Exploitation of Signal Information for Mobile Speed Estimation and Anomaly Detection

Mostafa Z. Afgani

A thesis submitted for the degree of Doctor of Philosophy.
The University of Edinburgh.
2010

Abstract

Although the primary purpose of the signal received by a mobile handset or smartphone is to enable wireless communication, the information extracted can be reused to provide a number of additional services. Two such services discussed in this thesis are: mobile speed estimation and signal anomaly detection. The proposed algorithms exploit the propagation environment specific information that is already imprinted on the received signal and therefore do not incur any additional signalling overhead. Speed estimation is useful for providing navigation and location based services in areas where global navigation satellite systems (GNSS) based devices are unusable while the proposed anomaly detection algorithms can be used to locate signal faults and aid spectrum sensing in cognitive radio systems.

The speed estimation algorithms described within this thesis require a receiver with at least two antenna elements and a wideband radio frequency (RF) signal source. The channel transfer function observed at the antenna elements are compared to yield an estimate of the device speed. The basic algorithm is a one-dimensional and unidirectional two-antenna solution. The speed of the mobile receiver is estimated from a knowledge of the fixed inter-antenna distance and the time it takes for the trailing antenna to sense similar channel conditions previously observed at the leading antenna. A by-product of the algorithm is an environment specific spatial correlation function which may be combined with theoretical models of spatial correlation to extend and improve the accuracy of the algorithm. Results obtained via computer simulations are provided.

The anomaly detection algorithms proposed in this thesis highlight unusual signal features while ignoring events that are nominal. When the test signal possesses a periodic frame structure, Kullback-Leibler divergence (KLD) analysis is employed to statistically compare successive signal frames. A method of automatically extracting the required frame period information from the signal is also provided. When the signal under test lacks a periodic frame structure, information content analysis of signal events can be used instead. Clean training data is required by this algorithm to initialise the reference event probabilities. In addition to the results obtained from extensive computer simulations, an architecture for field-programmable gate array (FPGA) based hardware implementations of the KLD based algorithm is provided. Results showing the performance of the algorithms against real test signals captured over the air are also presented.

Both sets of algorithms are simple, effective and have low computational complexity – implying that real-time implementations on platforms with limited processing power and energy are feasible. This is an important quality since location based services are expected to be an integral part of next generation cognitive radio handsets.

Declaration of originality

I hereby declare that the research recorded in this thesis and the thesis itself was composed and originated entirely by myself in the School of Engineering and Science at Jacobs University Bremen and in the School of Engineering at The University of Edinburgh.

The exceptions to the above are as follows:

- The MATLAB code used to simulate the wireless fading channel is partially based on a previous effort by Van Duc Nguyen and Birendra Ghimire
- The speed estimation algorithm based on the spatial correlation function of the channel is an extension and continuation of an earlier study by Karim Khashaba and myself in the School of Engineering and Science at Jacobs University Bremen

Mostafa Z. Afgani

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor Professor Harald Haas for his continuing guidance, support and mentoring over the years. This work would surely be poorer without his invaluable advice, motivation and insistence on quality. Thanks are also due to Dr Sinan Sinanović who has helped me tremendously with his advice, insight and constructive feedback.

I would also like to acknowledge the financial support provided by the Agilent Technologies University Relations Program and the BIS (Bremerhavener Gesellschaft für Innovationsförderung und Stadtentwicklung mbH) within the T.I.M.E. program of the State of Bremen/Bremerhaven, Germany. In particular, I would like to thank Roy Macnaughton and Peter Cain from Agilent Technologies UK Limited for their faith in the project and the support they have given me in the form of helpful practical advice, equipment and challenging test data sets.

Sharad Nagappa, Gabriel Rilling, Nick Johnson and Christine Evers are friends whom I cherish greatly for making this journey an interesting one. They are responsible for many late nights and curious conversations that I will always remember. I also thank Mrs Nicola Ferguson for her support, my fellow students and the cheerful academics at the Institute for Digital Communications.

Last but not least, I would like to offer my heartfelt thanks to Mariya Shisheva and my family for their patience, encouragement, love and understanding through the years.

Contents

List of figures

List of tables

Acronyms and abbreviations

2D	two-dimensional
ACF	autocorrelation function
AFD	average fade duration
AoA	angle of arrival
AWGN	additive white Gaussian noise
BS	base station
BT	Bluetooth
BUFG	clock buffer
CCPS	clock cycles per sample
CDF	cumulative distribution function
CE	clock enable
CMOS	complementary metal oxide semiconductor
CR	cognitive radio
CRE	channel response estimate
CWT	continuous wavelet transform
DCM	digital clock manager
DF	decimation factor
DKLD	differential KLD
DL	downlink
DPRAM	dual-port RAM
DR	dead reckoning
DSIM	digital signal interface module
DSP	digital signal processing
DVB-T	terrestrial digital video broadcasting
ECG	electrocardiogram

FCC	Federal Communications Commission
FFT	fast Fourier transform
FPDKLD	fixed-point DKLD
FPGA	field-programmable gate array
GNSS	global navigation satellite systems
GPS	global positioning system
GSM	global system for mobile communications
HDL	hardware description language
IBUFG	global clock input buffer
ICA	information content analysis
INS	inertial navigation system
I/O	input/output
ISM	industrial, scientific and medical
KLD	Kullback-Leibler divergence
KLDR	KLD ratio
L1D	$\mathscr{L}_1$ distance
LAN	local area network
LCR	level crossing rate
LoS	line-of-sight
MIMO	multiple-input multiple-output
ML	maximum likelihood
MSB	most significant bit
MT	mobile terminal
Ofcom	Office of Communications
OFDM	orthogonal frequency division multiplexing
PMF	probability mass function
PoT	power of two
RAM	random access memory
RF	radio frequency

