for two separate scenarios: first, for the initial tuning and setting up of the radio in the factory so that it meets its design specification; and secondly, for correcting faults which may occur in

the radio during normal operation. An expert system could be used in both of these applications to aid an engineer in his task of impairment minimisation.

The information source from which the expert system was able to diagnose faults in the radio was the signal constellation [11]. This is a graphical representation of the received information whose characteristics change in the presence of certain faults. The signal constellation is also the information source which is used by a human expert to perform fault diagnosis.

An extensive search of the literature on expert systems failed to reveal any published work in this area. There are numerous publications [12,13,14] on the approaches and techniques used for designing expert systems, but little information on systems which are actually operational.

This study started, therefore, with virtually no available knowledge of how expert systems can actually be applied to technical problems. In those circumstances it can be argued that if solutions are found to specific practical tasks, then the knowledge gained from this will be of benefit. It is valuable because it enables effective methods for developing expert systems to be generated. Further the applicability of particular techniques to specific problem types can be judged. The resulting decrease in effort required to produce acceptable working knowledge-based systems for other purposes will encourage their development and use which will, in turn, mean many of the potential benefits of these systems will be reaped. The problem to be tackled was well defined with the information source that a human expert uses to perform the fault diagnosis known. An expert system was required to diagnose the faults occurring in a 16 QAM radio. The signal constellation was used to provide the input data to estimate the impairments present. In this study existing knowledge-based techniques were used in order to evaluate their applicability to the chosen problem, rather than developing new methods. This choice permitted an operational prototype system to be developed and evaluated within the timescale of the project. The alternative of developing new methods might have produced better techniques for implementing a system but it is unlikely that it would have given a useful indication of their suitability for practical application within the period of study.

Applying existing knowledge-based techniques to solve a specific defined problem in this way diverts from the main body of expert system research conducted by the artificial intelligence (AI) community. Knowledge-based systems have formed a major area of AI work since the early 1970's when the first expert system DENDRAL [15] was reported. This has led to much of the work involving the simulation of human intelligence and developing techniques which mimic this intelligence. However, taking an engineering standpoint, with a problem requiring a solution, and applying expert system methods to achieve a solution does not involve any appreciation of the "intelligence". Without the constraint of "intelligence" in the system, the application of the existing knowledge-based techniques to the problem of fault diagnosis of the 16 QAM digital microwave radio equipment can be tackled. The remainder of the thesis details the 16 QAM radio equipment, the effects of the faults, the knowledge-based

techniques developed to detect them and the evaluation of the expert systems' perfomance.

### 1.1 Thesis Outline

The description of the work starts in Chapter 2 with some basic information on digital radio. The most significant benefits derived from digital transmission are presented to explain why digital radio is gaining in popularity [16,17]. The basic elements of a digital radio are initially described. 16 QAM modulation is then discussed and its benefits over other modulation techniques in terms of efficiency and ease of implementation are explained. The possible fault sources, apart from channel impairments, and their effects on the radio signal are identified and discussed. In the final section of Chapter 2 the 16 QAM signal constellation is presented and its characterisation is explained using a set of geometric features. The geometric feature set forms the parameters which are used to provide the information to establish the fault levels. Without the required parameters, it would prove impossible accurately to assess the faults present. This chapter forms an introduction to 16 QAM digital radio, its faults and their effects on the signal constellation. Only with an adequate appreciation of the working radio can the development of the expert systems be fully understood.

Since it was not possible to gain access to a 16 QAM radio throughout the period of the project, a radio model was developed which could be used to evaluate the expert systems' performance. The radio model is described in Chapter 3. Various requirements of the

model for this work, and the specific type of model chosen, are identified along with the hardware on which it was implemented. The structure of the radio model, and the techniques that were used to simulate the elements of this structure, are explained. Results obtained from the radio model in the form of the relationships between the introduced distortions and the signal constellation's geometric feature set are also recorded. These relationships form the information which is used to create rules for the development of a rule-based system. This chapter discusses, therefore, the choice of radio model and its implementation, and then presents the results used in the generation and evaluation of the knowledge-based systems.

In Chapter 4 a brief review of expert systems, their merits, structures, languages and impact is presented. Their areas of application and the potential benefits to be derived from their use are discussed. The basic structures of various expert system types are then explained together with the method by which they search the available data. Programming languages and shells available for use with expert systems are presented before discussing details of some of the systems already developed and their areas of application.

The rule-based and machine learning systems formed the basis for the expert system techniques used during the project. The rulebased approach was chosen because it is the most widely tried method and it can, in some respect, model the way a human expert performs the fault diagnosis. The machine learning method was chosen to permit a more automated system to be produced, which would not require such a great input of human expertise to devise the rules.

The first approach to constructing a knowledge-based system in this study took a rule-based approach; this is described in Chapter 5. Two methods of implementing a rule-based system were attempted and both are detailed. General purpose tools called "shells" exist and can be used to develop expert systems. In this case the development of a rule-based system using a Prolog shell was initially undertaken. Another rule-based system was also written in the C programming language based on similar rules. The generation of these rules from the fault/feature relationships obtained in Chapter 3 is explained. Results obtained while evaluating the diagnostic performance of the C based system are discussed.

The second method of creating an expert system, using a machine learning technique, is detailed in Chapter 6. An account is provided of the structure of the machine learning system, the algorithms within the structure and how the training of the system is performed. The machine learning system was tested on a 16 QAM digital radio; the faults introduced and the diagnostic performance of the system are presented. The machine learning system was also tested on the radio model and these results are recorded and discussed. A comparison of the rule-based and machine learning systems is given. This has led to the development of a hybrid system which is also described. The performance achieved by the hybrid system on tests with the radio model is recorded and reviewed.

In the final chapter the conclusions drawn from the work presented in the thesis are discussed. A number of suggestions for additional work are also offered. These suggestions follow from the results and conclusions of the research reported in this thesis.

#### CHAPTER 2

#### DIGITAL RADIO

### 2.0 Introduction

Digital modulation techniques are increasingly being used for radio transmission [18,19]. These methods are used in satellite/earth stations and in terrestrial line of sight radios. Using signal regeneration techniques and error detecting and correcting codes [10,20,21], digital transmission allows the carrier to noise ratio (CNR) to be reduced without increasing the error rates. With an 8 state trellis coding system [16] a further 4 dB reduction in CNR can be achieved for no increase in the error rate.

The introduction of spectrally efficient modulation techniques [22,23] has lead to digital transmission becoming more widely applied. These techniques, such as quadrature amplitude modulation (QAM), offer the advantages of digital transmission including: signal regeneration; error detecting and correcting coding; and signal encryption. Figure 2.0.1 shows how the spectral efficiency of multilevel QAM systems increases with the number of signal states, and also the resultant increase in CNR required.

In this Chapter a brief outline of the basic elements of a digital radio is presented. In Section 2.1 the elements of a typical radio transmitter and receiver are described. The purpose of each block within the radio's structure and its interconnection with the rest of the radio is also explained. Section 2.2 gives an account of 16 QAM modulation and its implementation, discusses the strengths and



Figure 2.0.1 Spectral efficiency of multi-level QAM systems and resultant increase in carrier to noise ratio (CNR).

weaknesses it exhibits over other modulation techniques, and shows how tradeoffs in efficiency and ease of implementation have led to 16 QAM being the most widely applied modulation technique for digital radios. The possible fault sources, apart from channel impairments, and their effects on the radio are identified and discussed in Section 2.3. Finally, in Section 2.4 the signal constellation features which are used to describe the geometry of the signal constellation are presented. The calculation of the features and the unique properties they describe about the constellation, are also explained.

#### 2.1 Typical Radio

The block structure of a typical digital radio transmitter and receiver is shown in Figure 2.1.1. In the transmitter section of the radio, the intermediate frequency (IF) modulated signal is first amplified and then bandpass filtered to ensure that the signal remains within the regulatory bandlimits. The filtered signal is next predistorted to compensate for the non-linear distortions introduced by the power amplifier.

The non-linearities present in the power amplifier depend upon the type of amplifier being used. Travelling wave tube (TWT) [24] amplifiers are the most common; solid state amplifiers, which are more efficient [25], are under development and are gradually being introduced. The characteristics of a TWT power amplifier are discussed in Section 3.3.







Receiver

The automatic gain control (AGC) amplifier is included to ensure that the signal fed into the radio frequency (RF) power amplifier is at the correct power level. This level is set so that the transmitted signal is at a high enough power for efficient operation, whilst not driving the amplifier too far into its nonlinear regions.

The non-linear regions of the power amplifier produce amplitude to amplitude conversion (AM-AM) effects and amplitude to phase conversion (AM-PM) effects. These effects cause signals to have an amplitude and phase distortion dependent upon the signal magnitude, as shown in Figures 2.1.2 and 2.1.3 respectively. As many of the preferred modulation techniques require linear transmitter and receiver operation to enable the received signal states to be easily distinguished/detected, non-linear effects impose severe constraints on the system operation.

After levelling by the AGC amplifier the IF signal is up-converted to RF before the high power amplification stage. The AGC loop has a bandwidth considerably less than the information bandwidth to ensure no distortion is introduced by the levelling process. This is then output via a waveguide filter to the antenna for transmission. The waveguide filter also prevents the transmission from extending outside the specified regulatory frequency ranges by reducing the level of the unwanted sidebands generated in the up-conversion process.

The receiver also has a waveguide filter connected to its antenna, this discriminates against any unwanted signals which are



Figure 2.1.2 Typical input/output amplitude characteristics of a travelling wave tube amplifier.



Figure 2.1.3 Typical input/output phase characteristics of a travelling wave tube amplifier.

present. The waveguide filter facilitates multiplexing, dropping out the different channels in the frequency spectrum. The signal is then down-converted from RF to IF and is filtered to produce the required IF signal, which is adaptively equalised [26,27] before passing to the demodulator. The adaptive equaliser helps to remove some distortions introduced from the channel, in particular, group delay. The AGC amplifiers at RF and IF ensure the signal remains at the correct power level for the accompanying processing circuits.

A typical 11 GHz 16 QAM microwave radio will have a transmission rate of 140 Mbits/sec and a 40 MHz bandwidth, with a spectral efficiency of 3.5-3.7 bits/sec/Hz. The repeater station spacing will be between 20 and 50 km with a transmitting power of between 25 to 40 dBm.

Some radios may not include all of these elements, or may contain additional components, e.g. additional filtering stages. However, Figure 2.1.1 shows the outline of a radio which provides the functions of an operational system.

# 2.2 16 QAM Modulation

There are many types of digital modulation techniques currently in use. These techniques include amplitude shift keying (ASK), frequency shift keying (FSK), phase shift keying (PSK) and a derivative of ASK and PSK, quadrature amplitude modulation (QAM). There are also various other hybrid techniques apart from QAM, which are variants of the methods mentioned above, such as quaternary phase shift keying (QPSK) and quadrature partial response signalling

(QPRS). In addition to these methods, there are some more specialised techniques [28,29] which improve system performance under specific interference conditions. All of these alternatives have tradeoffs with regard to bandwidth efficiency, noise immunity and technical realisability. Feher [30] details the currently available spectrally efficient digital modem techniques.

The number of signal states can vary, depending upon the method being used, from the simple binary case of 2 states up to experimental systems of 1024 states. The systems with 1024 states are normally 1024 QAM systems; they are not in widespread use however, because of the highly linear systems that are needed to implement them. These systems are in the experimental stage and should come into limited use in the next few years.

Currently the most common type of signalling used for high data rate systems is 16 QAM. A 16 QAM modulated signal has a theoretical Nyquist rate [31,32] of 4 bits/s/hz, although practical systems achieve only 3.5-3.7 bits/s/hz. (Bits/s/hz is the number of binary bits which can be transmitted each second in 1 hz of bandwidth). This is however, a large improvement on the spectral efficiency of a purely binary signal with a maximum theoretical Nyquist rate of 1 bits/s/hz.

The theoretical Nyquist rate for any modulation technique is never achieved because of imperfect filtering. 64 QAM systems have a theoretical Nyquist rate of 6 bits/s/hz, but, because of the severe filtering requirements of these systems those currently in use achieve only 4.5-5.0 bits/s/hz. A 6 dB increase of carrier to noise

ratio (CNR) is needed for these 64 QAM systems over the 16 QAM systems, to achieve the same BER. Figure 2.0.1 shows the spectral efficiencies of these systems and the increase in CNR. This increase in CNR is required because the signal states in a 64 QAM constellation have smaller amplitude and phase separation than those in a 16 QAM constellation and, therefore, are less noise tolerant.

Figure 2.2.1 shows the two dimensional amplitude phase diagram of the 16 different signal states of a 16 QAM signal constellation [18]. This diagram illustrates how the 16 different signal states are separated by amplitude and phase. There are 3 distinct amplitudes ( $\sqrt{2}$ ,  $\sqrt{10}$ ,  $3\sqrt{2}$ ) and 12 distinct signal phase values (18°, 45°, 72°, 108°, 135°, 162°, 198°, 225°, 252°, 288°, 315°, 342°) in a 16 QAM signal constellation. The signal constellation is described in greater detail in Section 2.4.

The modulation of a 16 QAM signal is performed by first splitting the binary data into two parallel bit streams at half the data rate. These two data streams are then converted into a four level signal. These two parallel four level signals are used to phase modulate two orthogonal carriers: an inphase (I) and a quadrature (Q) carrier. The two modulated carriers are then summed to produce a 16 QAM signal. Figure 2.2.2 shows a block diagram of such a 16 QAM modulator.

The 16 QAM constellation provides a good noise immunity [33,34,35] while still remaining relatively easy to modulate and demodulate. There are other 16 state signals, such as 1-5-10, V.29 and optimum [16], which provide greater immunity to certain types of



Figure 2.2.1 Amplitude-phase diagram of a 16 QAM constellation.



16 QAM Modulator

Figure 2.2.2 16 QAM modulator block diagram.



16 QAM Demodulator

Figure 2.2.3 16 QAM demodulator block diagram.

interference, such as multiplicative noise or phase jitter. However, these require more complex modulators and demodulators and hence are not widely used in digital microwave radio systems.

The structure of a 16 QAM demodulator is shown in Figure 2.2.3. The first stage of demodulation involves recovering the carrier [36,37], normally using a phase lock loop (PLL). This recovered carrier is then used to generate orthogonal (quadrature) reference signals. These two orthogonal signals are used to demodulate the inphase and quadrature components of the signal. From the four level inphase and quadrature components, the symbol timing recovery (STR) [36] is performed. This STR information is used in the four to two level converter to establish the correct sampling instant for both the I and Q channels. These two binary signals are then recombined to give the binary output signal.

QAM modulation techniques are now well tried and it has proved to be economic to produce and operate the equipment to implement them, whilst still providing a spectrally efficient method of transmission. Of these modulation techniques, 16 QAM is currently the most common, although 64 QAM and 256 QAM [38] are gradually being introduced for some terrestrial applications despite their requirement for greater equipment linearity and receiver CNR. The other signalling techniques [39] with superior interference rejection properties have not gained significant popularity in engineering applications primarily because of the requirement for increased sophistication in the modulators/demodulators.

# 2.3 Fault Sources in Digital Radio

There are many possible sources of faults in digital microwave radio relay equipment. Faults can occur in the channel which include: interference in the form of noise or a specific tone, multipath effects [40,41,42] or blockage of the direct path by a solid object such as a building. These channel faults are of considerable importance to the operation of a radio system; they can vary between those which are catastrophic in nature and so prevent any data transmission, or are relatively minor and increase the BER. However, since they are not part of the actual radio equipment, and so are not faults in the transmitter or receiver, these channel impairments will not be discussed further here. This thesis concentrates on the detection and analysis of equipment malfunctions.

In the transmitter and receiver of a digital radio, Figure 2.1.1, faults can occur in the filtering sections, amplification stages and the quadrature phase splitting during modulation and demodulation. If the IF and RF filtering, of both the transmitter and the receiver, are maladjusted introducing a slope, ripple or notch into the passband, additional intersymbol interference (ISI) will be caused. Without a completely flat passband there will be distortion of the signal spectrum causing ISI and a resultant increase in the BER. These faults occur if either resonant circuits for IF filtering or cavities for waveguide filtering are incorrectly tuned.

If the amplification stages are incorrectly aligned, due to a fault in the setting of one of the AGCs or the malfunction of one of the amplifiers, then the signal fed into the following section of the radio will be at an inappropriate level. Any of the amplifiers, in

either the transmitter or the receiver, can be driven at an incorrect level causing their output signal to be at an unsuitable level which can cause the radio to fail completely or the BER to increase. However, the RF power amplifier in the transmitter, usually a travelling wave tube (TWT) amplifier, will have highly non-linear characteristics and so a small maladjustment of this section will create a large signal distortion. This maladjustment is termed either an amplifier overdrive or underdrive from the preferred level, depending upon the precise input signal levels.

The quadrature splitters, both in the transmitter and the receiver, do not always split the signal by exactly 90°, and so the two signals may not be truely orthogonal. If the deviation from 90° is sufficiently great then a non-orthogonal carrier fault will result.

In the transmitter modulator if either the I or the Q channel 2 to 4 level coder is incorrectly adjusted, one or more of the signal amplitudes (or spacings) will be incorrect. This can occur in either or both channels (I and Q) and will cause the signal at the demodulator output to possess incorrect amplitude and phase information. This fault is termed a gap spacing level error.

In the receiver demodulator the carrier recovery system may not lock onto the carrier at all; in that event, there will be no phase information available for the demodulation process. The carrier recovery system may lock onto the signal with a phase difference which will introduce a fixed phase error to the signal for decoding. If the oscillator producing the carrier is not stable and

has a varying frequency, this will appear as phase jitter on the signal.

The signal constellation is viewed in the demodulator once the received carrier has been removed from the signal to bring the I and Q components down to baseband. Errors can occur in the STR circuitry and the four to two level converters. However, these faults happen in the system after the signal constellation has been viewed. Since the signal constellation is used as the information source for the fault detection, these faults have to be diagnosed using other techniques.

The faults described cover many of the operational problems associated with digital microwave radio transmitters and receivers. Some radios do not include all of these components and some have additional filtering and equalisation sections; the fault sets available will be different therefore, for each design of radio.

Figure 2.3.1 shows a normal 16 QAM signal constellation. A radio with a 3 dB TWT amplifier overdrive generates a signal constellation as illustrated in Figure 2.3.2. This shows how the outermost states of the constellation are rotated clockwise and compressed in amplitude, while the innermost states are contra-rotated and expanded in comparison to the normal signal constellation. Figure 2.3.3 shows a signal constellation of a radio with a non-orthogonality of the I and Q carriers of 5 degrees. The signal constellation must now be described by a set of parameters which provides a basis for describing the effect of each introduced



Figure 2.3.1 A normal 16 QAM constellation.



Figure 2.3.2 A 16 QAM constellation of a radio with a 3dB travelling wave tube overdrive.



Figure 2.3.3 A 16 QAM constellation of a radio with 5 degrees carrier non-orthogonality.

impairment. The set of constellation features detailed in the next section have been developed to describe the 16 QAM constellation.

# 2.4 <u>16 QAM Signal Constellation Features</u>

The signal constellation of a 16 QAM digital radio is viewed by a constellation analyser [43] connected to the receiver demodulator as shown in Figure 2.4.1. The signal constellation is an amplitude-phase diagram of the position of the signal states and the spread of each of these states. In signal constellations the inphase component lies along the x axis direction and the quadrature component along the y axis direction. The distance from the centre of the constellation (the origin) to each signal state provides a measure of the amplitude of that signal state. The angle formed between the x axis and a line through the origin and the centre of the state is the phase of the signal. The 16 QAM signal has 3 distinct signal amplitudes and 12 discrete signal phase values to describe the 16 signal states as shown in Figure 2.4.2.

Human experts who perform fault diagnosis on digital microwave radio use the signal constellation as their information source for the diagnoses. To allow a knowledge-based system to exploit the information in the signal constellation, for digital radio fault diagnosis, a method is required that represents the information in the signal constellation. A geometric feature set was developed to describe the signal constellation using a set of geometric parameters calculated from a set of 16000 signal sample values. The use of approximately 1000 samples per signal state, provides a suitably averaged description of the signal constellation.









Figure 2.4.2 16 QAM constellation showing 3 distinct amplitudes and 12 distinct phases present.

The geometric features calculated comprise:

- (1) % Expansion of outer states
- (2) % Expansion of inner states
- (3) %I + %Q gap spacing error
- (4) %I %Q gap spacing error
- (5) Constellation rotation (degrees)
- (6) Differential rotation of inner to outer states (degrees)
- (7) Non-orthogonality of constellation (degrees)
- (8) Ratio of number of inner points to outer points in sample set
- (9) I pool deviation (sum of I squared)
- (10) Q pool deviation (sum of Q squared)
- (11) I.Q pool variance (sum of I\*Q)
- (12) Correlation coefficient

This geometric feature set primarily describes the location (in amplitude and phase) of the signal states. Faults other than those considered here (e.g. phase jitter) can effect the shape of the signal state. If these faults were to be diagnosed, further geometrical features would be required. The feature set is calculated by first forming two reference squares from those of the eight signal states which are neither among the four innermost or four outermost signal states. The position of each signal state is taken to be the mean position of the cluster of points forming that state. Figure 2.4.3 shows these two reference squares. These squares are used to help to determine the expected amplitudes and phases of the inner and the outer signal states of an undistorted constellation.



Figure 2.4.3 Construction of inner and outer reference squares.

The geometric features listed are calculated in the following manner. The first two features, the expansion of the outer states and the expansion of the inner states, are calculated by taking the average percentage increase in observed amplitude of the four outermost signal states (or of the four innermost states) compared to the expected amplitude, given by the corners of the reference square. If the signal states are compressed rather than expanded, these features are negative. The equations for calculating the percentage expansion using the distances shown in Figure 2.4.4 are:

% outer expansion = 
$$\frac{r_{ave}^{-} r_{o}}{r_{o}} \times 100\%$$
 (2.4.1)

% inner expansion = 
$$\frac{ri_{ave} - ri}{ri} \times 100\%$$
 (2.4.2)

Where ro<sub>ave</sub> and ri<sub>ave</sub> are calculated by taking the average amplitude of all four of the outermost and innermost signal states.

Features 3 and 4 are calculated from the I and the Q gap spacing errors. The gap spacing errors are found by taking the percentage difference between twice the length of the side of the inner reference square, and the length of the side of the outer reference square in the I direction and in the Q direction. This gives the %I gap spacing error and the %Q gap spacing error. Using the dimensions shown in Figure 2.4.5 the %I gap error and the %Q gap error are found using:

%I gap spacing error = 
$$\frac{y_i - 3x_i}{y_i} \times 100\%$$
 (2.4.3)


Figure 2.4.4 Calculation of % expansion of the innermost and outermost signal states of a 16 QAM constellation.



Figure 2.4.5 Calculation of I and Q gap spacing error for a 16 QAM constellation.

$$\frac{\sqrt{q} - 3xq}{\sqrt{q}} \times 100\%$$
 (2.4.4)

Features 3 and 4 are subsequently calculated by taking the sum and difference of these, \$I + \$Q gap spacing error and \$I - \$Q gap spacing error.

Feature 5, the constellation rotation, is the average rotation of all of the 16 signal states in the received constellation with reference to a perfect constellation with its I component parallel to the x axis and its Q component parallel to the y axis. This is taken as the average of the sum of the phase angles of all 16 states minus 2880 degrees (the sum of the phases of the normal constellation). Taking the angles in Figure 2.4.6 this is calculated using:

constellation rotation = 
$$\frac{16}{(\Sigma \alpha n)} - 2880$$
 (2.4.5)

Feature 6, the differential rotation between the innermost and outermost signal states, is the mean of the phase difference between the innermost and the outermost signal states. This is calculated using the angles illustrated in Figure 2.4.7 by:

differential rotation = 
$$\sum_{n=1}^{4} \frac{\alpha in - \alpha on}{4}$$
 degrees (2.4.6)

Feature 7, the non-orthogonality of the signal constellation, is the mean of the difference between the angles of the corners of



Figure 2.4.6 Calculation of 16 QAM constellation rotation.