RMS	root mean square
ROC	receiver operating characteristic
ROM	read only memory
RRFSM	relative RF signature matching
RS	resource sharing
SC	selection combining
SCF	spatial correlation function
SDR	software defined radio
SNR	signal to noise ratio
SRL16	16-bit shift register lookup table
SVR	support vector regression
TDD	time division duplex
TDMA	time division multiple access
UL	uplink
UWB	ultra-wideband
WiBro	wireless broadband
WLAN	wireless local area network
WSSUS	wide-sense stationary uncorrelated scattering

Nomenclature

f_D	Doppler frequency
$f_{D_{\max}}$	Maximum Doppler frequency
θ	Angle of arrival
λ_c	Carrier wavelength
f_c	Carrier frequency
T_C	Coherence time
B_C	Coherence bandwidth
σ_τ	RMS delay spread
ν	Rayleigh fading envelope
N_ν	Number of level crossings per second
V	Signal level
V_{RMS}	Local RMS signal level
ϑ	Signal level normalised to V_{RMS}
$\overline{T}_{\text{fade}}$	Average fade duration
$\mathbf{h}(\tau, t)$	Channel impulse response vector
L	Number of multipath components
α	Relative path gain
ϕ	Phase shift
τ	Path delay
$\delta(\cdot)$	Unit impulse function
$\mathbf{a}(\theta)$	Array response vector
Ψ	Array response component
$\mathscr{B}$	Wavenumber
D_C	Coherence distance
$\rho(d, \theta)$	Spatial correlation function
Λ	Angular spread
γ	Angular constriction
$\theta_{\max}$	Azimuthal direction of maximum fading
F_n	n^{th} complex Fourier coefficient of $p(\theta)$

$p(\theta)$	Angular distribution of multipath power
$\mathbb{A}$	Leading antenna of speed estimator device
$\mathbb{B}$	Trailing antenna of speed estimator device
$H_{\mathbb{A}}$	Channel estimate from $\mathbb{A}$
$H_{\mathbb{B}}$	Channel estimate from $\mathbb{B}$
$H_{\mathbb{A},k}$	Channel estimate from $\mathbb{A}$ at database index k
$\tilde{t}$	Time required to travel a distance of d_a
d_a	Antenna separation distance
$\lvert\vec{v}\rvert$	Speed of the device
$N_{\mathbb{A}}$	Size of channel estimate database
ℓ	Database index for $H_{\mathbb{A}}$ yielding maximum correlation with $H_{\mathbb{B}}$
$\vec{x} \star \vec{y}$	Zero lag cross-correlation between $\vec{x}$ and $\vec{y}$
$v_{\max}$	Maximum speed detectable
$v_{\min}$	Minimum speed detectable
v_{est}	Estimated speed
T_s	Symbol period
$\hat{i}$	Number of symbol periods between channel response estimates
$\hat{i}_{\max}$	Maximum channel response estimate interval
$\hat{i}_{\min}$	Minimum channel response estimate interval
ς	A constant: $d_a / (T_s\, v_{\max})$
μ	Adaptive estimation rate control parameter
ϱ	Slope of the linear interpolation function for μ
ϵ	A small number to ensure convergence of μ
$\tilde{v}_{\max}$	Maximum speed detectable with μ set to unity
$\rho(d)$	Spatial correlation function for a fixed azimuthal direction
θ_0	Azimuthal direction of travel
K	Constant representing impact of multipath shape factors
ρ	Correlation between two CREs
Δt	Time delay between CRE estimates
ρ_k	k^{th} correlation coefficient
d_k	Distance corresponding to k^{th} correlation coefficient
$v_{\text{est}}^{(\text{RRFSM})}$	Speed estimate obtained using RRFSM
Δt_k	Time to k^{th} correlation coefficient

$k_{\max}$	Index to maximum permissible correlation
$k_{\min}$	Index to minimum permissible correlation
K'	Estimate of K
e_K	Relative error in the estimate of K
$\widehat{\rho}$	Erroneous correlation coefficient
e_ρ	Error in $\widehat{\rho}$
$\rho_{\max}$	Maximum value of permissible correlation value
$\rho_{\min}$	Minimum value of permissible correlation value
N_ρ	Number of consecutive CREs acquired
$\mathbb{R}_+$	Positive real numbers
P_n	Reference data set for KLD algorithm at time n
p_n	PMF of P_n
Q_n	Test data set for KLD algorithm at time n
q_n	PMF of Q_n
$D\left(p_n \Vert q_n\right)$	KLD between PMFs p_n and q_n
D_n	Short for $D\left(p_n \Vert q_n\right)$
$I_n(x)$	Information content of event x at time n
B	Histogram bin allocation
β	Total number of histogram bins
ε	Histogram pre-load value
D_{th}	KLD threshold
T_p	Signal period
T_w	Data/PMF window size
ψ	Signal period in samples
$\lvert P_n \rvert$	Data/PMF window size in samples
I_{th}	Information content threshold
N	Information cluster size (minus one)
$\sigma_{\text{I}\langle\beta\rangle}$	Standard deviation of information content in reference data set for β bins
m	A multiplicative factor connecting I_{th} to $\sigma_{\text{I}\langle\beta\rangle}$
R_{tp}	True positive rate
R_{fd}	False discovery rate
Σ_{tp}	Total correctly detected anomalous events
Σ_{ta}	Total number of anomalous events present

Σ_{md}	Total number of anomalous events that missed detection
Σ_{fp}	Total number of false positives
R_{fp}	False positive rate
Σ_{tn}	Total number of non-anomalous events
Δ_n	DKLD adjustment term at time n
$\mathtt{C}_{\mathtt{n}}$	Frequency count at time n
E	PoT equation exponent
M	Input word size for log lookup table
F_{clk}	System clock speed
r	Resource share factor
$\mathbb{L}$	Size of $\log_2$ lookup table
ΔB	Bin width
w	Input word size in 2s-complement format
$H(t,f,x)$	Spatiotemporal channel transfer function
ω	Doppler angles

Chapter 1
Introduction

OVER the past decade mobile handsets have rapidly evolved from simple single function telecommunications devices into compact, integrated platforms capable of providing a multitude of services. Apart from simply staying connected through voice calls and text messages, it is now possible to participate in video-conference calls, work on the move through applications hosted on the Internet, read online newspapers and e-books, enjoy streaming multimedia content and navigate by means of the global positioning system (GPS); amongst other things. As a result, it is not difficult to see why the mobile handset or smartphone is increasingly becoming a central part of the modern life.

Despite the rapid proliferation of these devices and supporting infrastructure, deficiencies in existing positioning and radio spectrum management technologies impose a limit on the usefulness of smart mobile handsets. This thesis describes a set of speed estimation and signal anomaly detection algorithms that can be used to alleviate some of these problems.

Speed estimation for mobile devices

The recent trend of integrating GPS receivers into the basic mobile platform allows for location awareness and opens up a whole new realm of possibilities – of which the most obvious is navigation. Some of the other exciting new services made possible by location information are: automatic labelling of videos and photographs with geographical data (geotagging), location specific help and suggestions (e.g. find the closest train station), proximity triggered alarms (e.g. alert when within 100 m of a bank) and augmented reality (e.g. highlighting points of interest). Additionally, precise location information can also help rescue workers quickly locate victims in case of an emergency and aid cognitive radio systems by exploiting geographical databases of spectral white space [1].

Considering that people generally spend more than 85% of their time indoors over a 24 hour period [2], it is unfortunate that global navigation satellite systems (GNSS) only work when

a direct line of sight exists between the receiver and at least four orbital satellites [3]. As a result, this particular method of location determination fails indoors and in other heavily shadowed areas such as "urban canyons". At the absence of GNSS, it is still possible to obtain approximate location coordinates using algorithms that produce either absolute or relative position estimates. Methods producing absolute location coordinates typically utilise the same basic ranging principles used by GNSS – the only difference is that terrestrial beacons such as wireless local area network (WLAN) access points [4] and global system for mobile communications (GSM) radio towers [5] are used in lieu of the satellites. Use of a dedicated network of ultra-wideband (UWB) beacons has also been suggested in the literature [6]. Although such methods can yield acceptable estimates of the position, the dense network infrastructure required is not always available or cost effective to implement.

The second group of location estimation algorithms produce results relative to a last known absolute position by means of the dead reckoning (DR) principle: given successive measurements of the speed, orientation and time, the velocity profile can be integrated over the observation period to yield an estimate of the position. Methods based on DR are attractive since no external points of reference are needed to obtain the relative component of the instantaneous position estimate; the only requirements are a starting point with known coordinates and reliable measurements of the heading and speed. Coordinates of the starting point can be obtained from the on-board GPS subsystem just before it fails while accurate heading information can be extracted from a device such as the ring laser gyro [7] or the electronic compasses integrated into modern handsets. A reliable method of speed estimation is therefore the last remaining challenge faced by DR based systems for position estimation.

To address this challenge, a pair of computationally simple and accurate methods of speed estimation have been developed [8, 9]. The algorithms exploit the frequency selectivity of wideband wireless channels observed in real radio environments. In addition to a wideband broadcast source (e.g. terrestrial digital video broadcasting (DVB-T)), a receiver with at least two antenna elements is required. The basic operating principle is simple: correlation between the channel transfer functions observed at the individual array elements is used to estimate the time delay and hence the speed. In its simplest form, the first algorithm assumes that a receiver with two antenna elements is aligned in the direction of motion. Then by comparing the instantaneous transfer function observed at the trailing antenna against those previously observed at the leading antenna, an estimate of the time required to traverse the

inter-antenna distance can be obtained. Since the antenna separation distance is assumed to be fixed and known *a priori*, an estimate of the speed is subsequently acquired by simply dividing the distance by the time delay estimate. No assumptions are made regarding the channel (except that it is frequency selective) and its autocorrelation function – unlike most mobile speed estimation algorithms described in the literature.

The concept is extended by introducing an equation that describes the spatial correlation function (SCF) of the wireless channel. The SCF provides a measure of the correlation between signal envelopes separated by a given distance. Therefore, given a set of correlation coefficients, it is possible to estimate the associated distances from the inverse of the SCF. If the time delay between successive measurements of the correlation coefficients is assumed to be known, the speed can be calculated from the distance estimates obtained. This procedure forms the basis of the second speed estimation algorithm. Although the SCF is channel specific, periodic estimates of the channel specific parameters can be obtained using the aforementioned two-antenna speed estimation algorithm.

The proposed methods are simple to implement and can exploit the channel estimates already obtained as a part of the data detection process. No external infrastructure support apart from a wideband signal source is needed. Software simulations are employed in order to evaluate the performance of the proposed speed estimation algorithms against a speed profile typical of pedestrians.

Detecting anomalies in signals

As the dependence on and demand for wireless data bearing services grows, so does the need for additional bandwidth to maintain reliability. Unfortunately, it is not possible to simply scale up the bandwidth due to the fact that the radio spectrum is a very limited resource. Although the electromagnetic spectrum is (for all intents and purposes) infinite, only a small fraction of it is usable for personal wireless communications as we know it today. Traditionally, the radio spectrum has been divided into a number of usable bands by regulatory bodies such as the Federal Communications Commission (FCC) in the USA and the Office of Communications (Ofcom) in the UK. Each of the bands are then assigned for exclusive access by a particular operator or service. A notable exception is of course the set of bands known as the industrial, scientific and medical (ISM) bands where emission from unlicensed consumer electronic devices is tolerated. Although this restrictive approach to sharing the radio spec-

trum provides a certain degree of reliability to a limited user base by means of interference protection, it does not help with the spectrum scarcity problem. In fact, it is a very inefficient use of the available resources since it is extremely unlikely that all of the bands are in use at the same time at a given place.