Figure 2.4.7 Calculation of inner to outer rotation for 16 QAM constellation.

both reference squares and 90 degrees. Figure 2.4.8 shows the angles taken to calculate this:

non-orthogonality of constellation =  

$$\frac{1}{8} [(180 - \delta i 1 - \delta i 3) + (\delta i 2 + \delta i 4 - 180) + (180 - \delta o 1 - \delta o 3) + (\delta o 2 + \delta o 4 - 180)] \text{ degrees}$$
(2.4.7)

The ratio of the number of points in the innermost signal states to the number of points in the outermost signal states is taken for the sample set. This ratio is the inner to outer ratio and is a measure of how evenly the points are distributed between the inner signal states and the outer signal states. The I and Q pool deviation gives an indication of the distribution of the clusters forming the signal states. For each point in each of the 16 states' clusters the square of the I component and the square of the Q component are summed to form the I and Q pool deviation

I pool deviation = 
$$\frac{16000}{\Sigma} \text{ In}^2$$
(2.4.8)

$$Q \text{ pool deviation} = \frac{16000}{\Sigma} Qn^2 \qquad (2.4.9)$$

Where In is the magnitude in the I direction of the nth point and Qn is the magnitude in the Q direction of the nth point.

The I.Q pool variance is the sum over all the points in the sample of the I direction magnitude multiplied by the Q direction magnitude.



Figure 2.4.8 Calculation of non-orthogonality of 16 QAM constellation.

I.Q pool variance = 
$$\sum_{n=1}^{16000}$$
 InQn (2.4.10)

The correlation coefficient is given by the I.Q pool variance divided by the square root of the I pool deviation multiplied by the Q pool deviation.

correlation coefficient =  $\frac{I.Q \text{ pool variance}}{\sqrt{((Ipool dev.)(Qpool dev.))}}$ (2.4.11)

This is a measure of the correlation of the noise in the signal in the I and in the Q directions.

These geometric features are mostly first order features. That is they are formed from linear combinations of the geometry of the signal constellation. There are four of the set, however, which are second order features: the I pool deviation, the Q pool deviation, the I.Q pool variance and the correlation coefficient. These second order features use quadratic and linear combinations of the signal geometry. Apart from the four features mentioned the remaining features are first order.

These simply calculated geometric features define the positions of each of the constellation states. They can be used as the information input to a knowledge-based diagnostic system for 16 QAM radio equipment analysis.

### CHAPTER 3

### DIGITAL RADIO MODEL

## 3.0 Introduction

The main focus of this thesis is on the diagnosis of faults in digital radios using knowledge-based systems. In the development of such systems it was necessary, of course, to have some means of accurately evaluating their performance, and for that a test vehicle was required. A model of a digital radio was seen as an appropriate vehicle. In the absence of existing models suitable for the purpose, a development was undertaken to design a model specifically for the evaluation of the diagnostic performance of knowledge-based systems.

In this chapter the various aspects of the radio model chosen, and the results obtained from it, are examined. Section 3.1 presents the various requirements of the radio model to permit the satisfactory evaluation of the knowledge-based systems' diagnoses, and the rationale behind the choice of the specific radio model type. The hardware used to implement the model (HP4948A In-Service Transmission Impairment Measuring Set [44]), the original intentions for its use, and the software available from Hewlett Packard to run on it are described in Section 3.2. Section 3.3 provides details of the specific structure of a digital radio which was simulated by the radio model. The techniques used to simulate the elements of the radio, apart from those available in the form of Hewlett Packard software for the HP4948A, are also explained. Finally, in Section 3.4 the relationships linking the distortions introduced into the radio model to the constellation geometric feature set are

catalogued. These relationships form the basis for creating the rules for a rule-base and those which are most significant, in the rule generation process are highlighted.

# 3.1 Model Requirements

The difficulty of access to an appropriate radio throughout the period of the project, combined with the possibility of the presence of unknown faults in the equipment, precluded the use of such a radio for evaluating the performance of the knowledge-based system. The presence of unknown faults could adversely affect the analysis of the distortions caused by deliberately introduced faults and could result in incorrect associations of distortions and fault conditions being formed. It was decided, therefore, that a digital radio model, would be used to permit the distortions caused by the various impairments in a digital radio to be examined.

If the impairments are introduced into a perfectly functioning model, all of the distortions will be attributable to the known faults and none to unknown problems. Each fault is introduced in measured amounts which allows the distortions to be examined in a quantified way. The use of a model, therefore, allowed faults to be introduced into a radio to obtain data to test the knowledge-based systems. Faults introduced into the model were diagnosed by the expert system and, by comparison with known types and levels of faults present, the accuracy of the diagnoses were subsequently determined. The distortions were characterised by the variation of the signal constellation geometric features, which were detailed in Section 2.4.

There were two available options to model the digital radio and its fault conditions: one makes use of an off-line software simulation and the other a signal processor running in real time. Using an off-line software simulation permits any specified accuracy of modelling for any component in the digital radio to be achieved. This accuracy in the modelling is obtained at the expense of computational time. The off-line simulation could be written in a high level language, making the coding and the finding of programming faults easier than if assembly language or machine code was used. The time needed to produce the code for the model, therefore, would be kept relatively short. Simulating each section of the radio would be relatively straightforward as there are already, within the Electrical Engineering Department at the University of Edinburgh, routines written to model specific elements of digital radios such as filtering stages. Interaction with the model, however, would be difficult; (by means of a constellation display), adjusting its various parameters during operation in the way one can when adjusting a real radio.

A signal processor running in real-time would allow the results to be produced in a continuous recognisable form (as a signal constellation), as well as recording the signals for further analysis. The signal processor would not necessarily be required to produce data at the same rate as real microwave radio. The effect of running it at a fraction of the normal rate would only reduce the rate of data collection compared to a real radio. The model could then be treated, therefore, as if it were a real radio, apart from the fact that data collection would be at 9.6 k bits/s rather than at 140 M bits/s. It would then be possible to view the constellation in

real-time and to monitor its variation with the introduction of specific faults. However, the amount of processing required to perform this type of modelling is large and the accuracy is less than that obtained from an off-line model. This reduction in accuracy is caused by the limited processing power available from a real-time processor.

The alternative of a real-time processor was chosen despite its potential accuracy being lower than that of an off-line model. It was found that the accuracy of the real-time processor was still high, as it used 16 bit arithmetic, and further it permitted interaction with the user in the form of a constellation display to indicate the effect of the introduced impairments. A commercial product, the HP4948A non-intrusive communications analyser, was chosen to implement this type of digital radio model as it was readily available. Section 3.2 details the HP4948A non-intrusive analyser.

# 3.2 HP4948A Non-Intrusive Analyser

The HP4948A non-intrusive analyser was originally designed for testing leased voice frequency data circuits. Figure 3.2.1 shows a HP4948A analyser unit. Conventional test methods require that the circuit be removed from service for test. The HP4948A can be connected to a suitable point in a network and it will monitor live modem signals to perform the required tests. As well as being able to test in-service circuits, it can be used on out-of-service circuits with conventional test sets providing a modem like signal, and it may be remotely monitored as one of a network of test devices.



Figure 3.2.1 HP 4948A Non-intrusive analyser unit.

By logging the results over a long time period, changes in performance can indicate impending faults, permitting preventive maintenance to be accomplished.

The processing is all performed digitally and the input signal to the unit is sampled by a 12 bit analogue to digital converter. These digital samples are passed to a specially designed signal processor which executes, in real-time, the various algorithms required to simulate the modems and perform the required measurements. The overall structure of the hardware of the HP4848A is shown in Figure 3.2.2.

The processor execution unit has been optimised to perform digital convolution, since this function represents the largest processing requirement for real-time computation. The execution unit is designed around a 16-by-16 bit multiplier. This multiplier is fed by two segments of memory, memory A and memory B. An arrangement of this kind allows the accumulate and add function (X \* Y + P) to be performed efficiently, which is required for digital convolution. There are various other scratch pads and registers which are used for the sequencing, control and manipulation of the data. The controlling software for this unit was developed using a software development system which links the routines and libraries together before compiling the assembly language into executable code. The code is then downloaded from the computer running the software development system to the HP4948A unit. This permitted new assembly language code routines to be written and linked to the existing routines to perform additional tasks. Without this facility the software for the unit would be required to be interconnected at each



Figure 3.2.2 Structure of HP 4948A processor.

stage of writing the code, and thus software preparation would have been a much more time consuming process.

The HP4948A has software developed to permit it to be treated as a universal modem for transmitting or receiving signals typically encountered in the commonly used modems. The available modems include: the Bell 209, CCITT V.29 and 8 DPSK. The software for these modems provides the basic building blocks for simulating a digital radio transmitter and receiver. Routines available for the simulation provide modulators, demodulators, adaptive equalisers and IF filters. The HP4948A, in conjunction with the software development systems, thus provides a unit which can be used to simulate the basic blocks of a communications system.

# 3.3 Elements of the Radio Model

A digital radio model was constructed on the HP4948A nonintrusive analyser. This was done by writing subroutines to simulate the elements of a radio; these subroutines were written in AM2910 assembly language [45]. The elements of the digital radio which were modelled are shown in Figure 3.3.1. There is no power amplifier predistorter, as shown in Figure 2.1.1, nor are any of the channel impairments such as multipath fading and other interfering signals included in the model; these are not part of the radio equipment. The data is transmitted at 2400 bits/s as opposed to a typical 16 QAM digital microwave radio transmission rate of 140 M bits/s. This model does, however, allow the introduction of the same impairments as would occur in a real radio.









The faults introduced into the radio model were: errors in the in-phase(I) or quadrature (Q) spacing levels, non-orthogonality of the I and Q carriers and travelling wave tube (TWT) power amplifier distortions. The I (or Q) spacing errors were introduced by directly altering the values of the levels in the subroutine performing the coding. The non-orthogonality of the I and Q carriers is simulated by a routine which adds an offset to the signal; this is dependent upon which of the 16 signal states is being transmitted. The offset is the difference between a perfect constellation and a constellation formed by non-orthogonal carriers. This operation is performed before any additional filtering so that the signal undergoes all the perturbations which it would have been subject to had the impairment been caused by non-orthogonal carriers.

These two types of fault were straightforward to introduce into the model, because both use look-up tables to alter the signal levels. These look-up tables were constructed by determining the appropriate value for each entry in the table to produce the required distortion. The distortion due to the TWT power amplifier, however, proved more complicated to simulate.

The distortions caused by a TWT power amplifier are in general non-linear and cause both AM-AM and AM-PM conversions. Figure 2.1.2 shows the effect of the AM-AM conversion in a TWT amplifier, and the AM-PM conversion is shown in Figure 2.1.3. Saleh [46] gives simple two parameter formulae for describing these TWT distortions; these formulae apply to both an amplitude-phase and to a quadrature non-linear model of a TWT amplifier.

The amplitude-phase model is given by:-

$$x(t) = r(t)\cos(\omega_{t} + \psi(t))$$
 (3.3.1)

where

x(t) is the input signal

r(t) is the modulating envelope

w is the carrier frequency

 $\psi(t)$  is the modulating phase

 $y(t) = A(r(t)) \cos(\omega_0 t + \psi(t) + \phi(r(t)))$  (3.3.2)

where

y(t) is the output signal

A(r(t)) the output signal amplitude is an odd function of r, with the leading term representing AM-AM conversion  $\phi(r(t))$  the output signal phase due to TWT non-linearities is an even function of r, with quadratic leading term representing AM-PM conversion.

 $A(r) = \alpha_a \frac{r}{(1 + \beta_a r^2)}$  (3.3.3)

 $\phi(r) = \alpha_{\phi} \frac{r^2}{(1 + \beta_{\phi}r^2)}$  (3.3.4)

where scale factors  $\alpha$  and  $\beta$  for A and  $\phi$  are determined by particular amplifier characteristics.

The alternative quadrature model where the output signal y(t) is expressed in rectangular co-ordinates is shown in Figure 3.3.2. where

$$p(t) = P(r(t)) \cos(\omega_{o}t + \psi(t))$$
(3.3.5)  

$$q(t) = -Q(r(t)) \sin(\omega_{o}t + \psi(t))$$
(3.3.6)

This expression for the TWT distortions is obtained from the amplitude phase model with

$$P(r) = A(r)\cos(\phi(r))$$
$$Q(r) = A(r)\sin(\phi(r))$$

and

 $P(r) = \alpha_{p} \frac{r}{(1 + \beta_{p}r^{2})}$  (3.3.7)

$$Q(r) = \alpha_q \frac{r^3}{(1 + \beta_q r^2)^3}$$
 (3.3.8)

On first inspection the quadrature model seems the most convenient form to simulate the TWT power amplifier distortions on the radio model. The I and the Q components of the signal are available directly in the HP4948A equipment. This means that conversion to polar co-ordinates would not be required.

The distorted levels of the I and Q components of the signal could then be computed directly:-

Inew	=	P(r)I	(3.3.9)
Qnew	=	Q(r)Q	(3.3.10)



 $-q(t)=Q(r(t)sin(w_t+))$ 

Figure 3.3.2 Quadrature non-linear model of a travelling wave tube power amplifier.

When using the HP4948A, however, it was found that P(r) and Q(r) could not be calculated because the amount of processing required more time than that available between the data samples. The calculation of r is performed by taking the square root of  $r^2$  and this requires the use of a time consuming iterative sub-routine. The division required to calculate P and Q also uses an iterative, and therefore computationally expensive, technique.

The simulation of the distortion, therefore, was achieved through an alternative strategy in which the quadrature model was rejected and the amplitude-phase model was re-examined. Saleh and Salz [47] gave the amplitude and phase relationships as:-

$$A(r) = \frac{2r}{1+r^2}$$
(3.3.11)

$$\phi(\mathbf{r}) = 60^{\circ} \frac{\mathbf{r}^2}{1 + \mathbf{r}^2}$$
(3.3.12)

These two equations are updates of equations (3.3.3) and (3.3.4) with the actual parameters for a specific TWT amplifier included. These are:  $\alpha_a = 2$ ,  $\beta_a = 1$ ,  $\alpha_{\phi} = 60^{\circ}$  and  $\beta_{\phi} = 1$ . This requires that the I and Q values be converted to polar co-ordinates. Performing this conversion and the calculation of A(r) and  $\phi(r)$  would require more processing than simply using the quadrature model. To avoid this a combination of the two forms of representing the distortions was developed. The magnitude of the signal, r, was calculated from the I and Q values, and from this A(r) and  $\phi(r)$  were evaluated.  $\phi(r)$ , the phase change, was then applied to the I and Q levels to correspond to the required phase shift. A(r) can be expressed as a multiplier of the I and Q values to give their magnitudes after the TWT distortions. This is done by:-

given from the I and Q values.

The range of the numbers that the HP4948A can handle is limited to -1 and +1. A(r) and k can both exceed this range so they require to be scaled along with the numerator and denominator of  $\phi(r)$  and A(r). These scaled versions are:-

$$A(r) = 0.5 \frac{r}{0.5(1 + r^2)}$$
(3.3.14)

$$\phi(\mathbf{r}) = 60^{\circ} \frac{0.5(\mathbf{r})}{0.5(1 + \mathbf{r}^2)}$$
(3.3.15)

$$k = \frac{A(r)}{r} = 0.5 + 0.5 \frac{0.5(1 - r^2)}{0.5(1 + r^2)}$$
(3.3.16)

This scaling of the signal is removed, after the TWT distortion has been introduced, by using a routine which alters the signal to give a specified root mean square (RMS) level of the signal. These steps, which are performed to model the TWT distortions, are shown in the form of a flow chart in Figure 3.3.3.

This technique still did not meet the speed requirements for implementing the required distortions. Furthermore, it was not



Figure 3.3.3 Steps in modelling travelling wave tube distortions.



possible to operate the HP4948A with lower data rates which would have provided the extra time needed to complete the processing.

Updating the phase of the I and Q values and altering their magnitude to distort the signal was an efficient technique. It allowed the use of the amplitude-phase model while not requiring conversions between rectangular and polar co-ordinates. The timeconsuming elements of the process were calculating A(r),  $\phi(r)$  and the square root of  $r^2$ . To overcome this inefficiency, a look-up table approach was used. By constructing the look-up table as a function of  $r^2$  no square roots had to be calculated, and k and  $\phi(r)$  values could be simply obtained. Splitting  $r^2$  onto 128 steps from 0 to 1 gives a look-up table with its address offset being the 7 most significant bits (MSB) of  $r^2$ . This has a step size of less than 1% in the look-up table, minimising the errors in  $\phi(r)$  and k. To account for the backoff on the TWT,  $r^2$  is altered before it is used as an address offset in the look-up table.

This technique, which is shown in the form of a flow chart in Figure 3.3.4, successfully models the TWT distortion in the radio simulation. The three fault types were simulated in a radio model as shown in Figure 3.3.1. The changes in the various constellation features with these introduced faults are detailed in Section 3.4.

Appendix B contains a listing of the subroutine which simulates the TWT distortion and the program which calls up the required subroutines to simulate a radio transmitter using the 4948A.



Figure 3.3.4 Technique for simulating travelling wave tube distortions.

TWT power amplifier overdrive and underdrive, spacing errors in the coder and non-orthogonal I and Q carriers were the faults introduced into the radio model. These faults were introduced singly to examine how they altered the constellation features of an otherwise good radio. The ranges of these distortions examined were:-

> TWT overdrive up to 10 dB overdrive in 2 dB steps TWT underdrive up to 8 dB underdrive in 2 dB steps Spacing errors in constellation -10% to +10% in 1% steps Non-orthogonality of carriers up to 5 degrees in 1 degree steps

A complete set of these variations for all the constellation features for each fault are given in Appendix A. Figures A1 to A12 show the variation of the geometric features for the different levels of TWT backoff. The "normal" level for the TWT amplifier was taken to be when the TWT amplifier was backed off by 12 dB. This figure was chosen because it represents a level with little constellation distortion, but with the TWT still being driven quite hard and, therefore, in its efficient operation region. A TWT amplifier backoff greater than 12 dB is an underdrive and one of less than 12 dB is an overdrive. Figures A13 to A24 show the variation of the geometric features with the varying degrees of non-orthogonality of the carriers. Figures A25 to A36 show the variation of the geometric features with the introduced spacing error of the constellation.

The geometric features which show a distinct relationship with the level of TWT amplifier backoff are the inner expansion, outer compression and the inner to outer differential rotation. These variations with TWT backoff are shown in Figures 3.4.1, 3.4.2

Note The TWT amplifier backoff is the reduction in the input signal from the level at which the output signal reaches saturation.



Figure 3.4.1 Variation of outer states compression with travelling wave tube backoff.







Figure 3.4.3 Variation of inner to outer rotation with travelling wave tube backoff.

and 3.4.3. These features vary over the whole range of the introduced TWT amplifier distortions. All of the other geometric features, apart from the correlation coefficient, remain constant over the full range of TWT backoff until the backoff is less than 6 dB (that is a TWT amplifier overdrive of more than 6 dB). This change, when the TWT amplifier overdrive is greater than 6 dB, arises from distortions causing the signal states to cross the normal decision boundaries introducing a change in the signal statistics. The correlation coefficient rises as TWT amplifier backoff is increased until a 6 dB underdrive is reached and, thereafter, remains constant.

The constellation features which show a discernible correlation with the spacing error in the modulator are the I gap spacing error and the Q gap spacing error. Figures 3.4.4 and 3.4.5 show the relationships between the introduced spacing error and the I and Q gap spacing errors. All of the other constellation features remain constant over the whole range of introduced spacing errors.

One of the constellation geometric features shows a clear relationship with the introduced non-orthogonality of the I and Q carriers. This feature is the non-orthogonality of the signal constellation; the variation of this feature with the introduced carrier non-orthogonality is shown in Figure 3.4.6. The other features all remain constant over the examined range with the exceptions of the I.Q. pool variance and the correlation coefficient. However, these features vary so little compared to their overall magnitude that they provide little information.



Figure 3.4.4 Relationship between the introduced gap spacing error and the I gap spacing error.



Figure 3.4.5 Relationship between the introduced gap spacing error and



Introduced non-orthogonality of the I and Q carriers (degrees)

Figure 3.4.6 Variation of the non-orthogonality of the constellation with the introduced non-orthogonality of the I and Q carriers.

The features which show a distinct relationship with the faults present provide a basis for establishing the level of each fault present. Those constellation features, which remain constant for most fault conditions, can be used to reinforce the conclusion that the radio is correctly adjusted since they would be expected to vary with the introduction of other unexamined faults (such faults would prevent the radio being correctly adjusted). The generation of rules to determine the condition of the radio and the levels of the faults present is detailed later in Chapter 5.

## CHAPTER 4

## EXPERT SYSTEMS

### 4.0 Introduction

In Chapters 2 and 3 accounts have been given of digital radio and a model of a digital radio. These accounts are a preliminary to the discussion of the use of knowledge-based systems for the purpose of diagnosis of faults in digital radios. This chapter provides a brief review of knowledge-based or (expert) systems which recently have become a popular area for research [1,2,3,4,48,49].

First in Section 4.1, the advantages and the most productive areas of use for these systems are discussed. The limitations and problems associated with the development and operation of these types of systems are also detailed. Section 4.2 outlines the basic structures of knowledge-based systems and the search techniques they employ. In Section 4.3, a brief overview of the programming languages used for expert systems, and of how these languages and expert system shells can be of use for system construction, is given. Some of the systems which have already been developed are presented in Section 4.4. The discussion includes their areas of expertise, operating structures, impact, advantages and disadvantages.

# 4.1 Advantages of Knowledge-Based Systems

Knowledge-based systems are optimised to solve, or help to solve, a problem which would normally be referred to a human "expert" in the field. They can approach, and in certain situations even

surpass, human performance levels. These systems capture human judgement or expertise for use in areas where there is a scarcity of skilled engineers. A system to aid or replace an expert is a valuable commodity which would release him or her to perform additional tasks. Areas of application of knowledge-based systems outwith engineering include: medicine, geology, chemistry, law, sonar data interpretation and speech understanding amongst others [50,51,52,53,54,55]. The majority of these systems are diagnostic type systems, but some have been developed for aiding, planning and advising functions.

Much of the work into expert systems has been performed by researchers from the Artificial Intelligence (AI) community, among whom there is substantial discussion as to whether these systems are "intelligent" or are merely an extension of conventional programming. This debate is centred around the large quantity of data which is programmed into the knowledge-bases of these systems and used to solve the problems. The point of contention is whether this constitutes a similar basis for the thought processes in humans and artificially intelligent systems, or if it is simply an elaborate form of programming. This debate has been going on for many years and will doubtless continue, but is relevant only to the definition of expert systems within AI, and not to their actual implementation and application. It is far more important to determine what can actually be achieved by such knowledge-based systems.

An expert usually has to find the solution to a problem from incomplete data using "expert knowledge" in the problem domain. A human expert will have a knowledge both of the problem area and of

the different ways that exist for solving the problem; this is termed "meta-knowledge" or knowledge of the knowledge. Once this metaknowledge (of the problem solving techniques) has been programmed in an appropriate form into a knowledge-base, it is possible to produce a useful expert system. These methods of problem solving are often referred to as heuristics. The knowledge engineer must understand both the system's structure and the problem area, so that he can correctly interrogate the expert and elicit the required information.