One of the goals of cognitive radio (CR) technologies is to address this problem. The term *cognitive radio* is used to describe smart, reconfigurable wireless communications devices that are capable of automatically adjusting their operating characteristics in order to adapt to changes in the radio environment. CR systems aim to simultaneously provide better quality of service for the end user and efficient spectrum utilisation by dynamically moving the communication link from crowded or occupied bands to ones that do not appear to be in use by a primary licensed system at that instant [10, 11]. In order to carry out this task, secondary CR devices perform *spectrum sensing* – a procedure used to identify "holes" (free bands) in the spectrum and characterise the radio environment [12]. While there are a number of diverse approaches to problem [13, 14], none is perfect. Energy detection based methods [15, 16] are limited by signal to noise ratio (SNR) constraints while methods relying on cyclostationary features [17, 18] are limited by the amount of *a priori* information available regarding the signal structure of the primary system. As a result of these shortcomings, spectrum sensing cannot completely avert the risk of interference that arises from dynamic spectrum sharing. Since interference generally leads to *anomalous* signal behaviour, an additional layer of simple signal processing algorithms that can help detect and characterise that behaviour is useful.

Anomaly detection refers to the process of locating unusual and unexpected events that may exist alongside nominal samples in a dataset. It is a process that is already utilised in large number of diverse application domains. Some examples include the detection of: unauthorised access to computer systems [19], fraud in financial services [20], irregularities in vital signs such as electrocardiogram (ECG) traces [21], etc. An extensive survey of current anomaly detection techniques and application domains is provided in [22].

Aiding spectrum sensing in cognitive radio is only one of the motivations for developing algorithms that can detect unusual events in communications signals. Such methods are also useful for identifying and debugging errors in complex signal protocols and radio frequency (RF) front-ends. Since the current "state of the art" techniques for detecting anomalous events in the physical layer signals simply involve the manual capture and analysis of large volumes of data sets, algorithms that can automatically capture and extract the atypical segments of a test

signal would lead to a significant time and resource utilisation efficiency improvement. However, a review of the relevant literature reveals that there is currently a lack of algorithms that can be readily applied for the detection of anomalous events in complex, information bearing communications signal traces with a low complexity overhead. Simplicity is important to ensure that the algorithm can be implemented efficiently in real-time on hand-held devices with limited processing power and battery life.

In order to address this challenge, a pair of simple and complementary anomaly detection algorithms based on statistical models of probability distribution and information theoretic measures have been developed. The first of the pair [23–25] is suitable for signals with an underlying periodic structure such as those from a time division duplex (TDD) system. The algorithm employs windowing to compare two segments of the signal separated by the period. Probability density estimates of the events in the windows are then obtained by means of histograms and compared using Kullback-Leibler divergence (KLD). This reduces the difference between the two data sets to a single non-negative real number that can be used as a discriminator for anomaly detection. Since the data sets compared are expected to have similar statistical properties under nominal circumstances, a large enough difference indicates the presence of anomalous events. The divergence based algorithm is tested against real wireless signal traces captured over the air in both software and hardware. The software implementation is based on the MATLAB software while the hardware implementation [26] is based on the Xilinx ML402 field-programmable gate array (FPGA) platform in order to perform real-time tests on high-bandwidth signals.

The second anomaly detection algorithm [23, 27] is also applicable to signals lacking a periodic structure (e.g. from systems utilising random access channels). Clean training data is first used to establish the reference probability of individual signal events. During the testing phase, the probability of the event under test is estimated after updating the reference probability distribution and used to compute the information content associated with the event. A low probability of occurrence translates to a high information content and vice versa. Since anomalous events are typically rare, the probability of occurrence for such events is low and hence the associated information content is high. Therefore, analysis of the information content can be used to detect the presence of anomalous signal events. A MATLAB based implementation is used evaluate the performance of the algorithm.

Thesis layout

A review of the existing literature on methods of mobile speed estimation and anomaly detection is given in Chapter 2 along with a brief description of the characteristics of a multipath fading mobile wireless channel. For each algorithm reviewed, a description of the core operating principle is followed by its strengths and weaknesses. The deficiencies identified in the literature are listed and set the goals for the research work presented in this thesis.

The proposed algorithms for mobile speed estimation are presented in Chapter 3. The wireless channel model is reintroduced due to its role in the design of the algorithms. The two-antenna approach is described first, followed by the algorithm based on the SCF of the channel. A simple model of pedestrian movement is used to test software implementations of the algorithms and the results are presented along with some key insights regarding the methods.

Chapter 4 follows with a description of the proposed signal anomaly detection algorithms. The chapter starts with definition of the two detector metrics (i.e. KLD and information content) and their properties. Since probability mass functions (PMFs) are at the heart of both algorithms, the use of histograms for PMF estimation is discussed next. Then the anomaly detection algorithms are described along with analysis of the computational complexity. Software implementations of the algorithms are tested against a set of real wireless signals recorded over the air and the results are presented. The detector performance with respect to various configurable system parameters is also evaluated and presented.

Implementation of the KLD based anomaly detection algorithm in FPGA hardware is described in Chapter 5. The first part of the chapter is dedicated to optimising and reducing the computational complexity of the algorithm. A method for efficient implementations of histograms and probability mass functions is also provided. An in-depth description of the architecture developed for FPGA based implementations of the algorithm is given next. A Xilinx ML402 platform is used to test the implementation against high-bandwidth signals in real-time and the chapter concludes with discussion of the results obtained.

Chapter 6 highlights the contributions made and provides a discussion on the limitations of the work presented. This chapter, and the thesis, ends with a section on the scope and directions for future research.

A listing of the relevant publications can be found on page 151.

Chapter 2
Background research

MOBILE speed estimation and anomaly detection are both rich fields of research with a correspondingly large body of relevant literature. Critical reviews of a number of papers from each field – representing methods commonly encountered in the relevant literature – are presented in this chapter. The goal is to provide a sense of the typical methodologies and their shortcomings with respect to the applications outlined in Chapter 1. A survey of the literature on speed estimation techniques is presented in Section 2.1.2 while existing algorithms for anomaly detection are reviewed in Section 2.2. A brief description of the wireless fading channel is also provided in Section 2.1.1 due to its central role in speed estimation algorithms.

2.1 Mobile speed estimation

2.1.1 The spatiotemporal wireless channel model

Multipath propagation is a common phenomenon in terrestrial environments where obstructions lead to reflections in the wireless channel. As a result of the reflections, the receiver senses multiple copies of the transmitted signal – each copy arriving after a delay corresponding to the length of the path traversed. A plot of path gains against path delays is known as the power delay profile of the multipath channel. Since the path delays are independent and typically unique, the signal copies combine constructively and destructively at the receiver – leading to signal amplitude and phase fluctuations (fading) that depend on time and space. For a static channel the variations are only along the space dimension; however, if the receiver is mobile relative to the transmitter and/or there exists mobile scatterers in the channel, the signal fades in both time and space due to Doppler shift. A typical example of the fading experienced by a receiver (474 MHz signal carrier frequency) travelling at a speed of 22 m/s (i.e. 80 km/h) relative to the transmitter is shown in Figure 2.1.

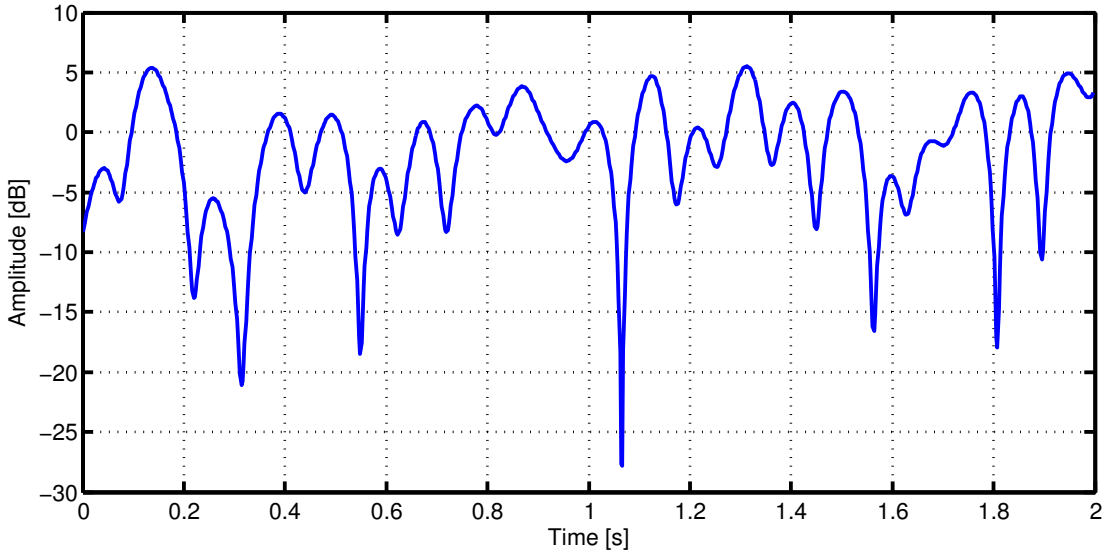


Figure 2.1: *Simulated multipath fading for a constant envelope signal with a carrier frequency of* 474 *MHz and a speed of* 22 *m/s. The maximum Doppler shift is* 35 *Hz and hence the coherence time of the channel is approximately* 5 *ms.*

Coherence time and bandwidth

The frequency of the fades observed by the receiver is directly related to its speed v and hence the Doppler shift f_{D_l} in each signal path l:

$$f_{D_l} = \cos(\theta_l)\frac{v}{\lambda_c}, \tag{2.1}$$

where θ_l is the angle of arrival (AoA) of the path and λ_c is the carrier wavelength. Assuming that the scatterers in the channel are static, the maximum Doppler shift, $f_{D_{\max}}$, experienced by the receiver is at an AoA of 0 or 2π degrees; therefore

$$f_{D_{\max}} = \frac{v}{\lambda_c}. \tag{2.2}$$

A higher Doppler frequency corresponds to a higher signal fade frequency and hence a shorter channel coherence time (defined as a correlation level above 0.5) [28]:

$$T_C \approx \frac{9}{16\pi f_{D_{\max}}}. \tag{2.3}$$

Using this definition, the coherence time of the channel shown in Figure 2.1 is approximately 5 ms. The frequency domain counterpart of T_C is the coherence bandwidth, B_C. It depends

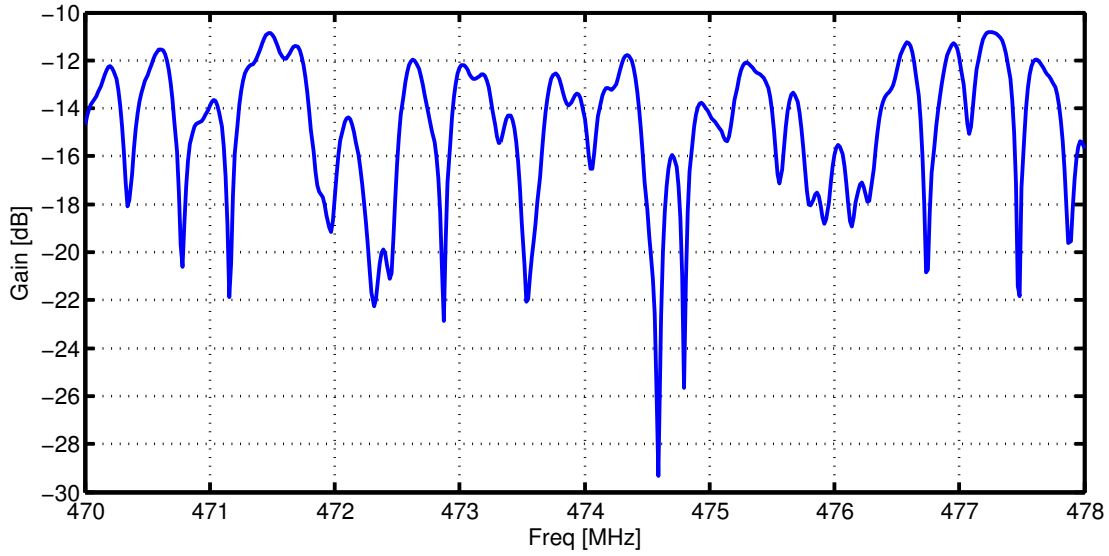


Figure 2.2: *Frequency selective fading in a* 8 *MHz wide channel with a center frequency of* 474 *MHz. The RMS delay spread,* σ_τ*, is* 1.88 μs *and hence the coherence bandwidth is approximately* 106 *kHz.*