A knowledge-based system may be able to perform some or all of the following:

- (1) Solve the problem
- (2) Explain its solution
- (3) Learn from tackling the problem
- (4) Update the knowledge-base
- (5) Determine whether or not a problem lies within its area of expertise

Any given knowledge-based system [14,56], however, will only be able to perform a subset of these tasks; this should not present any difficulty because, depending upon the application area, only a certain subset will be needed for each specific problem. The structure used to implement the expert system will be dependent upon the tasks to be undertaken. A review of structure types and their relative merits is given in Section 4.2. The most important requirement of an expert system is that it be able to solve the given problem. The requirements for all the other capabilities of an expert system are application dependent.

An explanation of the solution which has been arrived at may be required for several reasons. If the expert system is being used as an aid for a human expert, the human expert may require to see the reasoning behind the system's solution before being prepared to use it. If the human expert is not satisfied with the system's reasoning, it may be possible for him to use this information to help update the system's knowledge-base (if it is possible to alter the knowledge-base). An explanation of the solution is also vital when the expert system is being used as a teaching aid; under those circumstances it is necessary to make explicit the reasoning behind each decision if the user is to achieve an adequate understanding of the process.

The facility of learning [8,9,57] from performing the task allows a knowledge-base to be automatically improved upon if a particular solution to the problem proves to be incorrect or incomplete. This feature is potentially one of the most useful, but it is also one of the most difficult, to implement. Learning of this kind would imply that the initial generation of the knowledge-base would be less critical since it would be continually updated during operation. However, if the initial performance of the system has to be near perfect, then learning would not be of such great benefit.

If the system can determine whether a problem is within its area of expertise (or knowledge domain), then the system will not

attempt to find a solution when its expertise lies outwith the problem area.

There are many applications where carefully developed knowledge-based systems can be of great value. A variety of tasks which an "expert" is needed to perform have a narrow problem domain. Expert systems are particularly well suited to this type of problem since it is possible to produce a complete but still manageable knowledge-base. In many situations an expert system's performance can surpass that of a human expert purely because it is reliable and immune to the boredom of the task. Used as teaching aids, furthermore, these systems can train their operators and bring them up to "expert" levels thus allowing less skilled personnel to undertake the work, and so release the experts for other tasks.

There is, therefore, great potential for those systems to increase productivity. To date, productivity has been limited by lack of sufficient suitable expertise, although the availability of systems which can perform experts' tasks would seem to allow unlimited possibilities for production and development in many fields.

There are, however, several problems. The main difficulty is associated with the generation of a knowledge-base [12]. The problem area has to be reasonably small so that whilst the knowledge-base remains manageably small it can nevertheless be comprehensive. If the knowledge-base is incomplete (that is, information for solving certain problems is missing), or if it is too large to implement as part of an economic system, the expert system will be of no use.
Once it has been determined that an expert system represents a viable solution to a particular problem area, generation of the knowledgebase may still prove difficult. The knowledge-engineer must extract information from an expert who may not be able (or willing) to pass on this information. The final task in the generation of the knowledge-base is to determine if it is complete or if more details are required from the expert.

It is evident that the areas of application of expert systems are limited to some degree. However, it is possible to build useful expert systems if the problem domain is suitable and if the knowledge engineer can gain the required information to generate the knowledgebase. These systems can be expected to increase productivity and performance, and to provide an important aid for established experts. In addition these working systems then form a starting point for producing more elaborate systems for solving more complicated problems.

## 4.2 Expert System Structures

Most expert systems [5] organise their knowledge on three levels: the data, the knowledge-base and the control. Figure 4.2.1 shows a general structure for an expert system. The control, or "inference engine", is generally kept as simple as possible to minimise the work required to produce comprehensive explanations of decisions and to make changes and improvements to the overall system easier to implement. A well designed system will make use of redundancy in the information available to help reach a solution. Redundancy involves making use of several paths leading to the same



Figure 4.2.1 Structure of a knowledge-based system.

answer. Moreover, these multiple paths mean that the confidence in the solution [13,58,59] can be reinforced. The inference engine can also interface the data and the knowledge-base to update the knowledge-base. It is possible for the knowledge-base and the control to be merged into one unit, but it is generally more convenient and common to keep them separate.

There are four main types of structure for the control: rulebased, structured, logic and mathematical relationship systems. Other types of structure exist but these are generally combinations or slight alterations of these four. The following considers each of the main types of structure:

(1) Rule-based Systems

Rule-based or production systems [6,60] consist of a set of rules and a rule interpreter which decides the order in which the rules are triggered or accessed. The rules of the system are applied to the available information (the data) to reach one or more conclusions. These conclusions can have a "confidence level" which provides an indication or measure of the reliability of the final conclusion. Valid conclusions for this type of system would include a rule reporting that the problem was unknown to the system. This would indicate that the system was being used to solve an inappropriate problem.

Production systems can be forward or backward [6] driven, that is, the reasoning can be data driven or goal driven. Data driven, or forward chaining, takes a set of conditions or data and

uses these to reach a goal or goals. Goal driven, or backward chaining, uses a conclusion or conclusions and establishes the conditions required for these conclusions. These required conditions are compared to the available data to see if the goals are valid. The forward or backward chaining processes can be performed in one stage or multiple stages to reach the final conclusions. If there are several stages, the goals from one stage are used as the input data for the next stage. This type of multi-stage process can be termed a "blackboard" system [61]. A blackboard system is one in which hypotheses are put forward by several stages of a system and these are then added to, or altered by, other stages of the system. It is termed a blackboard because it can be compared to several different experts each putting forward hypotheses on a blackboard, with other experts then altering or updating the current best hypotheses held on the blackboard.

Rule-based systems require a comprehensive set of rules, known as a rule-base. These rules need to cover the conditions which would be encountered for every different conclusion being considered. During the development of such a system the knowledge engineer, who generates the rule-base, and the expert, from whom he extracts the information, must both ensure that all the required combinations of data and conclusions are adequately covered. Once a comprehensive rule-base is constructed this type of system can be very useful, as explanations of decisions can be obtained by recording which rules were used in reaching each conclusion. If, however, it is not possible for the expert to express his knowledge in the form of explicit rules to cover all the possible outcomes, then rule-based systems do not provide a suitable structure.

### (2) Structured Systems [6,62]

These are systems which can be expressed as a structure of nodes and arcs as in graph theory. These systems group information in a "natural" way for use within an expert system. Semantic nets or associative nets consist of nodes with connecting links representing the various associations present. For example, the link between an eagle and a bird could be "a kind of". This could then be used to find out what type of animal an eagle is. A system of this type can in many cases be more efficient than using a large number of production rules. However, graph searches can have combinational explosion problems when there is a large quantity of information available.

Frame systems are constructed of frames which group information about a particular topic into slots. The frames are organised into trees to form hierarchies of subjects and their attributes. These trees can then be matched to conclusions and, depending upon the degree of matching, a confidence in the conclusion can be given. An example of a frame for a person is: name, age, sex and height. Then, as part of a tree, a subject with specified attributes can be found. There are, however, some situations where there is no convenient way of structuring the knowledge, and in which case structured systems cannot be used. The search mechanisms for structured systems, like production systems, can be goal or data driven.

(3) Logic Systems [6]

These systems have all their knowledge expressed in the form of logical statements and they use this information to deduce the conclusions. For example, if the system knows that "all birds have wings" and is then given the information that "an eagle is a bird" it can then deduce that "eagles have wings". These logically based systems can be used for a variety of applications but not in situations where it is not possible to form logical statements to express the knowledge available to the system. These theorems of logic are often implemented using logic programming languages such as Prolog [63]. Using these methods with a knowledge-base of logical theorems and a given set of input data, inferences to specific conclusions can be made.

# (4) Mathematical Relationship Systems

Some systems provide either an exact or an approximate mathematical relationship [64,65,66] between the input data and the goal. There are many different types of mathematical relationship which can be used as the basis of one of these systems, but the exact relationship between the data and the goals must be known. These conditions are only infrequently met; if they are not, then there has to be a way of training a suitable mathematical algorithm. The algorithm after training must approximate the relationship between the data and the goal. Mathematical systems are suited to problems which involve numeric rather than symbolic data and in many situations provide the most efficient method of implementing an expert system. If an exact relationship is not known and training of an algorithm is impracticable, then a different method will be required.

These four types of systems are not mutually exclusive and some systems are built using combinations of the techniques to reach a solution to the problem. Each method reduces the scale of the problem by defining the search space in a different manner. Benefits and drawbacks can be found for each technique, with no single method or combination of methods proving to be the best for every application. The problem area under investigation must be carefully studied so that any structure or other search space reducing technique can be exploited to provide the simplest and most efficient expert system solution. Combinations of system types are often implemented using a blackboard structure.

Expert system shells [67,68,69] are used as a quick and easy method to develop expert systems. These shells can have any of these structure types or combination of these structure types and are discussed in Section 4.3

# 4.3 Languages and Shells for Expert Systems

There are many different programming languages used to implement expert systems. These range from the conventional programming languages [70,71] like Pascal and "C", to the languages used primarily by the AI community like Lisp [72] and Prolog [63]. In addition to these, many programming environments have been developed which are based on these languages and which simplify the implementation of systems.

Prolog and Lisp are the two languages most often used by AI programmers and expert system designers. Prolog has been chosen by

the Japanese for their Fifth Generation projects and is also widely used in Europe whilst Lisp is most common in North America.

Prolog is specifically designed for logic programming, and, hence, it is used differently from conventional languages. Writing a Prolog program involves specifying some facts about objects and relationships and then providing rules about the objects and their relationships. These relationships in Prolog are termed "predicates" and the objects are "arguments". When questions are asked about these arguements and their predicates, the Prolog program will infer a solution. Prolog can be viewed as a limited form of logic programming which infers a solution from a set of logical relationships. This provides a way in which many solutions to expert system problems can be viewed.

Lisp, a list processing language, knows little about numbers and, therefore, is of little use for applications which have large numeric computational requirements. Lisp deals mainly with symbols, the structures of lists of symbols and their relationships. There are many different variations of Lisp which limit the portability of Lisp systems between machines, but it provides a convenient vehicle for problems requiring symbolic manipulation. It is suited to areas where little numeric capacity is needed. This type of symbolic processing lends itself well to solving certain knowledge-based system implementation problems.

Prism and Poplog [73] are two useful programming environments used in the development of expert systems. Prism is not yet on the commercial market, but it is used at Hewlett-Packard as an internally

available development tool. It is a Lisp-based environment which Hewlett-Packard have been developing to aid their evolution of systems in the general field of AI. This software environment has not been written for run-time efficiency, but for ease of prototyping and improvement. The development of the machine learning system detailed in Chapter 6 was carried out using Prism and it revealed the significant increase in productivity offered by this type of software environment. Upon completion of the development stage the machine learning system was translated into Pascal to decrease its code size and run-time, and to increase its portability.

Poplog is an environment for list processing and logic programming. There are two programming languages available to the user of Poplog: Pop-11 and Prolog. Both Poplog and Prism provide many utilities to the user to make system development quicker and easier than it would be using conventional techniques. These environments, however, do require more powerful computers to run on than standard techniques. They are complete environments which the user enters upon logging on and remains within until logging out.

Poplog and Prism are best suited to applications in different problem areas as are their base languages. The choice of software environment is, therefore, dependent on the specific application and it should be the type of tool which will best exploit the most convenient methods for the problem solving. A choice of this kind will also be affected by system availability and familiarity of the user.

The expert system shell is a further advancement beyond programming environments which makes system development even more efficient. An expert system shell is simply an expert system with an empty knowledge-base. To create a working system, this empty knowledge-base has to be filled with data in the required form. This puts constraints on the problems which a shell can be expected to handle and so a shell must be specifically tailored to a particular problem type; its areas of application, therefore, are limited. Where suitable shells already exist they allow quick system development, minimising the work required from the knowledge engineer to produce the knowledge-base. The requirements are that:

- A production system shell must have a set of rules for the interpreter to trigger.
- (2) Structured system shells require the frames, nodes and links to be established for the inference engine to interrogate.
- (3) Logically based shells require specification of the logical theorems and relationships that form its knowledge-base.
- (4) The knowledge-base of the mathematically based shell is created by training the algorithms in the shell, or by providing mathematical functions relating the data to the conclusions.

Although these shells can greatly speed up the process of constructing an expert system, there are a few associated problems. In order to make full use of the benefits available from an expert system shell it must be possible to express the knowledge of the domain in a form suitable for the knowledge-base of that shell. If a shell with the desired attributes for tackling a given problem is not

available, it is generally better not to use a shell than to use an unsuitable one. For any particular problem under examination a shell with the correct attributes must be found. If a suitable shell is not available, then the use of a different and less suitable shell will probably create more problems than simply trying to implement an expert system from scratch.

A shell which uses a knowledge-base of an appropriate form and which searches that knowledge-base efficiently can greatly reduce the time required to produce prototype systems, and the major advantage will be the speed at which prototyping can be performed. The prototype allows the performance of the system to be evaluated when solving the given problem. If this performance proves satisfactory, then a final system can be produced with the shell and knowledge-base as a model.

The reason for using shells only for prototyping is that many shells are not written for efficiency of code size and speed of operation, but for ease of generation of an expert system. To increase the speed of operation and the variety of machines which can use the system, these prototypes require to be efficiently implemented.

## 4.4 Present Systems and Their Impact

There has been much research into expert systems in recent years. These systems are often called intelligent knowledge-based systems (IKBS) rather than expert systems or knowledge-based systems. Whatever they are called they all aim to achieve the same goals of

replacing or aiding a human expert. Figure 4.4.1 gives a selection of the systems which have been developed and which have formed the basis for most of the publications on knowledge-based systems. These systems can be used to illustrate the various features in an expert system. They use varying types of knowledge-base and have different search mechanisms which have been developed for theproblem types that are encountered. Most of these systems are diagnostic systems. This is partly because of the high demand for diagnosis, and partly because expert systems are best suited to diagnostic type applications where the problem domain can be kept narrow and well defined.

The three most documented working systems from the literature are MYCIN [74], PROSPECTOR [25] and XCON/R1 [75]. MYCIN is a medical diagnostic system, PROSPECTOR is a system to aid the interpretation of geophysical data for mineral exploration and XCON/R1 is used to aid the configuration of VAX computers.

MYCIN is a backward chaining rule-based system which proved successful in diagnosing antimicrobial therapy. Although MYCIN is not widely used because of its need for a large amount of processing power, it has spawned many other knowledge-based systems which use a similar structure. These systems are used in situations with a smaller problem domain than MYCIN's; the reduction in the domain allows the system to remain at a manageable size. Two systems in particular are derived directly from the MYCIN projects: EMYCIN [15] (Empty or Essential MYCIN) which is an expert system shell using MYCIN's structure and search mechanisms, and NEOMYCIN [15] which is a system to help generate the rule-base for MYCIN.

YSTEM	EXPERTISE	DEVELOPED BY
ASNET VTERNIST DX YCIN JFF ipmeter ivisor	Glaucoma diagnosis Internal medicine Medical consultation Antimicrobial therapy Pulmonary function tests Oil exploration	Rutgers Pittsburg University Ohio State University Stanford Stanford/PMC Schluberger
ENDRAL ROSPECTOR ART/DASD LAS CON/R1 ACSYMA DLGEN DAH ASP/SIAP	Molecular structures Mineral exploration Computer fault analysis Oil well log Circuit analysis Computer configuration Symbolic integration Planning DNA experiments Robotic planning Signals to symbols	Stanford SRI IBM/Stanford Amoco/Rutgers MIT CMU/DEC MIT Stanford SRI Stanford

Figure 4.4.1 A selection of expert systems and their areas of expertise.

PROSPECTOR is a system which is used for mineral exploration. It provides help for geologists in locating certain types of mineral deposits by examining information from the survey data. PROSPECTOR covers only a limited number of geological structure types, but even so it has been successfully used to find a previously overlooked valuable mineral deposit. This provided a demonstration that a strategy which keeps the problem domain sufficiently narrow makes it possible to use these types of systems to solve significant problems.

XCON/R1, a system developed by the Digital Equipment Corporation (DEC) to help configure VAX computer systems, has now been in use for several years. This system has saved DEC several million dollars in producing specific system configurations to meet a customers' specification.

Although there is much literature detailing theoretical architectures for expert systems and their search methods, there seems to be little material on operational systems [76,77,78]. The main problem appears to be that the expectation of the capabilities of knowledge-based systems is too high and, hence, the developed systems require too much processing power to be economically viable. Apart from the systems already mentioned, most of the working systems in existence are small and cover very limited problem areas. By concentrating on practical applications for problems which have realisable solutions, expert systems will come into their own. If too much time is spent on deciding upon the intelligence of a system, or on systems to cope with unbounded problem areas, then expert systems will continue to be very much of a theoretical nature.

As expert systems which work on bounded problem areas are developed it is expected that their use will increase rapidly. The increased use of expert systems will ensure that the systems become more refined and knowledge engineers become more familiar with the types of solution which best suit their problem types. Greater availability of expert systems will release many experts to perform other more productive work, or will simply provide an aid for experts and improve their reliability. Benefits of increased production and reliability without an accompanying increase in personnel will be achieved. Reliance upon certain key personnel, whose absence would otherwise completely stop production, will be reduced. This particular benefit could create a problem: an expert may not be willing to impart his knowledge for a system which could affect his job security. This difficulty, however, is unlikely to be very great as systems will have to be built on very narrow problem domains if their construction is not to prove impracticable. Human experts cover broader domains than could be achieved by an expert system and, therefore, could not be replaced solely by any such system.

The narrow problem domains which would be suitable for expert system application include the areas of automated test and specific fault diagnosis of a wide range of equipment. These are areas of production where reliability and speed of test and adjustment currently present problems. With suitable knowledge-based systems these problems can be minimised and production rates improved. The following chapters in this thesis provide an example of the application of knowledge-based systems to one specific area: diagnosis of faults in digital microwave relay equipment.

### CHAPTER 5

#### RULE-BASED SYSTEM

### 5.0 Introduction

In Chapter 4, four main types of control for knowledge-based systems were described. The first of these, the rule-based system, was selected for the task of fault diagnosis in a digital radio. This method was chosen because it is currently the most commonly used technique for implementing diagnostic expert systems, and intuitively rules with a set of conditions for a given conclusion are the most direct solution to this type of problem. Rule-based systems can require a great deal of "expert" knowledge about the problem and can take a long time to produce, but they do provide an understandable solution to the problem. This comprehensibility arises from the ability to examine the actual rules being used to find out how the diagnosis is actually being performed.

Two approaches to implementing the rule-based system were attempted: one used a Prolog expert system shell and the other was written in the C Programming language. Both of these methods of implementing a diagnostic rule-based system are presented; the Prolog expert system shell is described in Section 5.2 and the C based system in Section 5.3. The rules used for both methods of implementing the rule-based system are the same; the generation of these rules which define the relationships between features and faults, is detailed in Section 5.1. The performance of the C based system, when used for diagnosing faults in the radio model, is

reported in Section 5.4. A brief summary of this Chapter is presented in Section 5.5.

## 5.1 Rule Generation

Figure 5.1.1 shows a flow diagram of the steps involved in the generation of the rules, these steps are described in more detail in the remainder of this Section. The rules which express the relationships between the features and the faults are the same irrespective of which method of implementing the rule-based system is used. In Section 3.4 the variation of each geometric feature with the introduced fault conditions was presented. The figures in Appendix A illustrate the variation between the features and the faults. Certain features formed a distinct relationship to specific fault conditions; these variations of features with introduced faults are shown in Figures 3.4.1 - 3.4.6. Other features, however, demonstrated a less pronounced association with the fault conditions, and these features were not used to provide information about the levels of the specific faults.

The TWT power amplifier overdrive and underdrive faults were most closely related to: the outer compression and the inner expansion of the constellation states, and to the differential rotation of the inner and outer constellation states. The geometric features bearing the clearest association to the introduced gap spacing error in the modulator were the I gap and the Q gap spacing errors. The parameter which demonstrated a clear correlation with the introduced non-orthogonality of the I and Q carriers was the nonorthogonality of the signal constellation.



Figure 5.1.1 The steps involved for rule generation.

These six geometric features formed the basis for the "generation" of the rules to perform the fault detection and to determine the levels of the faults which are present. Some of the geometric parameters showed such a weak coupling to all of the faults under examination, that it was impossible to extract from them the information about which of the distortions were present. These geometric features were not used to generate any of the rules, but they would be employed to determine other unexamined fault types which affected these parameters in a quantifiable manner. The constellation features which were unused were: the I pool deviation, the Q pool deviation, the I.Q pool variance and the correlation coefficient.

The remaining constellation features were used, in conjunction with the features which exhibited a clear relationship to specific faults, to determine whether the radio was well adjusted. A rule was generated which determines if the radio is "normal" (or well adjusted). This is the first test to be carried out on the constellation and it checks that the expected level of each feature within the constellation is, in fact, found. If the radio is established as being correctly adjusted the system will indicate this to the user and terminate its analysis.

To identify the possible fault conditions, it is necessary that rules be written to attribute the variation of specific features to specific fault conditions. This is not a trivial task since the geometric features which vary in a certain manner due to one of the introduced faults can be altered by the presence of another fault. The non-orthogonality of the I and Q carriers and the constellation

spacing faults only affect their own determining features. However, when the TWT amplifier overdrive is 6 dB or greater all of the geometric features start to vary. The variation of all of the parameters is caused by incorrect decisions being made about which signal state specific symbols belong to. The abrupt changes in the feature set indicate that the signal constellation is being received incorrectly. The changes can aid the correct determination of the level of TWT overdrive, but adversely affect the diagnoses of the other fault types. The techniques for overcoming these difficulties are explained in the descriptions of the rules for establishing the gap spacing errors and the non-orthogonality of the I and Q carriers.

In order to establish whether a TWT amplifier underdrive or overdrive fault exists and, if so, what magnitude of fault is present, we use the outer states compression plus inner states expansion (the inner to outer expansion), and the inner to outer states differential rotation, as the determining features. The inner to outer expansion was chosen in preference to using the two geometric features of expansion and compression because both features appeared to vary in a similar manner, and using one combined feature made the writing of the rules simpler. The inner to outer expansion variation with TWT backoff is shown in Figure 5.1.2. The rule to establish whether there is a TWT overdrive present is:

> If the inner to outer expansion is greater than 8.5%, and the inner to outer differential rotation is greater than 3.5 degrees, then there is a TWT overdrive.



Figure 5.1.2 Variation of inner to outer expansion with travelling wave tube backoff.

The threshold values of 8.5% and 3.5 degrees were selected after examining the relationships shown in Figures 5.1.2 and 3.4.3.

TWT underdrive is determined by the following: If the inner to outer expansion is less than 8.5% and the inner to outer differential rotation is less than 3.5 degrees, then there is a TWT underdrive.

To establish whether there is an error present in the coding levels the following rule is used:

If the I gap spacing error or the Q gap spacing error are not zero, then there is a spacing error present.

The presence of an error in the orthogonality of the I and Q carriers is concluded by:

If the measured non-orthogonality of the constellation is not zero, then there is an error of non-orthogonality of the I and Q carriers.

These rules determine whether the radio is normal or if there are any of these faults present. However, if the radio is not normal and there are no faults present, there is a rule to indicate that the radio is in an unknown condition and the processing is stopped.

The remaining rules establish the levels of each fault present. If there is a TWT overdrive or TWT underdrive fault, then there are a set of rules to establish the level of overdrive and underdrive. These rules use the expected values of inner to outer expansion and inner to outer differential rotation for each specific level of TWT backoff. These expected values were found by examining the relationships shown in Figure 5.1.2 and Figure 3.4.3. When the expected values of these features correspond to a specific level of overdrive or underdrive, the conclusion formed by the rule is that this is the actual level of the TWT overdrive or underdrive. An example of one of these rules is:

> If the inner to outer expansion is 17% and the inner to outer differential rotation is 7.5 degrees then there is a TWT amplifier overdrive of 4 dB.