on the root mean square (RMS) delay spread σ_τ of the channel [28]:

$$B_C \approx \frac{1}{5\sigma_\tau}. \tag{2.4}$$

The relationship implies that channels with large RMS delay spreads have smaller coherence bandwidths. When wideband wireless systems such as terrestrial digital video broadcasting (DVB-T) attempt to use a channel with a low B_C, the channel is perceived to be *frequency selective*. An example of such a channel for a σ_τ of 1.88 μs is provided in Figure 2.2. The coherence bandwidth of the channel in this case is approximately 106 kHz.

Level crossing rate and average fade duration

For a Rayleigh fading channel, level crossing rate (LCR) and average fade duration (AFD) are a pair of important statistical measures of the envelope fading process. The LCR describes the rate with which the fading envelope (normalised to the local RMS signal level) is expected to cross a given signal level in the positive-going direction. The number of crossings per second, N_ν, is given by [28]

$$N_\nu = \int_0^\infty \dot{\nu} p(\nu = V, \dot{\nu})\, d\dot{\nu} = \sqrt{2\pi} f_{D_{\max}} \vartheta e^{-\vartheta^2} \tag{2.5}$$

where $\dot{\nu}$ the first derivative of the envelope $\nu(t)$, $p(\nu = V, \dot{\nu})$ is the joint density function of ν and $\dot{\nu}$ at $\nu = V$ and $\vartheta = V/V_{\text{RMS}}$ is the specified level normalised to the RMS value of the local signal level. The level crossing rate clearly depends on the speed since the function incorporates the maximum Doppler frequency observed, $f_{D_{\max}}$. The maximum number of level crossings per second are observed when $\vartheta = 1/\sqrt{2}$; i.e. when the specified level is 3 dB below V_{RMS} [28].

While the LCR describes the frequency of fades, AFD provides a measure of the average period of time for which the received signal envelope is below some signal level V. For a Rayleigh fading signal envelope, the expression for AFD, $\overline{T}_{\text{fade}}$, is [28]

$$\overline{T}_{\text{fade}} = \frac{e^{\vartheta^2} - 1}{\vartheta f_{D_{\max}} \sqrt{2\pi}}. \tag{2.6}$$

As expected, the AFD also depends on $f_{D_{\max}}$ and hence the speed.

Spatiotemporal channel impulse response

For a receiver array with multiple antenna elements, the impulse response vector $\mathbf{h}(\tau, t)$ of a time varying multipath channel can be expressed as [29, 30]:

$$\mathbf{h}(\tau, t) = \sum_{l=0}^{L-1} \alpha_l \exp\left[j\left(\phi_l + 2\pi f_{D_l} t\right)\right] \mathbf{a}\left(\theta_l\right) \delta\left(\tau - \tau_l\right) \tag{2.7}$$

where L is the total number of multipath components. For the l^{th} path, α_l is the relative gain, ϕ_l is the carrier phase shift, θ_l is the AoA and τ_l is the time delay. $\delta(\cdot)$ is the unit impulse function and $\mathbf{a}\left(\theta_l\right)$ is known as the array response vector. When the signal and the antenna array (containing m antennae) at the receiver are restricted to a two-dimensional (2D) space, the array response vector is given by

$$\mathbf{a}\left(\theta_l\right) = \begin{bmatrix} \exp\left(-j\Psi_{l,1}\right) \\ \exp\left(-j\Psi_{l,2}\right) \\ \exp\left(-j\Psi_{l,3}\right) \\ \cdots \\ \exp\left(-j\Psi_{l,m}\right) \end{bmatrix} \tag{2.8}$$

where $\Psi_{l,i} = \left[x_i \cos\left(\theta_l\right) + y_i \sin\left(\theta_l\right)\right] \mathscr{B}$. The spatial coordinate pair $\left(x_i, y_i\right)$ represents the location of antenna element i in the 2D plane. $\mathscr{B} = \frac{2\pi}{\lambda_c}$ is known as the wavenumber [30]. Although not explicitly indicated by the equation, L, α, ϕ, f_D, θ and τ are all generally time-varying parameters.

For line-of-sight (LoS) channels, the relative strength of each component can be modeled by a Rician random variable, with the Rician parameter representing the relative strength of the LoS component. A Rayleigh distributed random variable is used instead when there is no single dominant multipath component. This is usually the case indoors or in other shadowed environments where there is no LoS path between the transmitter and the receiver. The phase shift is often assumed to be uniformly distributed within the interval $[0, 2\pi]$. The AoA is highly dependent on the relative geometry of the environment and the heights of the scatterers and the receiver. Although the assumption that the AoA is uniformly distributed within the interval $[0, 2\pi]$ is valid under certain circumstances, there are other distributions that are more suitable for a given channel model. An overview is provided in [30]. The delay associated with each multipath component is generally assumed to be exponentially distributed [29].

The effect of a wireless spatiotemporal Rayleigh fading multipath channel realised using (2.7) on a constant signal envelope is illustrated using Figure 2.3, Figure 2.4 and Figure 2.5. The channel is 8 MHz wide with a carrier frequency, f_c, of 474 MHz – corresponding to the first channel of the DVB-T service. The relative gain (α), phase shift (ϕ) and delay (τ) values for each of the 20 multipath components are obtained from the DVB-T physical layer standards specification document [31]. Since the AoA values (θ) are not specified, it is assumed to be a random variable uniformly distributed in $[0, 2\pi]$ for array response calculations. The Doppler shift for each path is also random and calculated using (2.1) where the angle is also uniformly distributed in $[0, 2\pi]$.

Figure 2.3 shows that the channel is indeed time varying and frequency selective (at a fixed point in space). The coherence time is approximately 4.5 ms and the coherence bandwidth is approximately 106 kHz. Behaviour of the same channel in the space-frequency plane at a fixed point in time is shown in Figure 2.4. The plot indicates that the channel is selective in both frequency and space. Finally, Figure 2.5 shows the spatiotemporal channel observed at a single signal frequency. It confirms that the channel described by (2.7) is indeed jointly time varying and spatially selective.

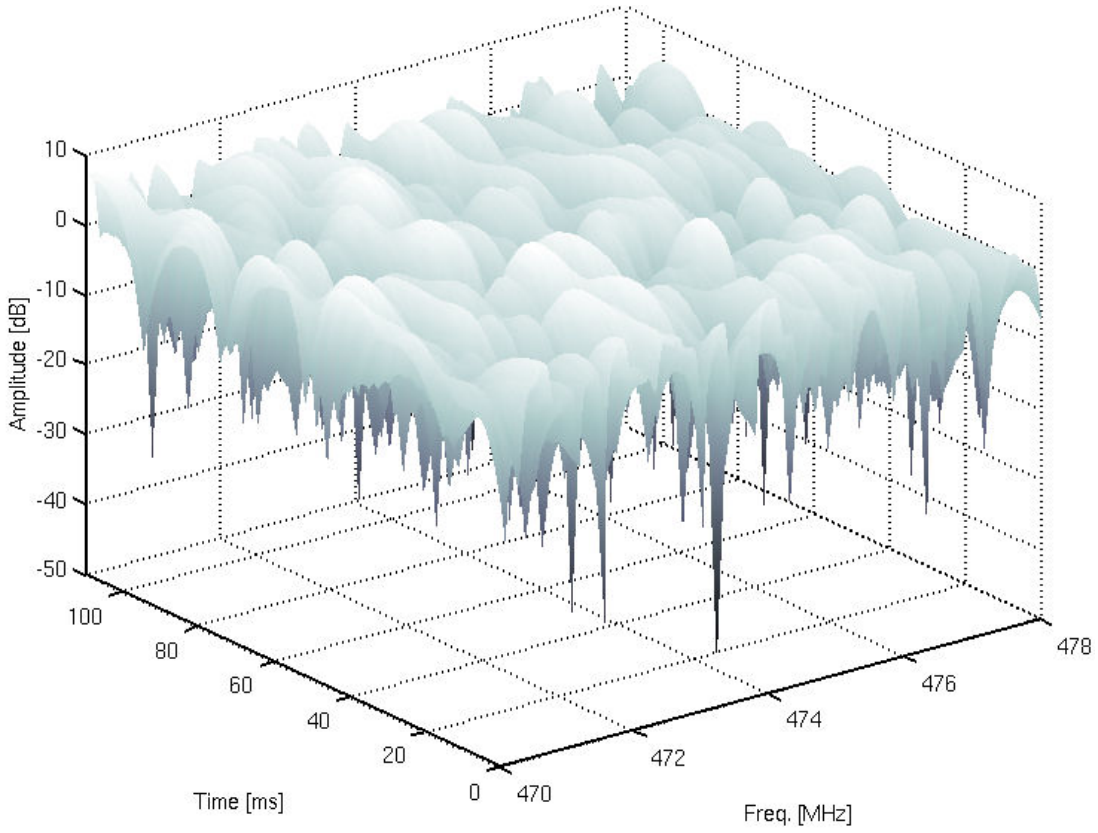


Figure 2.3: *Time-frequency response of a simulated* 8 *MHz wide Rayleigh fading channel with* 20 *multipath components. The center frequency of the channel is* 474*MHz and the speed of the receiver relative to the transmitter is* 25 *m/s. The time-varying and frequency selective nature of the channel is clearly visible.*

Spatial correlation

Quantities that provide a sense of the degree of correlation between successive channel states in the time and frequency domain (T_C and B_C) have been described previously using (2.3) and (2.4) respectively. A similar quantity, D_C, can also be defined for the spatial dimension. For a static omnidirectional Rayleigh fading channel, it is simply [32]

$$\begin{aligned} D_C &= vT_C \\ &\approx \frac{9\lambda_c}{16\pi}. \end{aligned} \tag{2.9}$$

By this definition, the coherence distance of the channel shown in Figure 2.5 is approximately 0.11 m.

A general description of the spatial correlation function in terms of multipath shape factors