These rules are used if a TWT overdrive or underdrive has been indicated by the earlier rules.

Several factors must be accounted for before the level of a constellation spacing error can be determined. The receiver is required to "lock" onto a signal constellation which is generated by a pseudo random binary sequence (PRBS). Since there is no information for the receiver to establish a reference phase, it is possible for the receiver to lock onto the signal in any one of four orientations, corresponding to phase shifts of 90 degrees. To overcome the problem of the I and Q values and the sign of the errors being interchangeable, the sum and the difference of the I and Q gap errors are used as the determining parameters. This removes information regarding whether the fault occurred in the I or Q leg of the coder, but the magnitude of the fault can still be diagnosed.

The variation of the I plus Q gap spacing errors with the introduced spacing errors is shown in Figure 5.1.3. Two rules are used to establish the level of the gap spacing error: one is used when there is a TWT overdrive fault and the other if there is no TWT overdrive present. The two rules are required because the I and Q gap error features are altered for high levels of TWT overdrive. In the presence of TWT overdrive, the I and Q gap errors increase by approximately equal amounts so the spacing error in the modulator can be estimated by taking the difference between the I and Q gap errors. The calculation of the I and the Q gap errors introduces a scaling factor of two over the introduced spacing error, and this scaling is removed by the rules.

The two rules for determining the level of the spacing errors are:

If there is a spacing error and there is no TWT overdrive, then the level of the spacing error is interpreted as twice the sum of the I gap and Q gap spacing errors.

If there is a spacing error and there is also a TWT overdrive present, then the level of the spacing error is identified as twice the difference between the I gap and the Q gap errors.

The error on the estimate of the spacing error is increased by the presence of TWT overdrive.

The level of the non-orthogonality of the I and Q carriers is determined by the measured non-orthogonality of the signal

Note The rules used will not correctly identify the gap spacing errors if the I and Q channels are both in error by the same amount and there is a TWT overdrive.



% Introduced gap spacing error

Figure 5.1.3 Variation of I plus Q gap spacing errors with the introduced gap spacing error.

constellation. However, if there is a TWT overdrive present then the confidence in the estimate of the level of the non-orthogonality error is diminished.

There are also rules which provide an explanation of the fault diagnosis that has been performed. The explanatory rules examine which faults are present, and indicate to the user the values of the features which were used to determine the faults.

## 5.2 Prolog Shell

An expert system shell written in Prolog was used for the first attempt at developing a rule-based system. The shell (KS-299) is based on a shell written by Tecknowledge Inc. (KS-300). Appendix C is a listing of the source code for this shell. The Prolog expert system shell is an interpreter of a knowledge-base of rules which must be of the form:

Rule N : if PREMISE then CONCLUSION

Where rules 1 to N correspond to combinations of PREMISEs and CONCLUSIONs.

A PREMISE is a simple proposition about the value of a variable, or that a fact is known or unknown. It can also be a combination of propositions built up using "and" and "or".

A CONCLUSION can either simply state the assumed value of a variable or it can also include a level of confidence in its assumption. A CONCLUSION may also be a combination of simple conclusions built up using "and".

The confidence in a CONCLUSION expresses the degree of belief in that CONCLUSION and in our system it lies in the range from 0 to 1000 inclusive. A confidence level of 0 indicates that there is no belief in the CONCLUSION at all. The reliance upon a CONCLUSION is unqualified if the level of confidence has a value of 1000.

The knowledge-base also contains a list of questions which can be asked of the user to ascertain facts or the values of variables. These are given in the form:

QUESTION obtains "information" - where a QUESTION is a question to ask the user and "information" is the variable or fact which answers the QUESTION.

Figure 5.2.1 is an example of part of the knowledge-base used with this shell to perform the fault diagnosis of the digital microwave radio equipment. The complete knowledge base used is in Appendix D. Figure 5.2.2 is a flow diagram of the sequence of tasks performed by the shell in performing its fault diagnosis.

The system is similar to MYCIN in that it performs its search of the knowledge-base by backward chaining. When asked to find "information" it searches to see if a QUESTION refers to the "information". If there is, the QUESTION is used and the shell will not use any rules to establish the "information". If, however, there is no suitable QUESTION, the rules are searched for those which have a CONCLUSION that indicates a value for the "information". The system then attempts to satisfy the PREMISE of each of these rules. This is performed by finding all of the rules with a CONCLUSION

```
'What is the correlation coefficient'
      finds correlation.
'Are the I and Q gap errors different (y=>1, n=>0)'
      finds iq diff.
rule 1:
           if expan=8.5
                and iqgap=0
                and rot=0.5
                 and nonorth=0.25
                and ratio=1
           then fault=normal cf 950.
rule 2:
           if expan=12 or expan=17 or expan=22.5 or expan=27.5
           then expan_cond=overdrive_expan.
rule 3:
                expan=5.5 or expan=3.5 or expan=2.5 or expan=1.5
           if
           then expan_cond=underdrive expan.
rule 4:
           if drot=0.5 or drot=1.5 or drot=2.5
          then drot_cond=underdrive drot.
```

93

Figure 5.2.1 Excerpt from the knowledge-base used with the prolog hell.



Figure 5.2.2 The sequence of tasks performed by the shell for its fault diagnosis.

relating to each PREMISE before moving on to the next PREMISE. The shell first attempts to satisfy those rules which are non-recursive, that is rules in which the PREMISE and the CONCLUSION do not both refer to the "information". This enables the shell to establish something about the "information" preventing it from using the same rule to complete its own PREMISE when attempting to satisfy a recursive rule.

The system uses confidence factors in a similar way to MYCIN. The confidence in the PREMISE of a rule is assigned to the CONCLUSION of that rule. This confidence level will then be modified by the confidence factor specified in the CONCLUSION of the rule. If the confidence in the PREMISE is A, and the confidence given in the CONCLUSION of the rule is B, then the new confidence C in the CONCLUSION is given by:

$$C = 1000*(1-(1-A/1000)*(1 - B/1000))$$
(5.2.1)

Unless A and B are independent, there is no theoretical basis for this method of manipulating confidence factors. However, as A and B are normally subjectively determined by a human "expert" this is not too important. Provided the depth of search (the number of levels of rules used) is not too great (less than five levels), this method of manipulating confidence levels is documented as working in practice.

When trying to establish the PREMISE the shell attempts to satisfy each part of the PREMISE in turn. The confidence in the PREMISE is given as the lowest confidence of any conjunction

(A and B) or the highest confidence of any disjunction (A or B) that occurs in the PREMISE. If the confidence is less than 200 the PREMISE will not be satisfied. If in a conjunction, working from left to right, the system cannot satisfy any one condition, then it will not try to satisfy the others and will fail the PREMISE. In a disjunction, working from left to right, the shell will stop as soon as it establishes a condition with a confidence of 1000 and will satisfy the PREMISE without attempting to satisfy the rest of the disjunction.

The system has no means of ensuring that a user's reply is valid, nor does it permit the user to specify a confidence in a reply; it simply assumes a confidence level of 1000. The system cannot access any of the arithmetic function of Prolog and so rules requiring any arithmetic cannot be written. Care is needed if recursive rules are to be used. If there is a recursive rule, then there also must be a non-recursive rule with the same 'information' in its CONCLUSION to prevent the recursive rule initiating an endless loop of calls on itself, trying to satisfy its own PREMISE.

The knowledge-base generated to perform the fault diagnosis of the digital radio model consisted of 12 QUESTIONS to interrogate the user on the values of the constellation features. To detect the faults there were 27 rules in the rule-base. The rules detected single fault conditions, but not the interactions caused by multiple faults. There were no recursive rules used, which ensured that their associated problems were avoided. The greatest depth of search by these rules was three, thus the method of manipulation of the confidence factors will remain adequate. The knowledge-base was simple to generate using the expert system shell for development. Any information which could be elicited from the user was specified as a question in the standard form for the system. The rules which made use of this user-entered information were put into the knowledge-base in the standard form. The order in which the rules are entered into the knowledge-base is unimportant as the shell's interpreter searches the rules using a backward chaining search. The above factors combine to ensure that the only difficult task involved in generating the knowledge-base is defining the rules for establishing each fault level.

The twelve questions and twenty seven rules covered all the single fault conditions for the four faults under investigation. The faults were:

TWT amplifier overdrive up to 8 dB in 2 dB steps TWT amplifier underdrive up to 8 dB in 2 dB steps Unequal constellation spacing levels from -10% to +10% in 1% steps Non-orthogonality of I and Q carriers up to 5 degrees in 1 degree steps.

When the system was tested on the digital radio model, with single introduced fault conditions, all of the faults and their levels were correctly detected. This performance is not surprising since there is a separate rule corresponding to each level of each fault condition. The shell was subsequently abandoned since one requirement on the system is that it can diagnose multiple faults. With this shell a separate rule corresponding to each combination of faults and levels would be required. The number of rules required to cover all these possible combinations of fault conditions would be 1134. This would result in a knowledge-base which would be too large to be practicable, but if the shell had the facility to access simple arithmetic functions (addition, subtraction and inequalities), then the number of rules needed would have remained manageable.

Although this shell was abandoned it did prove to be a suitable prototyping vehicle for a simple rule-based system. The generation of the knowledge-base was simple and straight forward as only a standard form of questions and rules was required and its operation was not dependent on their order. This shell would provide an excellent method of generating an expert system if the determining features for every fault encountered in the problem were mutually exclusive. Once a system was generated and tested, it could be translated from the Prolog shell and knowledge-base to another programming language. This would allow the system's code to be more compact, efficient and machine portable. If the TWT overdrive had been kept to less than 4 dB, then the determining features for the distortions would have been mutually exclusive and this shell could have been successfully used. However, without modifying the shell it was not suitable for this application. Modification of the shell would have been a major undertaking, and it was preferable to put effort into an alternative approach to solving the problem.

## 5.3 C Based System

Having tried the Prolog shell to implement the rule-based diagnostic system and found it to be of limited value, an alternative approach was chosen. The programming language 'C' was chosen to implement the system because C expertise is available within the Department of Electrical Engineering at Edinburgh University.

The C programming language was originally designed for, and implemented on, the UNIX [79] operating system which is used by all the Department's computing machines. The C programming language has control flow, data structures and a large set of operators which have not been restricted to one particular area of application. It is, therefore, a suitable language to use during the development of an expert system.

A structure for the system had to be decided upon now that it was no longer constrained by the expert system shell. The method an 'expert' would use to go about performing the fault diagnosis was first examined. One possible approach an expert might use is shown in Figure 5.3.1. This method can be viewed as a blackboard type technique, where the expert initially forms a hypothesis as to what faults are present and stores this as if on a 'blackboard'. A set of rules then uses this hypothesis to form a further set of hypotheses on the magnitude of each of the fault conditions which are present. This information is then examined to determine if it agrees with the data available from the signal constellation feature set. If there is any disagreement the process is repeated to alter the nonconforming hypotheses. When the constellation data and the hypotheses agree, the faults, their levels and an explanation of the reasoning behind the diagnosis is given.



Figure 5.3.1 An expert's approach to digital radio fault diagnosis.

The rule-based system written in C has a similar structure to the approach shown in Figure 5.3.1, but without the option to rerun the process to alter incorrect hypotheses. Figure 5.3.2 details the structure of the rule-based system. The rule-based system does not have the option to alter hypotheses because all of the information available to the knowledge-base is dmbedded in the rules for generating the diagnoses. The expert probably has additional information which he does not initially use for his diagnosis and this helps him to decide whether the final diagnosis is correct or not. This may be information previously learned (from another expert, literature of experience), or it may be something that was not originally noticed. The rule-based system has to use all the knowledge available in one pass to perform the fault diagnosis, and it has no facility to 'learn'; any extension to the operation has to be achieved by physically modifying the rule-base. The lack of a built-in 'learning' capability is the main operational difference between the C rule-based system and the human 'expert'.

The system performs its fault diagnosis by initially accessing the signal constellation features which provide the information about the constellation states' geometry. These features are then quantised to a set of discrete values to allow a reduction in the number of rules required to perform the fault detection. The limited number of values permitted for each feature implies that a rule's PREMISE can indicate that the level of a feature needs to be a specific value. A range of values would have to be specified in the rules if there were no restriction on the values that each feature could have. To increase the system's resolution extra levels would be required for each feature at the quantisation stage, and


Figure 5.3.2 The structure of the rule based system.

additional rules to deal with these intermediate feature and fault levels would need to be written. The existing rules in the knowledge-base would, however, remain unchanged. If the rules were written to cover ranges of feature levels, a complete new set of rules would be needed to improve the system's accuracy.

The quantised feature levels provide the input to the rules which generate a hypothesis on the known faults could be present. The possible radio conditions are combinations of:

- (1) Radio is normal
- (2) TWT amplifier overdrive or underdrive
- (3) Error in the signal constellation levels
- (4) Non-orthogonal I and Q carriers
- (5) Unknown fault.

When the features all correspond to the expected values for a normal working radio, the system indicates no fault and ceases processing.

Distortions are indicated by:

If the inner to outer expansion is above 8.5%, and the inner to outer rotation is above 3.5 degrees, then a TWT amplifier overdrive is indicated.

If the inner to outer expansion is less than 8.5%, and the inner to outer rotation is less than 3.5 degrees, then a TWT amplifier underdrive is suggested.

If the I or the Q gap spacing does not equal 0%, a constellation signal spacing level error is indicated.

A non-orthogonal I and Q carrier error is suggested if the measured non-orthogonality of the constellation is not 0 degrees.

If the radio is not 'normal' and none of the known fault conditions are indicated the system outputs 'fault unknown' and stops processing.

The initial hypothesis about which distortions are present is used as the input for the rules which establish the magnitude of the faults. If there is a TWT amplifier overdrive fault present, its magnitude is estimated from the inner to outer expansion and the inner to outer rotation values. These two features are also used to determine the value of the TWT amplifier underdrive if it is present. TWT amplifier overdrive and underdrive are mutually exclusive fault conditions. The level of spacing error, if present, is estimated using the sum of the I and Q spacing errors when there is no TWT amplifier overdrive. If there is TWT amplifier overdrive and constellation spacing errors, the difference between the I and Qspacing is used to establish the value of the spacing error. To determine the level of the non-orthogonality of carriers the value of the measured non-orthogonality of the signal constellation is used. These should give an estimate of the levels of the faults which have been detected. However, if a fault's presence is indicated, the geometric features may not exactly match the PREMISE of any of the rules detailed above. If this is the case then the next set of rules is triggered.

This set of rules will estimate the level of the faults using a subset of the information available from the signal constellation features. The confidence in the conclusions of these rules is not as great as the confidence in the previous set of rules since some information has been ignored. These rules use the geometric feature which has the strongest dependence on the introduced distortion.

When all of the distortions which are present and their levels have been established, the rules which provide an explanation of the fault diagnosis are implemented. These rules detail the determining features of each detected distortion. When the indicated faults have been identified, their levels and explanations of the diagnosis are output to the user to aid decisions about what remedial action should be taken. The explanations are included to permit the user to make a separate evaluation of the diagnosis. If he disagrees with any aspect of the system's conclusions, then different action from the recommended can be taken.

This structure for the system was chosen as it seemed a 'natural' way to perform the diagnosis, while still remaining relatively easy to implement in C. By partitioning the rule-base into sets of rules, which put forward hypotheses about the faults and their levels, the ordering of the rules was simplified. The various sets of rules also ensured that the addition and modification of rules would be less complicated than if one large rule-set had been used. This follows because the rules in each rule-set take the problem only one step further without involving complicated interactions between rules. The transfer of information between

rule-sets is performed by the intermediate hypotheses, with the outputs from one set forming the inputs to another rule-set.

This modular approach provided a workable technique for producing a rule-based expert system for performing the fault diagnosis. The constellation data and the required quantisation levels form the input to the program. This information is processed by forty rules similar to those used in the Prolog shell, but using arithmetic capabilities to handle multiple fault conditions as well as single fault conditions. The rules are of the same form as the Prolog shell:

If PREMISE then CONCLUSION

Where the PREMISE can be any combination of conjunctions and disjunctions of propositions, and the CONCLUSION is a combination of conjunctions of conclusions.

## 5.4 <u>Performance of the C Based System</u>

The rule-based system was tested with a variety of introduced fault conditions. Initially the testing examined the performance when diagnosing faults on the radio model with only one distortion type present at any given time. The faults investigated were:

TWT amplifier overdrive and underdrive, non-orthogonality of the I and Q carriers and errors in the constellation spacing levels. These distortions are detailed in Section 3.3. Figure 5.4.1 shows the ranges of the faults examined and Figure 5.4.2 summarises the

Fault	Range	Stepsize
TWT overdrive	up to 8dB	2dB
TWT underdrive	up to 8dB	2dB
Non-orthogonality of the I and Q carriers	up to 5 degrees	1 degree
Constellation spacing errors	-10% to +10%	1%

Figure 5.4.1 The ranges of faults examined using the knowledge-based systems.

Fault levels:

Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected

#### Level

Non-orthogonal carriers	1	Non-orthogonal carriers	1
Non-orthogonal carriers	3	Non-orthogonal carriers	3
Non-orthogonal carriers	5	Non-orthogonal carriers	5
		Gap spacing error	0
Gap spacing error	- 9	Gap spacing error	-9
		Non-orthogonal carriers	0
Gap spacing error	- 6	Gap spacing error	-6
		Non-orthogonal carriers	0
Gap spacing error	-3	Gap spacing error	-3
		Non-orthogonal carriers	0
Gap spacing error	3	Gap spacing error	3 .
		Non-orthogonal carriers	0
Gap spacing error	6	Gap spacing error	6
		Non-orthogonal carriers	0
Gap spacing error	9	Gap spacing error	9
		Non-orthogonal carriers	0
TWT overdrive	8	TWT overdrive	8
		Gap spacing error	0
TWT overdrive	6	TWT overdrive	6
		Gap spacing error	0
TWT overdrive	4	TWT overdrive	4
TWT overdrive	2	TWT overdrive	2
TWT underdrive	2	TWT underdrive	2
TWT underdrive	4	TWT underdrive	4
TWT underdrive	6	TWT underdrive	6
TWT underdrive	8	TWT underdrive	8

Figure 5.4.2 The output from the rule-based system for diagnosing single fault conditions

results obtained from single fault condition tests. The system was seen to correctly identify all of these fault conditions.

The system was then tested with combinations of two faults introduced at one time. These results are detailed in Figure 5.4.3. Finally, the performance of the rule-based system was examined with three simultaneously occurring faults. Figure 5.4.4 details the results obtained from the system for three faults present at one time.

The single fault conditions were all correctly detected by the rule-based system for all the ranges tested. The TWT amplifier overdrive and underdrive faults were detected in 2dB steps over the full range. With an 8 dB or 6 dB TWT amplifier overdrive a spacing error was also indicated but with a value of 0%. The non-orthogonal I and Q carrier faults were correctly identified over the range 0 degrees to 5 degrees in 1 degree steps. However, for a 5 degree non-orthogonality a spacing error of 0% was again indicated. The spacing errors were correctly detected from -10% to +10% in 1% steps, while a non-orthogonal carrier fault of 0 degrees was specified. Thus the rule-based approach correctly detected all the individually present faults. In certain cases additional faults were indicated as occurring, but with zero level.

The detection of the multiple fault conditions, including both double and triple fault conditions, was not as accurate as that of the single fault conditions. For the cases where there was no TWT amplifier overdrive of 6 dB or greater, the diagnoses of the radio condition were as accurate as for singly occurring faults. However,

Fault levels:

Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected Level

Gap spacing error	3	Gap spacing error	3
Non-orthogonal carriers	3	Non-orthogonal carriers	3
TWT underdrive	4	TWT underdrive	4
Non-orthogonal carriers	3	Non-orthogonal carriers	3
TWT overdrive	8	TWT overdive	8
Non-orthogonal carriers	3	Non-orthogonal carriers	0
		Gap spacing error	1
TWT overdrive	2	TWT overdrive	2
Non-orthogonal carriers	3	Non-orthogonal carriers	3
		Gap spacing error	0
TWT overdrive	6	TWT overdrive	6
Non-orthogonal carriers	3	Non-orthogonal carriers	2
		Gap spacing error	-4
TWT underdrive	4	TWT underdrive	4
Non-orthogonal carriers	1	Non-orthogonal	1
TWT overdrive	8	TWT overdrive	8
Non-orthogonal carriers	1	Non-orthogonal carriers	0
<u> </u>	-	Gap spacing error	0
TWT underdrive	4	TWT underdrive	4
Gap spacing error	6	Gap spacing error	6
<u>1</u>		Non-orthogonal carriers	0
TWT overdrive	2	TWT overdrive	2
Gap spacing error	6	Gap spacing error	6
No.		Non-orthogonal carriers	0
TWT overdive	6	TWT overdrive	6
Gap spacing error	6	Gap spacing error	5
		Non-orthogonal carriers	0
TWT overdrive	8	TWT overdrive	8
Gap spacing error	6	Gap spacing error	1
		Non-orthogonal carriers	0

	111					
¥.	Faults Introduced	Level	Faults Detected	Level		
TWT	underdrive	4	TWT underdrive	4		
Gap	spacing error	3	Gap spacing error	3		
			Non-orthogonal carriers	0		
TWT	overdrive	2	TWT overdrive	2		
Gap	spacing error	3	Gap spacing error	3		
			Non-orthogonal carriers	0		
TWT	overdrive	6	TWT overdrive	6		
Gap	spacing error	3	Gap spacing error	2		
			Non-orthogonal carriers	0		
TWT	overdrive	8	TWT overdrive	8		
Gap	spacing error	3	Gap spacing error	1		
			Non-orthogonal carriers	0		

Figure 5.4.3 The output from the rule-based system for diagnosing double fault conditions.

Fault levels: Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected Level

TWT overdrive	2	TWT overdrive	2
Gap spacing error	9	Gap spacing error	9
Non-orthogonal carriers	1	Non-orthogonal carriers	1
TWT overdrive	6	TWT overdrive	6
Gap spacing error	3	Gap spacing error	0
Non-orthogonal carriers	3	Non-orthogonal carriers	2
TWT overdrive	6	TWT overdrive	6
Gap spacing error	6	Gap spacing error	6
Non-orthogonal carriers	1	Non-orthogonal carriers	1
TWT overdrive	8	TWT overdrive	8
Gap spacing error	3	Gap spacing error	2
Non-orthogonal carriers	3	Non-orthogonal carriers	0
TWT overdrive	2	TWT overdrive	2
Gap spacing error	3	Gap spacing error	1
Non-orthogonal carriers	3	Non-orthogonal carriers	3

Figure 5.4.4 The output from the rule-based system for diagnosing triple fault conditions.

for multiple faults with TWT amplifier overdrive of 6 dB or more there were some errors. The system always specified the correct level of TWT amplifier overdrive, but the other faults were erroneously diagnosed. The presence of the other faults was detected, but their levels were incorrectly identified. This problem arises from the fact that the high levels of TWT amplifier overdrive affect the features which were used to determine the other fault conditions. The features that establish the TWT amplifier overdrive were not sensitive to the levels of the other faults present.

The performance of the fault detection of multiple faults occurring simultaneously would be improved by using a two stage diagnostic process. The first step would involve the detection and removal of any TWT overdrive maladjustment, the second stage being the diagnosis and correction of the remaining faults. Thus, the effect of a TWT overdrive impairment that hindered the estimation of other fault conditions would be removed.

#### 5.5 Summary

The relationships between the features of the constellation and the introduced impairments in the digital radio model, presented in Section 3.4, provided suitable information to create a set of rules to perform the fault diagnosis of the digital radio model. Only those fault/feature relationships which provided clear information regarding specific faults and their levels were used. From these relationships a set of rules was generated which were used to form the knowledge-base for both the Prolog shell and for the Cbased system.

The Prolog shell proved to be a fast vehicle for producing a prototype system, but it had no arithmetic capabilities so the most efficient forms of the rules could not be used. Thus, this method was only used to produce a system to diagnose and detect single fault conditions. A C-based system using the rules obtained by examining the fault feature relationships was produced to detect and diagnose both single and multiple occurring faults. The rules for this system were entered in the order which they would be used for the fault analysis. The performance of the C-based system proved good, its limitations in accuracy being attributable to the quantisation of the written rules. This accuracy would be improved by a greater number of rules. The main drawback of this system is the work required to create it. The knowledge engineer has to generate all the required rules (more rules are needed for improved accuracy) and these rules must then be ordered to ensure correct operation of the system. However, this method did produce a working system which could diagnose the range of fault conditions, in the digital radio model, being examined.

#### 115

#### CHAPTER 6

### MACHINE LEARNING SYSTEM

### 6.0 Introduction

Knowledge-based systems in general require a considerable amount of work from a knowledge engineer to imbed in the system the information that is required to perform the specified task. The great effort needed will not necessarily prove prohibitive to the system construction if: either there are many identical units required, or the benefits accrued from one or a few units are considerable. However, certain applications require many similar systems (units with the same structure but different contents of their knowledge-bases) to cover all of the different uses of the system. This is the case for fault diagnosis of 16 QAM digital radios. There are many different types of 16 QAM radio, each of which could use the same structure of system to perform the fault diagnoses, but would require different knowledge-bases.

One method of overcoming the problem of the quantity of work required from a knowledge engineer is to produce a system of a suitable structure which can generate its own knowledge-base. This is termed a machine learning system and the 'learning', or creation of the knowledge-base, is performed by training the system on specific examples of fault conditions. A machine learning system requires work by a knowledge engineer to produce a satisfactory structure for the system, and operation initially in a training mode to encode the information into the system. When an appropriate system is produced it can be relatively easily trained for its required application. This type of system can be a simple decision tree system which incorporates a training mode to produce its knowledge-base, or it can be an adaptive system which uses adaptive algorithms to create the knowledge-base. The machine learning system described in this chapter uses adaptive algorithms to acquire, from the training examples, the information required to perform the fault diagnosis of the digital radio equipment.

The machine learning system uses techniques from two separate areas: distance classifiers from geometry and pattern analysis [80], and a recursive least squares (RLS) algorithm from adaptive filtering [81]. The structure of the machine learning system, and the ways in which the distance classifier and adaptive filtering techniques are implemented within the structure, are detailed in Section 6.1. A justification for the particular form of the algorithms chosen for the system is also given. The performance of the machine learning system in tests on a real radio is catalogued in Section 6.2 with Section 6.3 providing details of the results obtained for the fault diagnoses of the digital radio model.

In Section 6.4 a brief comparison of the rule-based and the machine learning system is made, and the choice of a hybrid system which uses techniques from both the rule-based and machine learning approach is described. The performance of the hybrid system in tests on the radio model is detailed in Section 6.5.

## 6.1 Structure of the Machine Learning System

The machine learning system has two 'levels' of operation: the first level uses a distance classifier to separate the problem into distinct conditions, and the second level uses recursive least squares estimators to ascertain the magnitude of each fault. Figure 6.1.1. shows the two level structure. The information input to the machine learning system, as for the rule-based systems, consists of the geometric features of the signal constellation. Both levels, the distance classifiers and the recursive least squares estimators, use the geometric features as the information for performing their processing.

The distance classifier used by the system employs the Mahalanobis distance [80] which is a matrix form of distance classifier. A distance classifier measures the distance in a geometric space between two points. The Mahalanobis classifier weights this distance depending upon the distribution of the points forming a cluster in the geometric space. When used in the machine learning system the Mahalanobis distance is implemented in an ndimensional space, corresponding to the n geometric features of the signal constellation. During training, feature sets are input to the system along with the corresponding condition of the radio. The three conditions of the radio shown in Figure 6.1.1 are: well conditioned, out of lock, and ball of noise. The 'well conditioned' mode corresponds to the radio condition when the signal constellation shows 16 distinct signal states. 'Out of lock' is the radio condition when the receiver carrier recovery circuitry fails to lock onto the signal and the constellation displays three concentric rings, an example of which is shown in Figure 6.1.2. 'Ball of noise' is the term given to the case when there is no amplitude or phase



Figure 6.1.1 The structure of the machine learning system.

information recovered and the signal constellation appears as one state of noise; Figure 6.1.3 shows an example of a signal constellation of a radio in this condition.

Several examples of geometric feature sets corresponding to each of the radio conditions are input during training. These feature sets form a cluster of points, one cluster for each condition, in the n-dimensional space (the position of each point in the space is determined by the corresponding feature set). A cluster in this feature space is also formed for good radios which pass the bit error rate (BER) test. After training is complete, the Mahalanobis distance of a radio (given by the feature set of that radio) from the centre of each radio condition cluster provides a measure of how close the radio is to that particular condition. The shortest Mahalanobis distance to any one cluster gives the best estimate of the condition of the radio. If the radio is well conditioned, that is there are 16 distinct signal states, the Mahalanobis distance from the mean of the radios which passes the BER test gives a measure of how far the radio is from being correctly adjusted. If the radio is not well conditioned, the system outputs the state to which the feature set is closest (out of lock or ball of noise in Figure 6.1.1) and no more analysis is performed. For a well conditioned radio the feature set is analysed further in the least squares estimation section of the system.

The Mahalanobis distance,  $r^2(\underline{x}(i),\underline{m}(j))$ , from  $\underline{x}(i)$ , the feature set, to  $\underline{m}(j)$ , the cluster centroid formed by the jth trained condition, is:



Figure 6.1.2 The signal constellation of a radio with the receiver phase out of lock.



Figure 6.1.3 The signal constellation of a radio showing ball of noise.

$$r^{2}(\underline{x}(\underline{i}),\underline{m}(\underline{j})) = (\underline{x}(\underline{i}) - \underline{m}(\underline{j}))^{T}\underline{C}(\underline{j})^{-1}(\underline{x}(\underline{i}) - \underline{m}(\underline{j}))$$
(6.1.1)

Where:

 $\underline{x}(i) = [x(i) \ x_2(i) \ \dots \ x_n(i)]^T$  is the ith feature set under investigation.

 $\underline{m}(j) = [m_1(j) \quad m_2(j) \quad \dots \quad m_n(j)]^T$  is the mean of the jth cluster.

 $\underline{C}(j)$  is the covariance matrix formed from the jth cluster's training set defined as:

$$\underline{C}(j) = \underline{E}(\underline{x} \ \underline{x}^{\mathrm{T}}) - \underline{m}(j)\underline{m}^{\mathrm{T}}(j)$$
(6.1.2)

 $\underline{C}^{-1}(j)$  is the inverse of this covariance matrix.

The inverse covariance matrix,  $\underline{C}^{-1}(j)$ , weights the distance due to each feature from the feature set in inverse proportion to the variance of that particular feature during the training of that cluster. This prevents one feature with a large variance dominating all of the other features in determining the distance from a cluster. The matrix,  $\underline{C}^{-1}(j)$ , also takes into account the effect of correlation between features, by applying the appropriate weightings from its off diagonal elements in the distance calculation.

The adaptive least squares estimator [81,82,83,84] section of the system is structured as shown in Figure 6.1.4. There is one linear combiner corresponding to each fault under investigation. The geometric feature set, Section 2.4, forms the input to the adaptive combiners, instead of using a time shifted signal input as is more



Figure 6.1.4 The structure of the linear combiners in the machine learning system.

commonly used in adaptive filtering. Each feature is fed directly to one of the taps of all of the combiners. The inner to outer ratio and the correlation coefficient are not used by the adaptive combiners. These two features are only used by the distance classifier algorithm to determine the condition of the radio. In addition to these ten geometric features the squares of the first order features are used as input to these combiners. The second order features (the I pool deviation, the Q pool deviation and the I.Q pool variance), are not squared. This is unnecessary as they are already of second order which allows the combiner to directly form a quadratic estimate of the relationship between the fault and the features.

The squared values of the first order features are taken so the adaptive combiners can account for non-linear effects. The adaptive combiners are linear classifiers and if only first order features are included as inputs, then only a first order or linear approximation [85] of the fault versus features relationship will be formed. Including second order terms in the inputs to the linear combiners produces a second order or quadratic approximation of the fault versus feature relationship. The quadratic approximation indicates that the relationships which are not purely linear can be estimated. A relatively small number of training examples are required for a second order approximation; a minimum of three examples is needed. Higher order approximation would incur the penalty of requiring a greater amount of training. The combiners are adaptively trained using a recursive least squares (RLS) algorithm [81], (see later for mathematical definition). The error in this

quadratic approximation of the fault versus feature relationship is minimised in a least squares sense.

The RLS algorithm was chosen as it provides fast convergence, minimising the training examples required, to the optimum combiner weights in a least squares sense. It also permits continuing training which allows new data to be input to improve the system's performance.

The structure of one combiner corresponding to a single fault is shown in Figure 6.1.5.

The tap weight adaption, which is performed during the training phase using an RLS algorithm, takes the features (including second order terms) as the inputs to the combiner. The magnitude of the fault corresponding to that particular combiner forms the training signal. Each combiner uses the same input feature set, but has a different fault type as a training signal corresponding to the particular fault the combiner is estimating. When the training phase is complete, the tap weights are fixed and remain unchanged until further training is initiated. To determine the levels of fault on a radio with arbitrary faults, the features (including the second order terms of each feature) are input to the combiners. The output from each combiner is the sum of the product of the tap weights with the input features. The output then represents the level of the specific fault that the combiner has been trained on.

For an input feature set the level of the fault, which corresponds to that combiner is given by:



Figure 6.1.5 The structure of each adaptive combiner.

$$y = \underline{x}^{T}\underline{h}$$

Where:

y is the estimate of the fault level  $\underline{x} = [x_0 \ x_1 \ \dots \ x_n]^T$  is the input feature set including the second order terms  $\underline{h} = [h_0 \ h_1 \ \dots \ h_n]^T$  is the coefficient vector

The deviation of this estimate from the actual measured fault level is:

$$e = y - y$$
 (6.1.4)

Where:

e is the deviation y is the actual fault level

The optimum tap weights  $(\underline{h}_{opt})$  are calculated to minimise the sum of the squared errors over the k training examples.

$$\frac{h}{opt}$$
 minimises  $\sum_{n=0}^{k} (y(n) - y(n))^2$ 

This value of  $\underline{h}_{opt}$  is given by the Wiener-Hopf equation:

 $\underline{\mathbf{h}}_{\text{opt}}(\mathbf{k}) = \underline{\mathbf{P}}(\mathbf{k}) \bullet \underline{\mathbf{r}}_{\mathbf{X}\mathbf{Y}}(\mathbf{k})$ (6.1.5)

Where:

 $\underline{P}(k) = \underline{r}_{xx}^{-1}(k)$  the inverse of the autocorrelation matrix formed by  $\underline{x}(k)$ 

$$\underline{\mathbf{r}}_{\mathbf{X}\mathbf{X}}(\mathbf{k}) = \sum_{n=0}^{\mathbf{k}} \underline{\mathbf{x}}(n) \underline{\mathbf{x}}^{\mathrm{T}}(n)$$
(6.1.6)

$$\underline{\underline{r}}_{xy}(k) = \sum_{n=0}^{k} \underline{\underline{x}}(n) \underline{\underline{y}}(n)$$
(6.1.7)

 $\underline{r}_{xy}(k)$  is the cross correlation of  $\underline{x}(k)$  and  $\underline{y}(k)$  over the k training examples.

For the set of k training examples these optimum tap weights can be calculated recursively. This allows progressive training without storing all of the previous data explicitly.

Equations 6.1.6 and 6.1.7 can be expressed recursively as

$$\underline{\underline{r}}_{XX}(k) = \underline{\underline{r}}_{XX}(k-1) + \underline{\underline{x}}(k)\underline{\underline{x}}^{T}(k)$$
(6.1.8)

$$\underline{\mathbf{r}}_{\mathbf{x}\mathbf{y}}(\mathbf{k}) = \underline{\mathbf{r}}_{\mathbf{x}\mathbf{y}}(\mathbf{k}-1) + \underline{\mathbf{x}}(\mathbf{k})\underline{\mathbf{y}}(\mathbf{k})$$
(6.1.9)

Substituting 6.1.8 and 6.1.9 into 6.1.5 gives:

$$\underline{h}_{opt}(k) = \underline{h}_{opt}(k-1) + \underline{P}(k)\underline{x}(k)e(k)$$
(6.1.10)

Where

$$e(k) = y(k) - \underline{h}_{opt}^{T}(k-1)\underline{x}(k)$$
 (6.1.11)

 $\underline{P}(k)$  can be determined recursively without inverting  $\underline{r}_{xx}$  after each training example. this is performed using the Sherman-Morrison recursion [86]:

$$\underline{P}(k) = \underline{P}(k-1) - \frac{\underline{P}(k-1)\underline{x}(k)\underline{x}^{T}(k)\underline{P}(k-1)}{1 + \underline{x}^{T}(k)\underline{P}(k-1)\underline{x}(k)}$$
(6.1.12)

To permit a greater weighting to be given to the most recent training examples, a time windowed version of this recursion is used. The exponentially windowed RLS is used which is a simple alteration to equation (6.1.12):

$$\underline{P}(k) = \frac{1}{\lambda} \{\underline{P}(k-1) - \frac{\underline{P}(k-1)\underline{x}(k)\underline{x}^{\mathrm{T}}(k)\underline{P}(k-1)}{\lambda + x^{\mathrm{T}}(k)P(k-1)x(k)}\}$$
(6.1.13)

Where  $\lambda$  < 1 and is usually in the range 0.9 <  $\lambda$  < 1.

The exponential time window has been shown [87] to be a reasonable approximation to the intuitively optimum rectangular window. The RLS algorithm is used because it provides the fastest possible convergence to the desired results; this minimises the number of training examples required. The training needed is independent of the correlation between the individual input features, and is performed off-line so the computational complexity of the RLS algorithm does not create a problem. Continuous training is not used, so the numerical stability of the algorithm is not a problem. The same feature set forms the input to all of the combiners so the inverse autocorrelation matrix,  $\underline{P}(k)$ , used for computing the tap weights is identical for each combiner. This means that the computational overhead for introducing additional fault combiners is small compared to the overall computational load. Using the geometric features of the signal constellation as the input, the fault levels can be estimated using the outputs from the trained linear combiners for each fault.

Appendix E contains a listing of the code used to form the version of the adaptive combiner segment of the machine learning system used in Section 6.3.

The RLS algorithms detailed above are equivalent to the Kalman form of the tap weight recursion [88]. The Kalman algorithms have been further developed to produce 'fast Kalman' algorithms [89] which have a computational complexity of order N rather than  $N^2$ . However, these 'fast' forms cannot be used for this application because they assume that their input is a time shifted sequence which is not the case for the linear combiners used here. Macchi [90] and Cowan [91] give a summary and comparison of these adaptive filtering techniques, their uses (noise and echo cancellation [92], speech prediction [93], medical application [94] and numerous others) and their convergence and stability properties.

## 6.2 Performance of the Machine Learning System on a Real Radio

The machine learning system was tested on a real radio in the laboratory. The digital radio used was an 11 GHz (16 QAM) digital

radio looped back at RF. The constellation data was collected using a constellation analyser (HP3709) directly connected to the I and Q monitor points of the radio receiver demodulator. A data signal was provided by the radio's 17 stage pseudo random binary sequence (PRBS) scrambler. The machine learning system was trained on a set of four deliberately introduced quantified faults. The training was performed by: altering the settings on the pre-set potentiometers by a specific number of turns, or by adding external filters to the receiver IF section to introduce known amounts of passband asymmetry into a correctly set up radio. Four types of fault were introduced: output amplifier overdrive; phase-lock potentiometer out of adjustment; quadrature capacitor maladjustment; and an asymmetry in the receiver's IF filters. The output amplifier overdrive was measured using an external power meter; the turns of the phase-lock potentiometer and the quadrature capacitor were assessed by eye; and the bandpass filter was adjusted off-line using a separate system with different instrumentation.

After training on these faults, the system was connected to the radio with an arbitrary set of introduced faults. This was done for several sets of faults and when the adjustments recommended by the system were completed the radio was again correctly set up. The output from the machine learning system was a graphical bar display of the faults which indicated the distance of the radio under test from a correctly aligned radio. Figure 6.2.1 shows the output from the machine learning system (which has been trained on four fault types) connected to a maladjusted radio. Figure 6.2.2 shows the output of the machine learning system after adjustments have been made to remove the faults indicated in Figure 6.2.1. The adjustments



Figure 6.2.1 The output from the machine learning system connected to a maladjusted radio.



Figure 6.2.2 The output from the machine learning system connected to the radio after the indicated adjustments have been made.

recommended by the machine learning system correspond exactly to the deliberately introduced impairments.

The top bar in the display represents the (Mahalanobis) distance of the radio under test from the mean of well aligned radios which passed the BER test. These good radios are input to the system during training to indicate when a radio reaches an acceptable level of performance. The lower set of bars represent the output of the adaptive combiners and correspond to the magnitude of the faults (or the adjustments required). These results demonstrate that the machine learning system can deal with several simultaneously occurring faults and accurately determine their levels. These faults all formed tolerably linear relationships with their input feature sets, making them suitable candidates for use with the linear combiners. Fault conditions which exhibit a highly non-linear relationship with the input features would not have been so accurately detected.

# 6.3 <u>Tests of the Machine Learning System on the</u> Digital Radio Model

The machine learning system's performance was tested with the same data used to test the rule-based system in Section 5.4. The same ranges of each impairment were investigated. First, the system was tested with only one fault present at one time; these results are shown in Figure 6.3.1. Then combinations of two simultaneous impairments were tested; these results are summarised in Figure 6.3.2. Finally the performance of the machine learning system was tested

## Fault levels:

Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected Level Non-orthogonal carriers 1 Non-orthogonal carriers 0.5 Non-orthogonal carriers 3 Non-orthogonal carriers 2.5 Non-orthogonal carriers 5 Non-orthogonal carriers 5.4 Gap spacing error -9 Gap spacing error -9 Gap spacing error -6 Gap spacing error -6 Gap spacing error -3 Gap spacing error -2.5 TWT underdrive 0.7 Gap spacing error 3 Gap spacing error 3 TWT underdrive 0.5 Gap spacing error 6 Gap spacing error 6 Gap spacing error 9 Gap spacing error 9 TWT overdrive 8 TWT overdrive 8.1 TWT overdrive 6 TWT overdrive 6.6 TWT overdrive 4 TWT overdrive 4.2 TWT overdrive 2 TWT overdrive 2.1 Gap spacing error 0.4 TWT underdrive TWT underdrive 2 1.4 TWT underdrive 4 TWT underdrive 3.4 TWT underdrive 6 TWT underdrive 5.4 TWT underdrive 8 TWT underdrive 7.5

Figure 6.3.1 The output from the machine learning system for diagnosing single fault conditions

Fault levels:

Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected Level

Gap spacing error	3	Gap spacing error	3.3
Non-orthogonal carriers	3	Non-orthogonal carriers	3
TWT underdrive	4	TWT underdrive	3.4
Non-orthogonal carriers	3	Non-orthogonal carriers	2.4
TWT overdrive	8	TWT overdive	16
Non-orthogonal carriers	3	Non-orthogonal carriers	-10
TWT overdrive	2	TWT overdrive	2
Non-orthogonal carriers	3	Non-orthogonal carriers	1.9
TWT overdrive	6	TWT overdrive	2
Non-orthogonal carriers	3	Gap spacing error	10.1
TWT underdrive	4	TWT underdrive	3.3
Non-orthogonal carriers	1	Non-orthogonal	0.5
TWT overdrive	8	TWT overdrive	3.6
Non-orthogonal carriers	1	Gap spacing error	4
TWT underdrive	4	TWT underdrive	3.2
Gap spacing error	6	Gap spacing error	6
TWT overdrive	2	TWT overdrive	2.6
Gap spacing error	6	Gap spacing error	4.8
IWT overdive	6	TWT overdrive	1.8
Gap spacing error	6	Gap spacing error	16
		Non-orthogonal carriers	2
IWT overdrive	8	TWT overdrive	10.5
Gap spacing error	6	Gap spacing error	7.5
		Non-orthogonal carriers	0.8

	Faults Introduced	Level		Faults Detected	Level
TWT	underdrive	4	TWT	underdrive	3.5
Gap	spacing error	3	Gap	spacing error	1.5
TWT	overdrive	2	TWT	overdrive	1.3
Gap	spacing error	3	Gap	spacing error	1.3
TWT	overdrive	6	Non-	orthogonal carriers	-1.3
Gap	spacing error	3	Gap	spacing error	9.8
TWT	overdrive	8	TWT	overdrive	3.5
Gap	spacing error	3	Gap	spacing error	10
			Non-	orthogonal carriers	1.1

Figure 6.3.2 The output from the machine learning system for diagnosing double fault conditions.

with three faults occurring at once; these findings are detailed in Figure 6.3.3.

The machine learning system indicated the correct level of the single TWT amplifier faults to within 1 dB for an overdrive of 8 dB through to an underdrive of 8 dB. The levels of the nonorthogonality of the I and Q carriers for single faults were detected to within 1° of the actual level. The levels estimated for single spacing errors were within 0.5% of the introduced error. The differences between the output of the system and the introduced fault levels arose from errors caused by the approximations used in the selected filter tap weight algorithm.

With multiple introduced fault conditions the machine learning system's performance degraded significantly. Fault estimates of multiple impairments which included high levels of TWT overdrive proved very unreliable. Estimates were up to: 8 dB in error for the TWT overdrive level; 10% in error for the signal constellation spacing fault; and up to  $3^{\circ}$  in error for the level of the non-orthogonality of the I and Q carriers. The fault detection for the cases with TWT underdrive or only low levels of TWT amplifier overdrive, 2 dB to 4 dB, were much better except that the system always failed to detect low levels of spacing error because the training sequence included high levels of TWT overdrive. These errors in the multiple fault cases were caused by the non-linear changes in the feature set versus fault relationships for certain fault levels. The nature of the adaptive combiners, which are linear classifiers, does not permit them accurately to model these nonlinear relationships.

Fault levels: Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected Level TWT overdrive 2 TWT overdrive 3 Gap spacing error 9 Gap spacing error 7.5 Non-orthogonal carriers Non-orthogonal carriers 1 0 TWT overdrive 6 TWT overdrive 1 Gap spacing error 3 Gap spacing error 8 Non-orthogonal carriers 3 Non-orthogonal carriers 0.6 TWT overdrive 6 TWT overdrive 10 Gap spacing error 6 Gap spacing error 1.5 Non-orthogonal carriers Non-orthogonal carriers 1 0.3 TWT overdrive 8 TWT overdrive 3.7 Gap spacing error 3 Gap spacing error -1 Non-orthogonal carriers Non-orthogonal carriers 3 0.2 TWT overdrive 2 TWT overdrive 1 Gap spacing error 3 Gap spacing error 2.5 Non-orthogonal carriers 3 Non-orthogonal carriers 2

Figure 6.3.3 The output from the machine learning system for diagnosing triple fault conditions.
The machine learning system's performance is affected to a much greater extent, than that of the rule based system, by the presence of high levels of TWT overdrive. Single fault conditions are correctly diagnosed by both systems as are multiple faults ( excluding TWT overdrives of >4dB ). However, creating the rule-base for the rule-based system is a slow process, whereas, the training of the machine learning system is a relatively quick proceedure.

## 6.4 Hybrid System

The machine learning and the rule-based systems both provide solutions to the problem of fault diagnosis for digital microwave radio equipment, using the signal constellation geometric features as the information source. The geometric parameters accurately represent the positions of the signal states of a 16 QAM signal constellation, and from these features the faults present can be diagnosed. Faults which do not affect the signal constellation (faults in the slicer and decoder circuitry) are undetectable using these systems and extra information would be required to correctly diagnose them.

The rule-based system requires much work by a knowledge engineer, examining the relationships between the faults in the radio equipment and the changes in the geometric features. Once these relationships have been established the knowledge engineer has to write a set of rules (in an appropriate form for the system) which express these relationships. The number of rules required increases as the required accuracy of the diagnoses is increased. The work by the knowledge engineer is appreciable; moreover, it must be repeated

for each different type of 16 QAM digital radio that the system is used with. The rules written for one specific type of 16 QAM radio would not necessarily work when applied to a different type of 16 QAM radio.

The machine learning system also requires to be trained separately on each type of radio with which is it used. However, this training needs only a few examples of each fault condition, and this can be performed in a few minutes. The training of the machine learning system will not produce a working system if it is trained on highly non-linear fault/feature relationships. This limitation of the machine learning system restricts the variety of fault conditions that the system can correctly diagnose.

The rule-based system requires extensive work by a knowledge engineer and its accuracy is dependent upon the size of the rulebase. While the machine learning system must be limited to fault/feature relationships which are almost linear, there is little additional work required from the knowledge engineer to perform the training. These systems complement each other, with the areas of weakness of one system being the areas of strength of the other system.

Without developing any new techniques, or using any other methods than those already detailed, an improved system is possible. This would take advantage of the robustness of the rule-based system to non-linear fault/feature relationships, and the accuracy, fine tuning capabilities and ease of training of the machine learning system. A hybrid of the two systems was formulated which consisted

of a combination of the conventional rule-based approach and of the machine learning system. The overall structure of the system remained the same as that of the machine learning system, shown in Figure 6.1.1. However, the adaptive algorithm section of the system also contains some rules as well as the adaptive combiners detailed in Section 6.1. The structure of this new section, which replaces the adaptive combiner section shown in 6.1.1, is detailed in Figure 6.4.1.

The rules are used to determine the level of the fault conditions which exhibit a non-linear fault/feature relationship. If these rules establish that there is a fault corresponding to a nonlinear relationship, then the level of the fault is output and no more analysis is performed by the system. However, if only faults corresponding to linear fault/feature relationships are present, the system continues on to the adaptive algorithm and proceeds with the processing as in the unmodified machine learning system.

The combination of the two systems in this form provides several advantages. The size of the rule-base and therefore the amount of work required to be done by the knowledge engineer is less. Rules are only required to cover the faults corresponding to nonlinear relationships which are a small subset of all of the faults under examination. These rules prevent the adaptive processing section of the system having to make diagnoses from non-linear fault/feature relationships. The adaptive algorithms provide an accurate fault diagnosis of faults with linear fault/feature relationships and the rules limit the adaptive section to these cases. If there are no non-linear conditions among the faults being



Figure 6.4.1 The structure of the hybrid system.

examined, then the rule-base is left empty and the system reverts to the machine learning system. If rules are required the knowledge engineer will be required to write them, and then train the machine learning system in the conventional manner. On the one hand, the work required by the knowledge engineer is greater than if no rules were used and it was left simply as the machine learning system; on the other hand, considerably less effort is needed than that required to produce a complete rule-based system.

The benefits of this system are that first, it does not have to be limited to purely linear relationships as the machine learning system does and secondly, it can provide more accurate results than the rule-based system (without a very large number of rules and a great amount of work by the knowledge engineer). This approach does mean that for those cases where there is a non-linear fault the system will have to use two passes to fully diagnose all of the faults. On the first pass the non-linear condition will be diagnosed, and then corrected; the second pass will diagnose the remaining faults. However, this penalty is small compared to the greater range of conditions that can be treated by such a system.

# 6.5 <u>Performance of the Hybrid System used to Diagnose Faults</u> in the Radio Model

The performance of the hybrid system was examined by testing it on a variety of induced fault conditions on the radio model. The results of these tests are summarised in the table given in Figure 6.5.1. This system correctly diagnoses all of the TWT overdrive faults of 6 dB or greater without attempting to establish

Fault levels:

Non-orthogonal carrier in degrees Gap spacing as a percentage TWT overdrive in dB TWT underdrive in dB

Faults Introduced Level Faults Detected Level

Ion-orthogonal carriers	1	Non-orthogonal carriers	0.5
Ion-orthogonal carriers	3	Non-orthogonal carriers	3.5
on-orthogonal carriers	5	Non-orthogonal carriers	4.8
ap spacing error	-9	Gap spacing error	-9.5
ap spacing error	- 6	Gap spacing error	-5.9
ap spacing error	-3	Gap spacing error	-3
ap spacing error	3	Gap spacing error	3.1
ap spacing error	6	Gap spacing error	5.8
ap spacing error	9	Gap spacing error	9.8
WT overdrive	8	TWT overdrive	8
WT overdrive	6	TWT overdrive	6
WT overdrive	4	TWT overdrive	3
WT overdrive	2	TWT overdrive	1.7
WT underdrive	2	TWT underdrive	1.9
WT underdrive	4	TWT underdrive	3.1
WT underdrive	6	TWT underdrive	5
WT underdrive	8	TWT underdrive	9
WT overdrive	2	TWT overdrive	0.85
on-orthogonal carriers	1	Non-orthogonal carriers	1.4
ap spacing error	9	Gap spacing error	9
WT overdrive	2	TWT overdrive	1.5
ap spacing error	3	Gap spacing error	2
WT overdrive	2	TWT overdrive	1
ap spacing error	6	Gap spacing error	6
WT underdrive	4	TWT underdrive	3
on-orthogonal carriers	1	Non-orthogonal carriers	0.5
WT underdrive	4	TWT underdrive	3.4
on-orthogonal carriers	3	Non-orthogonal carriers	3.2
WT underdrive	4	TWT underdrive	2.9
ap spacing error	3	Gap spacing error	2.8

d Level
ed Level
2.8
5.8
8
8
8
8

6

Non-	orthogonal carriers	1			
Gap	spacing error	6			
TWT	overdrive	6	TWT	overdrive	6
Non-	orthogonal carriers	3			
TWT	overdrive	6	TWT	overdrive	6
Gap	spacing error	-3			
Non-	orthogonal carriers	3			
TWT	overdrive	6	TWT	overdrive	6
Gap	spacing error	3			
TWT	overdrive	6	TWT	overdrive	6
Gap	spacing error	6			
Gap	spacing error	-3	Gap	spacing error	-3
Non-	orthogonal carriers	3	Non-	-orthogonal carriers	2

TWT overdrive

TWT overdrive

Figure 6.5.1 Summary of the performance of the hybrid system

the levels of any of the other faults present. This method, which circumvents the problem of determining the other faults present when there is a high level of TWT overdrive, allows the number of rules needed for the system to be kept to a minimum. However, it does mean that a two step process is required for the radio to be correctly adjusted. If initially there was only a TWT overdrive fault present, then the second examination of the radio by the system will show the radio as functioning correctly.

The performance of the system for detecting the remaining fault conditions (TWT overdrive up to 6 dB, TWT underdrive, spacing errors and non-orthogonality of carriers) proved excellent. The errors in the diagnoses remained constant for detecting single or multiple occurring faults. The results indicate that a radio could quickly and reliably be returned to correct adjustment provided the only faults present were those upon which the system had been previously trained.

# CHAPTER 7

## CONCLUSIONS

7.0 Summary

Two main areas have been addressed in the work presented in this thesis. First, a model of a digital radio was produced to provide the information required to generate and evaluate several knowledge-based systems. Secondly, the model was then used to assess the applicability of three knowledge-based systems to fault diagnosis of 16 QAM digital radio relay equipment.

The unavailability throughout the course of the project of a suitable 16 QAM digital radio transmitter and receiver precluded the use of actual radio equipment. An alternative to this equipment for the purpose of evaluating the diagnostic expert systems was a digital radio model which simulated the main components of a 16 QAM radio and into which faults could be deliberately introduced.

A model of a typical 16 QAM digital radio transmitter and receiver was successfully developed using two HP 4948A analysers, one as a transmitter and one as a receiver, with a transmission rate of 2400 bits/s. Distortions were introduced in quantified amounts into three of the components of the model's transmitter. The elements into which the impairments were introduced were the 2 to 4 level coder, the carrier phase splitter and the TWT amplifier. These faults caused constellation gap spacing errors, constellation nonorthogonality, and TWT overdrive and underdrive, respectively. The model with the facility to deliberately introduce quantified distortions provided an excellent vehicle both for producing the fault/feature relationships used to generate the knowledge-bases, and for evaluating the performance of the diagnosis of specific faults by the knowledge-based systems.

Two separate approaches to implementing a diagnostic expert system (rule-based and machine learning methods) were examined and then combined to generate a third technique.

The first of these initially use a Prolog shell (KS 299) to implement the rule-based system. This method of implementing a rulebased approach was dropped, however, as the constraints it placed on the rules (no arithmetic functions could be used) were too great. Instead a second method was employed in which the rule-based system was constructed by writing a set of ordered conditional rules in C. This method allowed access to all of the functions of the C programming language, permitting the rule-base to be kept manageably small.

The relationships between the faults and the features were thoroughly examined to ascertain which features could be used to determine most accurately the faults present and their magnitudes. Producing the ordered rules was a very slow process but, when completed, the system correctly diagnosed the faults present in the radio model under test. However, the accuracy of the diagnoses by the system was dependent upon the number of rules used. To increase the accuracy of the diagnosis of the faults in the radio, therefore, the stepsize between the estimates of the distortion levels made by the rules would have to be decreased. Almost any specified precision could be achieved, but at the expense of additional work by the knowledge engineer producing more rules. A rudimentary explanatory facility, consisting of a set of rules, was provided by the rulebased system.

It would be possible to diagnose other faults in a digital radio, apart from those examined, using the rule-based system. However, these faults would have to exhibit a clear unambiguous relationship between their magnitudes and the constellation features to permit rules to be written which estimate the fault levels from the feature set.

The second approach, using the machine learning system, was based on two separate techniques: a distance classifier and an adaptive filtering algorithm. The machine learning system was trained with data from the digital radio model. It's performance proved very poor when used to diagnose faults which included TWT overdrives of 6 dB or greater. This poor performance was caused by the non-linear relationship between high levels of TWT overdrive and the feature set. When limited to fault levels which did not form such non-linear associations within the feature set (TWT overdrives < 6 dB), the accuracy of the diagnoses made by the machine learning system was excellent.

The training of the machine learning system required that a minimum of three training examples of each fault condition be given. It proved to be quick and simple to perform the training required to produce an operational system.

The machine learning system was tested to evaluate its performance when used to diagnose faults in an actual radio. It correctly diagnosed the faults which had been deliberately introduced into the radio. None of the ranges of faults examined formed highly non-linear relationships with the feature sets. This limited the machine learning system to its efficient area of operation, thus producing accurate diagnoses of the fault conditions.

As with the rule-based system, the machine learning system could diagnose faults other than those already examined provided that they formed an unambiguous relationship within the feature set. If such a relationship did not exist, then additional features would have to be found. Additional constraints set by the machine learning system were that only faults exhibiting linear fault/feature relationships could be accurately diagnosed, and no explanatory facility to give the user information about the rationale behind each individual diagnosis was provided.

Whilst the rule-based and machine learning systems both provided solutions to the problem of fault diagnosis of digital microwave radio equipment, each had its own limitations. The rulebased approach proved robust, giving dependable but crude estimates of the fault levels. The production of the rule-base, however, proved to be very labour intensive, particularly when the number of rules required was increased to achieve better accuracy of the diagnosis. The machine learning technique proved less robust, since its application was limited to faults which do not form highly nonlinear relationships with the feature set. This approach does have the benefits, however, of requiring much less labour to produce a

working system and can potentially give high levels of accuracy of fault diagnosis.

The strengths of these two techniques clearly complement each other, and this led to the choice of a hybrid system which combined both approaches. This hybrid system uses a small set of rules that diagnose those faults which exhibit strong non-linear relationships with the feature set. The machine learning system is thus partitioned from certain fault conditions to ensure its performance remains high. The work required to produce this hybrid for one particular radio type is greater than for the machine learning system, but is considerably less than for the rule-based system.

If the rule-based part of the hybrid system determines whether a TWT overdrive of 6 dB or greater exists, and if so the system will not attempt to diagnose further faults. A second diagnosis would be required to discover any remaining impairments once the TWT overdrive had been corrected. The machine learning section of the system still provides no explanations of the fault diagnoses. Other than this the performance of the hybrid system proved excellent.

This work has demonstrated that it is possible to produce a knowledge-based diagnostic system for 16 QAM digital radio relay equipment. If the faults considered are limited to those which do not form highly non-linear relationships within the feature set, then the machine learning system is the most effective solution. However, if this constraint is not possible, then the hybrid system provides an excellent alternative.

## 7.1 Further Work

The knowledge-based techniques provided valuable tools for performing the fault diagnosis of 16 QAM digital radio relay equipment. However, there is much further work required to develop and refine these methods before they could be used with actual radio equipment to perform all of their required tasks.

Before these knowledge-based techniques could be applied to an actual radio it would be necessary to ensure that the data (or information source) detailing the faults under examination adequately described all of the faults. The data used is the feature set and it must be expanded to form a more complete set which could be used to provide the required information for diagnosing a comprehensive set of fault conditions.

If an operational system were developed using the machine learning or hybrid system, then some form of explanatory facility would be required. The explanations would be used by operators when trying to justify the validity of any particular diagnosis as a check on correct system operation, and could also be used as part of a teaching aid for unskilled personnel. The most promising sources of information within the machine learning system's structure are the covariance matrix  $(\underline{r}_{XX})$  and the tap coefficient vector (<u>h</u>) which hold the information from the training examples and, if properly exploited, could form the basis for producing explanations.

The hybrid system requires a knowledge engineer to write the rules which diagnose those faults which form highly non-linear

fault/feature relationships. To minimise the effort required to do this, and to ensure that the form of the rule-set was correct to allow it to fit in properly with the rest of the system, a shell could be developed. The shell would prompt the knowledge engineer for the appropriate information and would create the rules, speeding up the process of producing systems for different radio types.

To remove the need for rules in the hybrid system, a method of enabling the machine learning system to accurately diagnose faults which formed non-linear fault/feature relationships could be developed. In those circumstances, the knowledge engineer would not be required to produce any rules, making system generation simpler. One possible approach to achieve this would use higher order approximations of the fault/feature relationships (cubic and quartic rather than only quadratic); however, a technique would be required that would establish the necessary order of the approximations since higher order approximations incur the penalty of requiring longer training sequences.

Diagnostic expert systems have greater potential value for more complicated modulation techniques than for 16 QAM because the increased complexity of these systems provides greater problems for a human expert trying to perform the equipment fault diagnosis. A more complicated modulation technique (e.g. 64 QAM, 256 QAM and others) would require that a suitable feature set be developed to form the parameters from which a knowledge-based system could estimate the impairments present. If an adequate set of parameters were created for a 64 QAM (or any other modulation technique) radio, it would be useful to investigate the performance of the hybrid system with the

64 QAM equipment. This could show whether the increased problems of fault diagnosis created by the greater complexity of the radio could be overcome with the same form of diagnostic expert system as used with a 16 QAM radio.

#### REFERENCES

- [1] Harvey, J.J. 'Expert Systems : An Introduction', Electrical Communication, Vol. 60, No. 2, 1986, pp. 100-191.
- [2] Walter, A., 'Knowledge Systems: Principles and Practice', IBM Journal of Research, Vol. 30, No. 1, January 1986, pp. 2-13.
- [3] 'An AI/Expert Systems Primer', Knowledge Systems Laboratory, Stanford University, 1985, pp. 69-75.
- [4] Stefik, M., Aikins, J., Balzer, R., Benoit, J., Birnbaum L., Hayes-Roth, F. and Sacerdoti, E., 'The Organization of Expert Systems, A Tutorial', Artificial Intelligence, Vol. 18, 1982, pp. 135-173.
- [5] Weiss, S.M. and Kulikowski, C., 'A Practical Guide to Designing Expert Systems', Chapman and Hall, London, 1983.
- [6] Barr, A. and Feigerbaum, E.A., 'The Handbook of Artificial Intelligence', Vol. I, Pitman Books, 1982.
- [7] Crawford, T.M. & Martin, V., 'A Machine Learning Approach to Expert Systems for Fault Diagnosis in Communications Equipment', IEE Computer-Aided Engineering Journal, Feb. 1987, pp. 31-38.
- [8] Rowe, M., Veitch, A., Keener, R. and Lantz, B., 'An Adaptive/Learning Diagnostic System for Complex Domains: ET' Texas Instruments Engineering Journal, January-February 1986, pp. 66-72.
- [9] Hart, A.E., 'Experience in the Use of an Inductive System in Knowledge Engineering', British Computer Society, The Building of Expert Systems, December 1984, pp. 117-126.
- [10] Feher, K., 'Digital Communications Satellite/Earth Station Engineering', Prentice Hall, New Jersey, 1983.
- [11] Feher, K., 'Digital Modulation Techniques in an Interference Environment', Don White Consultants Inc., Gainsville, Va, 1977.
- [12] Welbank, M., 'A Review of Knowledge Acquisition Techniques for Expert Systems', British Telecom Research Laboratories, December 1983.
- [13] Prade, M., 'A Computational Approach to Approximate and Plausible Reasoning with Applications to Expert Systems', IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. PAMI 7, No. 3, May 1985, pp. 260-283.
- [14] Fink, P.K., Lusth, J.S, and Duran, J.W., 'A General Expert System Design for Diagnostic Problem Solving', IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. PAMI 7, No. 5, September 1985, pp. 553-560.

- [15] 'Completed Projects' Knowledge Systems Laboratory, Stanford University 1985, pp. 63-67.
- [16] Falconer, D.D., 'Bandlimited Digital Communications Recent Trends and Applications to Voiceband Modems and Digital Radio', in Biglieri, E. and Prati,G. 'Digital Communications' Elesevier Science Publishers B.V., 1986.
- [17] Hart, G. and Steenkamp, J.A., 'Future Trends in Microwave Digital Radio: A View from Europe', IEEE Communications Magazine, Vol. 25, No. 2, Feb. 1987, pp. 49-52.
- [18] Proakis, J.G., 'Digital Communications', McGraw-Hill, New York, 1983.
- [19] Feher, K., Tetarenko, R.P., Hartmann, P.R. and Prabhu, V.K., 'Digital Communications by Radio', IEEE Trans. on Comm., Vol. Com. 27, No. 12, December 1979, pp. 1749-1750.
- [20] Pierce, A.W., Barrow, B.B., Goldberg, B. and Tucker, J.R., 'Effective Application of Forward-Acting Error-Control Coding to Multichannel HF Data Modems', IEEE Trans. on Comm., Vol. Com 18, August 1970, pp. 281-294.
- [21] Clark, G.C. and Cain, J.B., 'Error-Correction Coding for Digital Communications', Plenum Press, New York, 1982.
- [22] Tou, C.P. and Roy, D.A., 'On Efficient Spectrum Utilization From the Standpoint of Communication Theory', Proc. IEEE, Vol. 68, No. 12, December 1980, pp.1460-1465.
- [23] Amoroso, F.'The Bandwidth of Digital Data Signals', IEEE Communications Magazine, No. 18-19, November 1980, pp. 13-24.
- [24] Balshem, H., Hewitt, B., and Thaler, H.J., 'Advanced R.F. Power Amplifiers for High Capacity Digital Radio Systems', Proceedings of International Conference on Communications 1987, Paper 1.6, pp. 28-33.
- [25] Aihara, S. Nishiumi, T., Fujiki, Y. and Fukuda, S., 'GaAs FET Power Amplifiers as Substitutes for TWT Amplifiers in Multilevel QAM Digital Radio System', Proceedings of International Conference on Communications 1987, Paper 1.2, pp. 6-10.
- [26] Falconer, D.D., 'Jointly Adaptive Equalization and Carrier Recovery in Two-Dimensional Digital Communication Systems', Bell System Technical Journal, Vol. 55, No. 3, March 1976, pp. 317-334.
- [27] Eleftherion, E. and Falconer, D.D., 'Adaptive Equalization Techniques for H.F. Channels', IEEE Journal on Selected Areas in Communications, Vol. SAC 5, No. 2, February 1987, pp. 238-247.
- [28] Thomas, C.M., Weidner, M.Y. and Durrani, S.M., 'Digital Amplitude-Phase Keying with M-ary Alphabets', IEEE Trans. on Comms., Vol. Com-22, No. 2, February 1974, pp. 168-179.

- [29] Foschini, G.S., Gitlin, R.D. and Weinstein, S.B., 'On the Selection of a Two Dimensional Signal Constellation in the Presence of Phase Jitter and Gaussian Noise', Bell System Technical Journal, Vol. 52, No. 6, July-August 1973, pp. 927-965.
- [30] Feher, K., 'Advanced Digital Communications', Prentice Hall, New Jersey, 1987.
- [31] Stremler, F.G., 'Introduction to Communication Systems', Addison-Wesley, Reading, Mass., 1982.
- [32] Nyquist, H., 'Certain Topics in Telegraph Transmission Theory', Transactions, AIEE, 47(2), April 1928, pp. 617-624.
- [33] Prabhu, V.K., 'The Detection Efficiency of 16-ary QAM', Bell System Technical Journal, Vol. 59, No. 4, April 1980, pp. 639-656.
- [34] Yamamoto, H., 'Advanced 16 QAM Techniques for Digital Microwave Radio', IEEE Communications Magazine 18-19, May 1981, pp. 36-45.
- [35] Dupus, P., Joindot, M., Leclert, A. and Soufflet, D., '16 QAM Modulation for High Capacity Digital Radio Systems', IEEE Trans. on Comms., Vol. Com. 27, No. 12, December 1979, pp. 1771-1781.
- [36] Feher, K., 'Digital Communications Microwave Applications', Prentice Hall, New Jersey, 1981.
- [37] Lindsay, I.A.B. and Dripps, J.H., 'A Novel Phase-Locked Loop for PSK Carrier Recovery', Digital Processing of Signals in Communications, IERE April, 1985, IERE Conference Publication number 62, pp. 173-177.
- [38] Fukuda, E., Daido, Y., Iizuka, N., Nakamura M., and Takenaka, S., 'A 256 QAM Digital Radio System with a Low Rolloff Factor of 20 Percent for Attaining 6.75 bps', Proceedings of International Conference on Communications 1987, Paper 52.2, pp. 1798-1802.
- [39] Oetting, J.D., 'A Comparison of Modulation Techniques for Digital Radio', IEEE Trans. on Comms., Vol. Com. 27, No. 12, December 1979, pp. 1752-1762.
- [40] Gott, G.F., Dutta, S. and Doany, P., 'Analysis of H.F. Interference with Application to Digital Communications', IEE Proc., Vol. 130, Part F, No. 5, August 1983, pp. 452-458.
- [41] Stein, S., 'Fading Channel Issues in System Engineering', IEEE Journal on Selected Areas in Communications, Vol. SAC 5, No. 2, February 1987, pp. 68-89.
- [42] Gott, G.F. and Staniforth, M.J.D., 'Characteristics of Interfering Signals in Aeronautical H.F. Voice Channels', Proceedings IEE, Vol. 125, No. 11, November 1978, pp. 1208-1212.

- [43] Haworth, D.J., Pottinger, J.R. and McKissock, M.J., 'Dedicated Display Monitors Digital Radio Patterns', Hewlett Packard Journal, Vol. 38, No. 7, July 1987, pp. 4-13.
- [44] Carder, N., Dunn, W.I., Elliot, J.H., Grieve, D.W. and Rhind, G.W., 'In-service Transmission Impairment Testing of Voice Frequency Data Circuits', Hewlett-Packard Journal, October 1987, pp. 4-15.
- [45] Advanced Micro Devices, 'Bipolar Microprocessor Logic and Interface Book', 1981 Advanced Micro Devices Inc.
- [46] Saleh, A.A.M., 'Frequency-Independent and Frequency Dependent Models of TWT Amplifiers', IEEE Trans. on Comms., Vol. Com. 29, No. 11,, Nov. 1981, pp. 1715-1720.
- [47] Saleh, A.A.M., and Salz, J., 'Adaptive Linearization of Power Amplifiers in Digital Radio Systems', Bell System Technical Journal, April 1983, pp. 1019-1037.
- [48] Nau, D.S., 'Expert Computer Systems', IEEE Computer Magazine, February 1983, pp. 63-85.
- [49] Gevarter, W.B., 'Expert Systems: Limited but Powerful', IEEE Spectrum, August 1983, pp. 39-45.
- [50] Barr, A. and Feigenbaum, E.A., 'The Handbook of Artificial Intelligence', Vol. II, Pitman Books 1982.
- [51] Nii, H.P., Feigenbaum, E.A., Anton, J.J. and Rockmore, A.J., 'Signal-to-Symbol Transformation HASP/SIAP Case Study', The AI Magazine, Vol. 3, No. 2, 1982, pp. 23-35.
- [52] Carhart, R.E., 'CONGEN: An Expert System Aiding the Structural Chemist', in Michie, D., 'Expert Systems in the Microelectronic Age', Edinburgh University Press 1979, pp. 65-84.
- [53] Markosian, L.Z. and Rockmore, A.J., 'A Paradigm for Dynamic Plan Generation and Execution Monitoring', IKBS in Defense Seminar, 19 June, 1984.
- [54] Duda, R., Gasching, J. and Hart, P., 'Model Design in the PROSPECTOR Consultant System for Mineral Exploration' in Michie D., 'Expert Systems in the Microelectronic Age', Edinburgh University Press 1979, pp. 153-167.
- [55] Johnson, S.R., Connolly, J.H. and Edmonds, E.A., 'Spectrogram Analysis: A Knowledge-Based Approach to Automatic Speech Recognition', The British Computer Society, The Building of Expert Systems, December 1984, pp. 95-103.
- [56] Mitchell, T.M., Steinberg, L.I. and Shulman, J.S., 'A Knowledge-Based Approach to Design', IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. PAMI 7, No. 5, September 1985, pp. 502-510.

- [57] Cohen, P.R. and Feigenbaum, E.A., 'The Handbook of Artificial Intelligence', Vol. III, Pitman Books 1982.
- [58] Zadeh, L.A., 'Making Computers Think Like People', IEEE Spectrum, August 1984, pp. 20-32.
- [59] Eddy, W.F. and Pei, G.P., 'Structures of Rule-Based Belief Functions', IBM Journal of Research, Vol. 30, No. 1, January 1986.
- [60] Feigenbaum, E.A., 'Themes and Case Studies of Knowledge Engineering', in Michie, D., 'Expert Systems in the Microelectronic Age', Edinburgh University Press, 1979, pp. 3-25.
- [61] Stammers, R.A., 'The MXA Shell', The British Computer Society, The Building of Expert Systems, December 1984, pp. 31-37.
- [62] Hayward, S., 'Is a Decision Tree an Expert System', The British Computer Society, The Building of Expert Systems, December 1984, pp. 185-192.
- [63] Clockson, W.F. and Mellish, C.S., 'Programming in Prolog', Springer-Verlag, 1981.
- [64] Kopec, G.E., Oppenheim, A.V. and Davis, R., 'Knowledge-Based Signal Processing', Trends and Perspectives in Signal Processing, Vol. 2, No. 3, 1982.
- [65] Kopec, G.E., 'The Integrated Signal Processing System ISP', IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP 32, No. 4, August 1984, pp. 842- 851.
- [66] Gevins, A.S. and Morgan, N.H., 'Ignorance-Based Systems' Proceedings 3rd Seminar on Application of Machine Intelligence to Defence, pp. 39A5.1-39A5.4.
- [67] Francis, J.C. and Leitch, R.R., 'Artifact: A Real-time Shell for Intelligent Feedback Control' The British Computer Society, The Building of Expert Systems, December 1984, pp. 151-162.
- [68] Quincy, E.A., 'Prolog Based Expert Pattern Recognition System Shell For Technology Independant User Orientated Classification of Voice Transmission Quality', Proceedings International Conference on Communications 1987, pp. 1164-1171.
- [69] Keen, M.J.R. and Williams, G., 'Expert System Shells Come of Age', British Computer Society, The Building of Expert Systems, December 1984, pp. 13-21.
- [70] Jensen, K., and Writh, N., 'Pascal User Manual and Report', Springer-Verlag, 1978.
- [71] Keringhan, B.W. and Ritchie, D.M., 'The C Programming Language', Prentice-Hall 1978.

- [72] Winston, P.H. and Horn, B.K., 'Lisp' Addison Wesley, Reading, Mass., 1981.
- [73] Hardy, S., 'A New Software Environment for List Processing and Logic Programming' in O'Shea, T. and Eisenstadt, M., 'Artificial Intelligence. Tools, Techniques and Applications', Harper and Row, New York, pp. 110-136.
- [74] Cendrowski, J. and Bramer, M., 'Inside an Expert System: A Rational Reconstruction of the MYCIN Consultation System' in O'Shea, T. and Eisenstadt, 'Artificial Intelligence. Tools, Techniques and Applications', Harper and Row, New York 1984, pp. 453-487.
- [75] Rosenbloom, P.S., Laird, J.E., McDermott, J., Newell, A. and Orcuich, E., 'R1-Soar: An Experiment in Knowledge-Intensive Programming in a Problem-Solving Architecture', IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI 7, No. 5, September 1985, pp. 561-569.
- [76] Bramer, M.A., 'Expert Systems: The Vision and the Reality', The British Computer Society, The Building of Expert Systems, December 1984, pp. 1-12.
- [77] Martins, G.R., 'AI: The Technology That Wasn't', Defense Electronics, December 1986, pp. 56-58.
- [78] Kline, P. and Dolins, S., 'Moving from Problems to Expert System Solutions', Texas Instruments Engineering Journal, January-February 1986, pp. 50-52.
- [79] Bourne, S.R., 'The UNIX System', Addison-Wesley, 1983.
- [80] Tze, B.S., 'Pattern Recognition', Marcel Dekker 1984.
- [81] Cowan, C.F.N. and Grant, P.M., 'Adaptive Filters', Prentice-Hall 1985.
- [82] Honig, M.L. and Messerschmitt, D.G., 'Adaptive Filters Structure, Algorithms and Applications', Kluwer Academic Publishers 1984.
- [83] Ljung, L. and Soderstrom, T., 'Theory and Practice of Recursive Identification', MIT Press, Cambridge, Mass. 1983.
- [84] Goodwin, G.C. and Payne, R.L., 'Dynamic System Identification Experimental Design and Data Analysis', Academic Press 1977.
- [85] Pipes, L.A., 'Applied Mathematics for Engineers and Physicists', McGraw-Hill 1958.
- [86] Broyden, 'Basic Matrices' Macmillan Press Ltd., 1975.
- [87] Porat, B., 'Second-Order Equivalence of Rectangular and Exponential Windows in Least-Squares Estimation of Gaussian Autoregressive Processes', IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP33, No. 4, October 1985, pp. 1209-1212.

- [88] Godard, D., 'Channel Equalization Using a Kalman Filter for Fast Data Transmission', IBM Journal of Research and Development, May 1974, pp. 267-273.
- [89] Falconer, D.D. and Ljung, L., 'Application of Fast Kalman Estimation to Adaptive Equalization', IEEE Trans. on Comms., Vol. Com. 26, No. 10, October 1978, pp. 1439-1446.
- [90] Macchi, O., 'Advances in Adaptive Filtering', in Biglieri, E. and Prati, G., 'Digital Communications', Elsevier Science Publishers B.V., 1986.
- [91] Cowan, C.F.N., 'Performance Comparisons of Finite Linear Adaptive Filters', IEE Proceedings, Vol. 134, Pt. F, No. 3, June 1987, pp. 211-216.
- [92] Qureshi, S.U.H., 'Adaptive Equalisation', Proc. IEEE 1985, 73, pp. 1349-1397.
- [93] Rabiner, L.B. and Schafer, R.W., 'Digital Processing of Speech Signals', Prentice Hall, 1978.
- [94] Linkens, D.A., 'Short-time-series Spectral Analysis of Biomedial Data', IEE Proceedings, Vol. 129, Pt. A, pp. 663-672.

# Bibliography

- Brown, K.E., Cowan, C.F.N., Crawford, T.M., and Grant, P.M., 'The Application of Knowledge Based Systems for Fault Diagnosis in Microwave Radio Relay Equipment.', Proceedings of International Conference on Communications 1987, Paper 33.3, pp1159-1163. Included as Appendix F.
- 2. Brown, K.E., Cowan, C.F.N., Crawford, T.M., and Grant, P.M., 'Knowledge Based Techniques for Fault Detection in Digital Microwave Radio Communication Equipment.', To be published in IEEE Journal on Selected Areas in Communications, Special Issue on Knowledge Based Systems July 1988.

# Appendix A

The relationships between the introduced impairments and the geometric features of the 16 QAM signal constellation.



compression



expansion



rotation



Relationship of travelling wave tube backoff with non-orthogonality of the constellation









Relationship of travelling wave tube backoff with constellation rotation



ratio









pefficient



compression









Relationship of introduced spacing error with non-orthogonality of



IOIJA hutopde dos to















A poor deviation 1004 V

Relationship of introduced spacing error with O pool dowistion


% Introduced spacing error Relationship of introduced spacing error with I.Q pool variance



COLTELATION COEFILICIENC TU

Relationship of introduced spacing error with the correlation coefficient























deviation



variance



coefficient

# Appendix B

The subroutine for the HP4948A to simulate TWT distortions and the routine for calling the various elements of the model transmitter.

.

The TWT distortions subroutine.

AME I TWT DISTORTIONS

HIS ROUTINE ALTERS THE I AND 0 POINTS BY TAKING VALUES FROM LOOKUP TABLES OR AMPLITUDE AND PHASE CHANGE

MTLN	SPOIDRO / EPYLERI	
HRAX	(IÚRÚ)	
RN SEL		
MINE	RUSI OF US 2.	ACTOCE .
PICEI	BOOL SPECUSED /	STURE I
DPL V	CRAIGRA CRA CHA	
ABRU	CICCLURI CLEY, REJ	
BRACE	VIUR 1.	
MOUL		
PILIVE	BUSE, SPEWGEZ >	°STORE Q
REBEL	-	
CUP	FAIL, TOIST	
NUVE	BUSL,CB	
MUME	PM.MB	THIS IS R**2
MOVE	MB, RY, CE+	
MTLX	MB	*MULTIPLY BY BACKOFF
ACLX	MB	En Enskert
MOVE	NONE NONE	
MOVE	PM.SP(IGP1)	
CUP	FAIL,\$0800	
MTZX	(IOR1), BUSL, RY	SHIFT 7 MSB OF P++2
CUP	FHIL, TOIST	INTO BOTTOM OF PM
MOVE	BUSL CE	AS LUT REAMEN
MOVE	PM.MB	STORE THIS PRANCY
CUP	FAIL, FLUT	STORE THIS BRANCH
AC/X	(PESD / BUSL BY	
ACZX.	(PESD)	
MOVE	NONE NONE	
NOVE	PM CP	
MOVE	MB. SP( TOP1 )	ADUACE OUADAE
CUP	FATI PHMDDI	CHHOE CHHNGE
MOVE	BUSL OB	
MUVE	SPCTORIS MP	
MOVE	SPAUSIEN DV	
RKSEL	OF CHOLON, KN	
MOVE	BUSI SP. TOPON	
RUSEI	BOOL, SPATOROV	
MOVE	OP/UCITY DV	
RKSEL	SPR WSEP 2, RK	RECUVER I AND Q
MOUE	Dilet CD/ LODA	
DICE	POSC, SPATUR 19	
RESEL	Dood putter	
0.10	FHSS, FHMUD	
MOUT	FHIL, TUIST	
MOVE	BUBL, CB	
NOVE	MB, PM	"BRANCH FOR LUT
U U P A C U P	FAIL, ALUT	
HUZX	(PESD), BUSE, RY	
HUX	(PFSD)	

MOVE	SPEIGRODERY PM.CR		
MTZX	(PESD .		
ACLX	MB SP(IOP1), RY	ISCALED BY 0.5 P	FOR THE EXPANSION
MOVE	PM.SP(IOR0) (PFSD)	I POINT	
ACLX	ME		
MOVE	NUNE, NUNE		
RMOVE	PM. SP(IOR1)	O POINT	
CUS SUBEND	PASS, SCALE	SCALE FOR CORRE	OT LEVEL

END

The transmitter calling routine.

SMR		7	
	LÜFF FILE LON	& TEL 03 : KE	
	ORG CUP	≸ŮŮŮ Pass.start	
	ORG	\$20	•
HRT	CONT MOVE CUS CUS CUS CUS MOVE CUS	FHILLMENTO BUSLLMENT BUSLLMENT PASS, SPLUAD PASS, TXINIT PHSS, TFRDAD FHIL, CNPLA BUSL, CH SF, ZERD, MA PASS, CNSTCH	"set up the loop in counter B " set up DAC no. " default constellation address
	OUP MOVE	FHIL,MALD00 Busl.sp(Branch)	default branch address
LOOP	LDCT USRP	PHSS.CSTIL! Fail.cnstel	~
	CUS	TEE.OUTPUT	
	MOVE RKSEL CJS PLSEL	SP(BRANCH), PK Pass,\$0000	
	$\odot 4 \odot$	CPD, TPNSFF	×
	OUF MOVE MOVE CUS	FH(L.NSUBIT BUSL CH MH NOME B(55 MODECH,N	"test modified parameter flag "% if set read dew parameters into "MB from MA
	DEFINE	\$E00000000	
	CUS	PASS. OPPPOC	
	DEFINE	\$E0000800	
	CUP Subend	PASS.TXLOOP	

END

Appendix C

The source code for the prolog shell.

1. /u4/peter/prolog/ks299/int Original by Steven Hardy. Reworked by Peter Ross. thor: dated: 6 Sep 84 by Peter Ross rpose: a simple rule interpreter in the style of KS-300 iginal code courtesy of Teknowledge Inc. — so don't make use of it thout acknowledging them. e rule base is made up of assertions of the form RULE: if PREMISE then CONCLUSION. ere RULE is an atom, PREMISE is a simple proposition of the form THING =. VALUE or THING is known or THING is unknown or is a combination of simple propositions built up using "and" and "or", where "or" binds tighter than "and", CONCLUSION is a simple conclusion of the form THING = VALUE or THING = VALUE of CONFIDENCE or is a combination of simple conclusions built up using "and" only. ING and VALUE can be any Prolog term of precedence less than 600. operator 'of' has been defined for convenience; it has precedence 599. allows you to have THINGs of the form ATTRIBUTE of OBJECT. VFIDENCE should be a number between O (no confidence at all) and 1000 ompletely sure) inclusive. , must also provide assertions of the form QUESTION finds THING. ere QUESTION is an atom giving a question to ask the user to get a valu the attribute. A question mark will be supplied by the system. If the stem can ask the user for a value, he will be asked as soon as the need ind, and only once. Valid replies are: uhy. ing for a MYCIN-like justification in terms of the goal tree, or show THING. ing for what is known about a THING, or show RULE. ing to see the rule identified by the given tag, or : - Command. ing for some arbitrary Prolog command to be run, or thing else, which the system will assume to be the value sought, with O because you said so. re are three extra useful predicates: watch switches on printing of the recording of attribute values nowatch turns it off reads file Old and writes file New (not equal) tidy(Old, New) so that New contains a nicley laid out version of rule base in Old. (980, fx, Esought, find, invoke, seek]). (980, xfy, [concludes, uses, refersto]). (975, xfu, :). (950, fx, if). (949, xfy, then). (948, xfy, because). (800, xfy, and). (750, xfy, or). (725, xfy, cf). % cf => certainty factor (600, xfy, finds). (600, fx, show). (599, xfy, of).

```
187
nd THING :-
      prompt(Old, ' ==>> '),
      abolish(active, 1),
      abolish(sought, 1),
      abolish(because, 2),
      seek THING,
      show THING.
ek THING :-
      sought THING,
      ! .
ek THING :-
      QUESTION finds THING,
      write(QUESTION),
      write('?'), n1,
      read (REPLY),
      ( REPLY = why
          ->
              why,
               seek THING
      ; REPLY = help
              help,
          ->
              seek THING
       REPLY = show SOMETHING
      1
              show SOMETHING,
          ->
              seek THING
      ; REPLY = (:- COMMAND)
          ->
              do_without_fail(COMMAND),
              seek THING
      / assert(sought THING),
        note(THING = REPLY of 1000 because ['you said so'])
      21
      1
ek THING :-
     assert(sought THING),
      ( nonrecursive(RULE, THING)
     / recursive(RULE, THING)
      1,
     notice(RULE),
     invoke RULE,
     fail.
K THING.
without_fail(COMMAND) :-
     COMMAND,
     1
without_fail(_).
oke RULE :-
     RULE : if PREMISE then CONCLUSION,
     PREMISE of CONFIDENCE,
     ( CONFIDENCE < 200
     ; note(CONCLUSION of CONFIDENCE because [RULE])
     >,
     ! .
ice(RULE) :-
     (watching -> write('****** Invoking '), write(RULE), nl; true),
     asserta(active(RULE)).
ice(RULE) :-
     retract(active(RULE)),
     fail.
```

```
1 or P2) of CONFIDENCE :-
     P1 cf C1,
      (C1 = 1000)
          -> CONFIDENCE = C1
      1 P2 cf C2,
        ( C1 > C2 -> CONFIDENCE = C1 ; CONFIDENCE = C2 )
      >,
      ! .
1 and P2) of CONFIDENCE :-
      P1 cf C1,
      ( C1 < 200
         -> CONFIDENCE = C1
      1 P2 cf C2,
       ( C1 < C2 -> CONFIDENCE = C1 ; CONFIDENCE = C2 )
      >,
      ! .
ING = VALUE of CONFIDENCE :-
      seek THING,
      ( THING = VALUE of CONFIDENCE because REASON
      I CONFIDENCE = 0
      3,
     ! .
ING is known of CONFIDENCE :-
      seek THING,
      ( ( THING = VALUE of C because REASON, C > 200 )
         -> CONFIDENCE = 1000
      I CONFIDENCE = 0
      >,
      1 .
ING is unknown of CONFIDENCE :-
     ( THING is known of 1000
         -> CONFIDENCE = 0
      / CONFIDENCE = 1000
      11
      1.
nrecursive(RULE, THING) :-
     RULF concludes THING,
     not(RULE uses THING).
cursive(RULE, THING) :-
     RULE concludes THING,
     RULE uses THING.
LE concludes THING :-
     RULE : if PREMISE then CONCLUSION,
     CONCLUSION refersto THING.
LE uses THING :-
     RULF : if PREMISE then CONCLUSION,
     PREMISE refersto CONCLUSION.
```

```
(P1 and P2) of CONFIDENCE because REASON) :-
   note(P1 of CONFIDENCE because REASON),
   note(P2 of CONFIDENCE because REASON).
THING = (VALUE1 and VALUE2) of CONFIDENCE because REASON) :-
   note(THING = VALUE1 of CONFIDENCE because REASON),
   note(THING = VALUE2 of CONFIDENCE because REASON).
THING = (VALUE of CONFIDENCE1) of CONFIDENCE2 because REASON) :-
   note((THING = VALUE) of CONFIDENCE1 of CONFIDENCE2 because REASON).
(PROPOSITION of CONFIDENCE1) of CONFIDENCE2 because REASON) :-
   note(PROPOSITION of CONFIDENCE1 of CONFIDENCE2 because REASON).
THING = (VALUE of CONFIDENCE1) of CONFIDENCE2 because REASON) :-
   note(THING = VALUE of CONFIDENCE1 of CONFIDENCE2 because REASON).
THING is unknown of CONFIDENCE because REASON).
PROPOSITION of C1 of C2 because REASON) :-
   C3 is (C1 * C2)/1000,
   note(PROPOSITION of C3 because REASON).
PROPOSITION of C1 because [REASON1]) :-
   remove(PROPOSITION of C2 because REASON2),
   1 .
   C3 is C1 + C2 - (C1 + C2)/1000,
   add(PROPOSITION of C3 because [REASON1;REASON2]).
PROPOSITION of C1 because [REASON1]) :-
   add(PROPOSITION of C1 because [REASON1]).
/e(Item) :-
   retract(Item),
   (watching -> write('--- deleted '), write(Item), nl; true).
(tem) :-
   assert(Item),
   (watching -> write('+++ added '), write(Item), nl; true).
   listof(R,active(R), [CURRENT(OTHERS]),
   tab(8),
   write('Your answer to this question will help me determine if the'.
   nl,
   tab(16),
   write('following rule is applicable:'),
   nl,
   show CURRENT,
   ( OTH + RS = []
   I nl,
     tab(8),
     write('Other relevant rules are: '),
     write(OTHERS),
     nl
   1.
: -
   tab(8), write('When you get the prompt ==>> vaild replies are:'),
   tab(16), write('- an answer to the question'), nl,
   tab(16), write('- why.
                                      to get a justification'), nl,
   tab(16), write('- show RULE.
                                      to have that rule printed'), nl,
   tab(16), write('- show THING.
                                      to see what is known about it'),
   tab(16), write('- (:- COMMAND).
                                      to have a Prolog command run'),
```

```
and P2 refersto THING :-
     ( P1 refersto THING
     / P2 refersto THING
     >,
     ! .
or P2 refersto THING :-
     ( P1 refersto THING
     / P2 refersto THING
     >,
     1
POSITION of CONFIDENCE refersto THING :-
    PROPOSITION refersto THING,
     1
NG = VALUE refersto THING :-
    ! .
NG is STATUS refersto THING :-
    ! .
WRULE :-
    RULE : if PREMISE then CONCLUSION,
    11
    tab(8), write(RULE), write(':'), nl,
    tab(10), write('if '), pwrite(PREMISE, 16), nl,
    tab(10), write('then '), pwrite(CONCLUSION, 16), nl.
ite(P1 and P2, Indent) :-
    1.
    pwrite(P1, Indent), n1,
    tab(Indent), write('and '), pwrite(P2, Indent).
te(P, _) :-
    urite(P).
THING :-
    Q = (THING = VALUE of CONFIDENCE because R),
    listof([CONFIDENCE, G], G, GS),
    !,
    sort(GS, SGS),
    tab(8), write('This is what is known about '),
    write(THING), write(':'), nl,
    bwrite(SGS).
THING :-
    sought(THING),
    !,
    tab(8), write(THING), write(' is unknown.'), nl.
h :-
    assert(watching).
tch :-
    abolish(watching, O).
te([]).
te(EEA, B](C]) :-
    bwrite(C),
    tab(16), write(B), nl.
```

```
U(OLD, NEW) :-
     (OLD == NEW -> write('Files must differ'), nl, fail; true),
     (exists(OLD) -> true: write('First file does not exist'), nl, fai
     assert(rulenumber(1)),
     see(OLD),
     tell(NEW),
     repeat,
             read(FACT),
             tidyprocess(FACT),
     .....
     told,
     abolish(rulenumber, 1).
yprocess(end_of_file).
UPPOCESS(FACT) :-
    output(FACT),
    nl, nl,
    1.
    fail.
out(NAME : if PREMISE then CONCLUSION) :-
    retract(rulenumber(N)),
    SUCC(N, N1),
    assert(rulenumber(N1)),
    write(rule), write(N), write(':'), nl,
    tab(8), write('if
                          /), purite(PREMISE, 14), nl,
    tab(8), write('then
                         '), pwrite(CONCLUSION, 14), write('.'), nl.
ut(QUESTION finds THING) :-
    write(''''), write(GUESTION), write(''''), nl,
    tab(4), write('finds '), write(THING), write('.'), nl, nl.
ut(P) :-
    write(P), write(', '), nl.
stof/3 behaves very like bagof/3, except that the collection of
swers it comes up with will never be empty. It will fail instead.
of(X,P,Set) :-
     bagof(X, P, Set),
     1 1
     X \== [].
rt/2 is a vrsion of Hoare's "Quicksort" algorithm designed to sort
rms of the form THING=VALUE of CONFIDENCE because LIST
creasing order of CONFIDENCE. The only specific reference to this
                                                            into
nd of term occurs within the definition of lesser/2, which succeeds
ly if its first argument is 'less' than its second. So, you could
sily adapt sort/2 to many other sorting jobs.
(L. Sorted) :-
    sort(L, [], Sorted).
([X1L], RO, R) :-
    partition(L, X, L1, L2),
     sort(L2, R0, R1),
     sort(L1, [X:R1], R).
([], R, R).
```

r

ľ

```
tition(CX(L), Y, CX(L1), L2) :-
        lesser(X, Y),
       !,
       partition(L, Y, L1, L2).
rtition(CX(L], Y, L1, CX(L2)) :-
       11
       partition(L, Y, L1, L2).
rtition([], _, [], []).
sser(X = _ cf C1 because _/ X = _ cf C2 because _) :-
C1 < C2.</pre>
```

# Appendix D

The knowledge-base used with the prolog shell to perform the fault diagnoses.

what is the % expansion + compression' finds expan what is the % I + G space error' finds iccap. that is the % I space error." finds igap. what is the % G space error' finds ggap. what is the constellation rotation' finds rot. what is the differential rotation' finds drot. What is the nonorthogonality' finds nonorth. What is the inner to outer ratio' finds ratio. What is the I + Q deviation' finds iqdev. What is the I.Q pool variance' finds iqvar. What is the correlation coefficient' finds correlation. Are the I and Q gap errors different (y=>1,n=>0)' finds iq\_diff. ule1: if expan=8.5 and iggap=0.0 and rot=0.5 and nonorth=0.25 and ratio=1.0 then fault=normal cf 950. ule2: if expan=12.0 or expan=17.0 or expan=22.5 or expan=27.5 then expan\_cond=overdrive\_expan. ule3: if expan=5.5 or expan=3.5 or expan=2.5 or expan=1.5 then expan\_cond=underdrive\_expan. ule4: if drot=0.5 or drot=1.5 or drot=2.5 then drot\_cond=underdrive\_drot.

ule5: if drot=5.0 or drot=7.5 or drot=10.0 or drot=12.0 then drot\_cond=overdrive\_drot. ule6: if iqgap=1.0 or iqgap=2.0 or iqgap=3.0 or iqgap=4.0 or iqgap=5.0 or iqgap=10.5 then iqgap\_space=positive\_iqgap. ule7: if iqgap\=0.0 and iqgap\_space\=positive\_iqgap then iqgap\_space=negative\_iqgap. ule8: if nonorth=1.0 or nonorth=2.0 or nonorth=3.0 or nonorth=4.0 or nonorth=5.0 then nonorth\_cond=outofsquare. ule11: if nonorth=0.25 then nonorth\_cond=square. ule12: if expan\_cond=underdrive\_expan and drot\_cond≖underdrive\_drot then cond=twt\_underdrive. ule13: if expan\_cond=overdrive\_expan and drot\_cond=overdrive\_drot then fault=twt\_overdrive cf 750. ule14: if nonorth\_cond=outofsquare then cond=nonorth\_carrier . ule15: if iq\_diff=1 and iqgap\_space=positive\_iqgap then cond=positive\_gap\_error . ule16: if iq\_diff=1 and iqgap\_space=negative\_iqgap then cond=negative\_gap\_error . ule17: if iqdev=3600000 then fault=unknown\_spread cf 950. ule18: if cond=nonorth\_carrier and iqdev=30000 and iqdiff=0 and expan=8.5 and drot=3.5 and iqgap=0.0 then fault=nonorth\_only cf 950. ule19: if cond=positive\_gap\_error and nonorth=0.25 and expan=8.5 and drot=3.5 and iqdev=30000 then fault=positive\_gap\_only cf 950.

1.20:	
	if cond=negative_gap_error and nonorth=0.25 and expan=8.5 and drot=3.5 and iqdev=30000 then fault=negative_gap_only cf 950.
ule21:	if cond=twt_underdrive and iqdiff=0 and iqgap=0.0 and nonorth=0.25 and iqdev=30000 then fault=twt_underdrive_only cf 950.
J1e22:	if cond=twt_overdrive and cond=nonorth_carrier and iqdev=30000 then fault=twtun_nonorth cf 900.
01e23:	if cond=twt_underdrive and cond=positive_gap_error and iqdev=30000 then fault=twtun_pge cf 900.
1e24:	if cond=twt_underdrive and cond=negative_gap_error and iqdev=30000 then fault=twtun_pge cf 900.
1e25:	if cond=nonorth_carrier and cond=positive_gap_error and iqdev=30000 than fault=nonor_pge cf 900.
1e26:	if cond=nonorth_carrier and cond=negative_gap_error and iqdev=30000 then fault=nonor_nge cf 900.
le27:	if fault=twt_overdrive and cond=nonorth_carrier then fault=nonorth_carrier cf 899.
le28:	if fault=twt_overdrive and cond=positive_gap_error then fault=positive_gap cf 750.
1927:	if fault=twt_overdrive and cond=negative_gap_error then fault=negative_gap cf 750.

# Appendix E

The source code for the adaptive combiner section of the machine learning system.

```
nclude (stdio.h)
nclude <math.h>
efine sigma 1
efine landa 0.98
efine signif 0.5
efine signifi 0.5
efine iloop for(i=Q;i<=16;i++)
efine _loop for(j=0;j<=16;j++)
efine k3loop for(k=0; k<=2; k++)
itialise(p,h,s)
this function initialises the covariance matrices, the inverse
variance matrix, the tap weights and the input vector. Sigma
st be defined in the calling program if this is used elsewhere*/
uble p[17][17];
uble h[17][3];
uble s[];
t i, j, k;
000
      -
      1000
              <
              p[i][]=0.0;
      k3100p
              {
              h[i][k]=0.0; /*tap weights for all three filters*/
              3
      s[i]=0.0;
      >
oop
      p[i][i]=sigma;
      3
r(s,y,h,e)
this calculates the error between the filter output and the
sired output*/
uble s[];
uble y[];
uble h[17][3];
uble e[];
uble up;
t i, k;
loop
      -
      yp=0.0;
      iloop
              yp=yp+s[i]*h[i][k];
      e[k] = y[k] - yp; /*calculate error for each filter*/
      >
```

```
utoci(p, s, e, h)
this updates the covariance inverse and the tap weights#/
ouble (17][17])
ouble s[];
uble e[];
ouble h[17][3];
                                 ï
ouble at[17][17];
ouble m[17];
ouble n[17];
nt i, j, k;
uble fi
O. O;
loop
       •
       n[i]=0.0;
       n[i]=0.0;
       1000
               •
               pt[i][j]=0.0;
               mCi]=mCi]+sCj]*pCj]Ci];
               n[i]=n[i]+p[i][j]*s[j];
               >
      >
000
       f=f+m[i]*s[i];
1/(landa+f);
000
       £
       11000
               -{
               pt[i][j]=n[i]*f*m[j];
               p[i][j]=(p[i][j]-pt[i][j])/lamda; /*update
                           the inverse autocorrelation matrix*/
               >
      k3100p
               h[i][k]=h[i][k]+n[i]*f*e[k];
      3
Iter(p,h,s,z,y,e,input)
this first finds the number of training examples then
lls the functions to train the tapweights*/
uble p[17][17];
uble h[17][3];
uble s[];
uble z[];
uble u[];
uble e[];
LE *input;
t i, j, bi
canf(input, "%d", &i);
r(j=0; j<i; j++)
      5
      values(z, y, input);
      second(z,s);
      err(s,y,h,e);
      autoci(p, s, e, h);
      }
```

```
alues(r, y, input)
suble z[];
ouble u[];
ILE *input;
nt i, k;
or(i=0; i<=11; i++)
       fscanf(input, "%lf", &z[i]);
31000
       fscanf(input, "%1f", &y[k]);
cond(z,s)
ouble z[];
uble s[];
this routine calculates the second order features*/
15 1;
or(i=0;i<=6;i++)
      s[i]=z[i];
)r(i=7;i<=9;i++)
       s[i]=z[i+1];
r(i=10; i<=16; i++)
       -
       s[i]=s[i-10]*s[i-10];
       >
ips(h, output)
uble h[17][3];
LE *output;
t i, k;
1000
       £
       iloop
                ٠
                fprintf(output, "%lf \n", h[i][k]);
                >
      fprintf(output, "\n");
      >
in()
uble p[17][17];
uble h[17][3];
uble s[17];
uble z[15];
uble e[3];
uble y[3];
ar in_file[20];
LE #input, *output, *fopen();
ar out_file[20];
ar thr_file[20];
```

```
hile(1)
       <
       fprintf(stderr,"\n please enter input file:\n");
       scanf("%s", in_file);
       if (input = fopen(in_file, "r"))
                break;
       fprintf(stderr, "\n cannot open %s \n", in_file);
       >
hile(1)
       -{
       fprintf(stderr, "\n please enter output file:\n");
       scanf("%s", out_file);
       if (output = fopen(out_file, "w"))
                breaki
       fprintf(stderr, "\n cannot open %s \n", out_file);
       >
nitialise(p,h,s);
ilter(p, h, s, z, y, e, input);
aps(h, output);
close(input);
close(output);
```

### Appendix F

A copy of the paper entitled "The Application of Knowledge-Based Systems For Fault Diagnosis in Digital Microwave Radio Equipment", which was presented at the IEEE International Conference on Communications in June 1987.

#### THE APPLICATION OF KNOWLEDGE BASED SYSTEMS FOR FAULT DIAGNOSIS IN MICROWAVE RADIO RELAY EQUIPMENT

#### K E Brown, C F N Cowan, 1 M Crawford\* and P M Grant

The University of Edinburgh, Department of Electrical Engineering The Kings Buildings, Edinburgh, EH9 3JL \* Permanent Affiliation Hewlett Packard QTD, South Queensterry, West Lothian

his paper investigates exploitation, by "expert" s, of the information present in the signal lation of a 16 QAM radio, for fault diagnosis. To gate the practical application of expert system ques a 16 QAM radio model was developed on a tt Packard non-intrusive communications analyser. approaches to the expert system are under gation, a conventional rule based system and a new he learning system based on adaptive pattern ition techniques. Both techniques are described teir relative performance is compared using data he radio model. Typical synthesised faults are up 3 TWT overdrive and underdrive, up to 5 degrees thogonality of I and Q carriers and up to +1- 10% g error in the constellation. Further information is ed on the machine learning system's performace on radio system operating at 11GHz when feeding o itself at RF.

#### troduction

ct

There are a range of problem types with varying es of explicitness of knowledge about relationships en causes and effects. These range from those everything is known and can be expressed in the of explicit elementary logical rules to cases where the lving rules are not known and many of the variables f a continuous nature. Fault diagnosis in digital lies partway along this range, where some rules of are known, but complicated combinatorial effects so present. This leads to the requirement to find the appropriate method for performing the fault osis. Human experts use the information in the er signal constellation [1] (figure 1 gives point of to the constellation) to diagnose the faults in digital wave radios. The signal constellation also provides cellent information point which is suited to use in an rt" system. The rule based [2] and the machine ng [3] systems described here, both use the 16 QAM constellation information to perform their fault osis.

In our system 16000 samples from the constellation aken, 1000 samples for each of the signal states. sample is represented by the magnitude of the I ase) and Q (quadrature) levels of the signal. These



FIGURE 1 DIGITAL RADIO TRANSMITTER AND RECEIVER

samples are used to generate a geometric feature set which describes the signal constellation.

The geometric feature set comprised:-

- [1] % Expansion of outer points
- [2] % Expansion of inner points
- [3] %I+%Q gap spacing error
- [4] %I-%Q gap spacing error
- [5] Constellation rotation (degrees)
- [6] Differential rotation of inner to outer points (degrees)
- [7] Non-orthogonality of constellation (degrees)
- [8] Ratio of number of inner points to outer points in sample set
- [9] I pool deviation (sum of I squared)
- [10] Q pool deviation (sum of Q squared)
- [11] I.Q pool variance (sum of I\*Q)
- [12] Correlation coefficient

These features are simply calculated using the I and Q coordinates for each sample and by estimating which of the 16 QAM constellation points has been received. Figure 2 shows a 16 QAM constellation and the points taken for reference in calculation of the various geometric features. These geometric features are used as the inputs to the diagnostic systems.

### **33.2.1.** CH2424-0/87/0000-1159 \$1.00 © 1987 IEEE



GURE 2 GEOMETRY OF 16 QAM SIGNAL ONSTELLATION WITH MEASURED SIGNAL ARAMETERS

#### ule Based System

Initially a Prolog expert system shell was used. This based on KS399 by Tecknowledge Inc., forms a vard chaining system similar to the structure of IN [1]. This system was subsequently abandoned as uired too many rules to cover every possible fault tion, due to the deficiency of this shell not providing to simple arithmetic capabilities which are needed himise the number of rules.

A rule based system was then written in the C age. Figure 3 shows the overall structure of the rule system. There are four distinct sets of rules within ystem, each performing a certain task that would ally be done by the human expert. The rule based ach requires that the data fall within a specific set of te levels, or for the rules to be written to cover any ted data ranges. The features are initially processed antise each feature so rules covering a discrete range tures can be written. This quantised feature set is used directly by the rule based system.

The first set of rules takes the quantised feature set etermines the faults which could possibly be present. eature set is then processed by another set of rules to sh the actual magnitude of each of the detected The next set of rules takes the feature set and nines levels for any faults which were not covered by revious sets of rules. The confidence level of the from these rules is not as great as for the previous ets. This structure allows the system to suggest the ece of a fault, while indicating its level as zero, which e of use to an operator when the initial adjustment ted by the system does not cure the fault in the Next there is a rule set to indicate which were the hining features for establishing each fault. Finally s information is output to the operator.



204

FIGURE 3 STRUCTURE OF RULE BASED SYSTEM

The rule based technique cannot return to a previous level as a human might, when a poor decision has been made, to improve future decisions. This return by an expert forms part of the learning process, to update knowledge. In a rule based approach this would involve modifying the rules to include the new information.

The system could be altered so that when instructed by the user some rules could be invoked to alter the rule base to take account of any errors found.

#### 3. Machine Learning System

There are two levels to the machine learning system as shown in figure 4. The features are first fed into a distance classifier which uses a special form of the weighted Euclidean distance, the Mahalanobis distance [4], to distinguish between radio states, such as receiver out of lock, ball of noise and well conditioned system.

This first stage sets up an n-dimensional space, for the n features, then each radio state has examples of its feature set input as training examples so each state forms a cluster in the space. For a new feature set the Mahalanobis distance is calculated to each of the clusters and the smallest distance indicates the received state. The distance from the received features to the cluster centres also gives a measure of how well the radio is set up.

33.2.2.

the distance classifier indicates that the radio st is well conditioned, that is there are 16 distinct ates, then the features are input to the adaptive section for further analysis. If the radio is not diffioned the system outputs the state which most corresponds to the given features (out of lock or oise in figure 4) and no more processing is done.

e adaptive filtering part of the system is shown in When the system enters this mode of processing it features form a more linear relationship to the states making the linear classifiers more ate than the unconstrained distance classifier. A finite impulse response (FIR) filters [5] are set i one filter for each expected fault. The feature e input data to each filter with the tap weights for culated during the training phase using a recursive fares (RLS) algorithm [6]. This type of algorithm in to minimise the number of training examples to steady state. Each filter is trained for its in fault with a series of different fault levels and



RE 4 STRUCTURE OF MACHINE



RE 5 CONFIGURATION FOR ADAPTIVE ATORS FOR EACH MODELLED FAULT

prresponding feature sets. Once training is the feature set of the radio under investigation is d each filter output corresponds to the level of icular fault.

e machine learning system allows "learning" which based approach did not readily do. If a result is be incorrect the system can be retrained to give act output for that condition.

#### 4. Performance Assessment

The tests show the performance of each system, under a range of magnitudes for each individual fault and a variety of combinations of faults.

- The fault ranges examined were:-
- [1] Up to 8dB overdrive by 2dB steps
- [2] Up to 8dB underdrive by 2dB steps.
- [3] Non-orthogonality of carriers up to 5 degrees by 1 degree steps.
- [4] Spacing error in constellation from -10% to +10% by steps of 1%.

Figures 6, 7 and 8 show how the first three of these faults effect the constellation shape.



FIGURE 6 16 QAM CONSTELLATION WITH TWT OVERDRIVE



FIGURE 7 16 QAM CONSTELLATION WITH TWT UNDERDRIVE



GURE 8 16 QAM CONSTELLATION WITH ON-ORTHOGONAL CARRIERS

#### **Rule Based System**

Single faults were detected correctly by the rule d system for all the ranges tested. The TWT drive and underdrive is detected in 2dB steps over the range. For 8dB and 6dB overdrive a spacing error indicated but its value was defined as 0%. The nonogonal carrier faults were detected correctly over the e 0 to 5 degrees in 1 degree steps, however, for 3, 4 5 degrees a spacing error of 0% was again indicated. spacing errors were correctly detected from -10% to % in 1% steps, while a nonorthogonal carrier fault of egrees was also suggested. Thus the rule based oach correctly detected all the single faults present, for some cases indicated faults which are not present, still gives the level of that fault correctly as 0.

The detection of multiple fault conditions was not as rate as that of single fault conditions. If the TWT drive was less than 6dB the diagnosis was as accurate or the single fault case. For multiple faults with TWT drive of 6dB or greater there were some errors. The m gave the correct level of TWT overdrive and ectly indicated the presence of other faults but ided erroneous levels. This problem was caused by effect of TWT overdrive on the features which were to determine other fault levels, however, the features to establish the level of TWT overdrive were not tive to the other faults.

#### Machine Learning System

The machine learning system indicated the correct of fault for single TWT faults to within 1dB for 6dB drive to 8dB underdrive. The levels of the orthogonality were detected to within 1 degree of the al level. The detected levels for the spacing errors within 0.5% of the actual error. The differences een the system output and the actual fault levels is from errors due to the approximations used in the ted filter tap weight algorithm. The machine learning system again failed to provide sensible estimates of fault levels for multiple faults which include high TWT overdrive. Estimates can be up to 8dB in error for the overdrive level. The estimates for underdrive or low values of overdrive, 2dB to 4dB, are much better, except for some cases of spacing faults which the system could not detect. Table 1 summarises these results and provides a comparison between the systems

Fault	Error in Rule Based System	Error in Machine System
TWI	< ± 1dB	< ± 1dB
Spacing	< ± 0.5%	<=050%
Non orthogonal	< ±0.5 dcg	< ± 0.5deg
Multiple*	As for single conditions	As for single conditions
Multiple**	TWT < ± 1dB	TWT< = 8dB
	Spacing < ± 4%	Spacing< ± 6%
	Non-orthogonal < ± 3 deg	Non-orthogonal < = 3 deg

"(no TWT overdrive 6dB or greater)

\*\*(including all faults)

TABLE 1 PERFORMANCE COMPARISON OF RULE BASED AND MACHINE LEARNING SYSTEMS

#### 5. Discussion

The rule based and the machine learning system both perform well for single fault conditions, while falling down to some extent with multiple fault conditions. The rules for the rule based system are established by examining the changes in the feature set for single fault conditions and writing rules to match these changes. Then rules are written to try to account for the interactions between the determining features for each fault. This accounts for the accuracy of single fault level detection as the rules were written using data for single faults. The reduction in accuracy for multiple faults arise since the rules do not take full account of the interactions on certain features by different faults.

The machine learning system demonstrated degraded performance on some of the multiple fault conditions, especially those with 6dB or greater overdrive and spacing errors because of the nonlinearities in the changes of the feature set for these ranges.

# 6. Demonstration of Machine Learning System on an 11GHz Radio System.

The machine learning system was trained on an 11GHz digital radio (16 QAM) looped back on itself at RF. The data is collected using a constellation analyser (HP3709) directly connected to the I and Q monitor points of the radio receiver demodulator. The data signal is provided by the radio's 17 stage PRBS scrambler. The system is trained on deliberately introduced faults, such as maladjustments of a particular potentiometer or adding external filters to the IF chain to give passband assymetry. The output is given in a graphical display of faults/commands for adjustments as in Figure 9. Figure 9a is the output of the system when it has been trained on four types of fault (TWT overdrive, phase-lock out of

33.2.4.





### RE 9a OUTPUT FROM MLS NECTED TO FAULTY RADIO

------

Red's Mit conditioned

Bit imet fram radies atist month # 1001

thesaus des Bellines - 1 ants + 2 all everating 1

mana sur a set of publicant it want if the spent



RE 96 OUTPUT FROM MLS AFTER ECTION OF INDICATED FAULTS

nt, quadrature capacitor maladjusted and an try in the system filters) and has been connected io with arbitary maladjustments in these faults. b is the output after the commands have been . The overdrive is measured using an external eter, the potentiometer adjustment is assessed by the bandpass filtering can be adjusted offline lifferent system.

#### lusions

e rule based and the machine learning systems wide solutions to the problem of fault diagnosis al microwave radio. The rule based approach birical rules of thumb to make the combinatorial of the faults manageable, whereas the machine technique treats the variables as being of a us nature and uses adaptive pattern matching ins to provide a solution. The rule based system and to be applicable where there was not a brward linear relationship between the faults and alting feature set. In conclusion the machine system proved more appropriate than the rule vestem for providing optimal adjustment of the here the underlying mechanisms are too complex simple rules of thumb to be applied.

combination of the two techniques would solve the problems using the suitability of each method ecific problem type. The rule based approach over the areas with large nonlinearities while the learning system is more applicable for the more oblem regions.

#### owledgements

e funding of the studentship by the SERC and the hip by Hewlett Packard are gratefully edged.

### 9. References

- Feher K "Digital Modulation Techniques in an Interference Environment" Don White Consultants 1977
- [2] Barr A and Feigenbaum E A "The Handbook of Artificial Intelligence" Pitman Books 1982
- [3] Crawford T M and Marton V " A Machine Learning Approach to Expert Systems for Fault Diagnosis in Communications Equipment" to be published in IEE CAEJ Special Issue on Expert Systems
- [4] Sing Tze B "Pattern Recognition" Marcel Dekker 1984
- [5] Cowan C F N and Grant P M "Adaptive Filters" Prentice Hall 1985
- [6] Honig M L and Messerschmitt D G "Adaptive Filters Structure. Algorithms and Applications" Kluwer Academic Publishers 1984