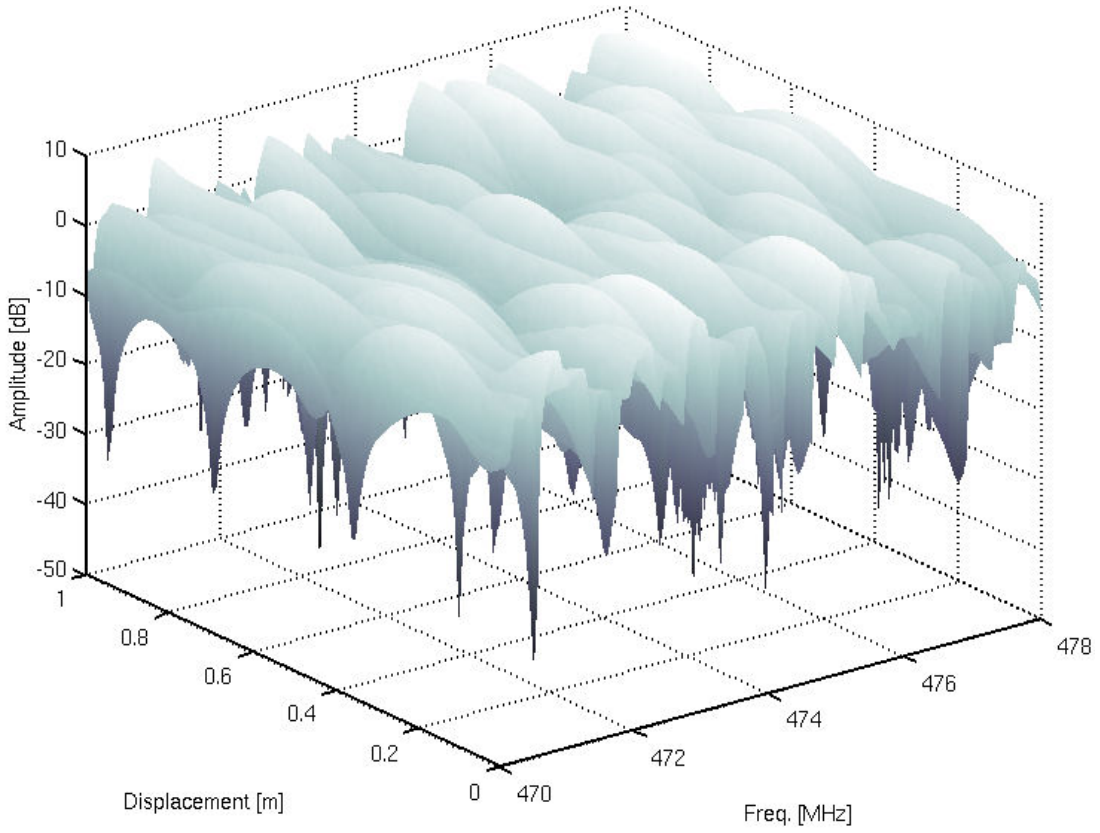


Figure 2.4: *Space-frequency response of a simulated* 8 *MHz wide Rayleigh fading channel with* 20 *multipath components. The center frequency of the channel is* 474*MHz and the displacement of the receiver is along a single spatial dimension only. Spatial variations in the channel response are observed in addition to frequency selectivity.*

is also provided in [32] and [28]. The spatial correlation, $\rho(d,\theta)$, between signal envelopes separated by a small distance d, along some azimuthal direction θ is given by

$$\rho(d,\theta) \approx \exp\left[-23\Lambda^2\left(1+\gamma\cos\left[2\left(\theta-\theta_{\max}\right)\right]\right)\left(\frac{d}{\lambda_c}\right)^2\right] \tag{2.10}$$

where Λ , γ and $\theta_{\max}$ are the multipath shape factors known as *angular spread*, *angular constriction* and *azimuthal direction of maximum fading* respectively. The expression in (2.10) is a Gaussian approximation of the true spatial autocovariance function of the received envelope

$$\rho(d) = \frac{E\left\{\nu\left(\overrightarrow{d_0}\right)\nu\left(\overrightarrow{d_0}+d\hat{d}\right)\right\}-\left(E\left\{\nu\right\}\right)^2}{E\left\{\nu^2\right\}-\left(E\left\{\nu\right\}\right)^2} \tag{2.11}$$

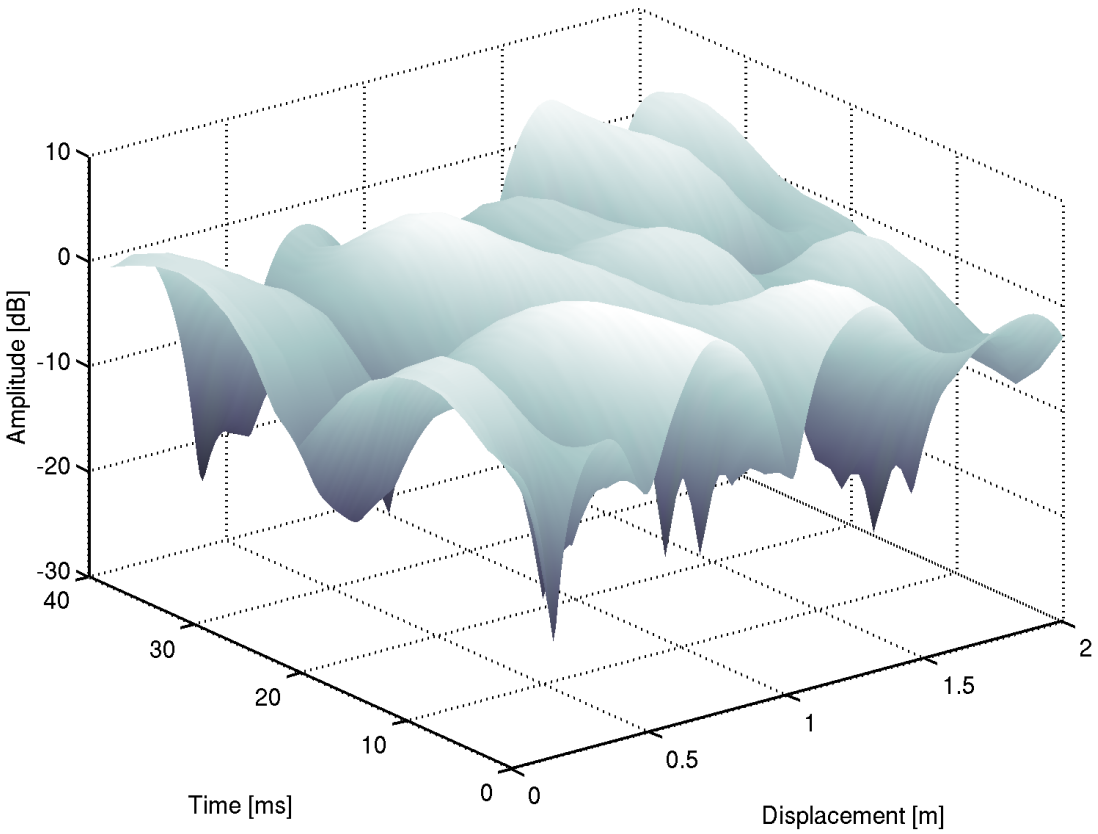


Figure 2.5: *Spatiotemporal response of a simulated* 8 *MHz wide Rayleigh fading channel with* 20 *multipath components. The center frequency of the channel is* 474*MHz and the speed of the receiver relative to the transmitter is* 25 *m/s. As expected, the channel exhibits joint spatial and temporal selectivity.*

where $\overrightarrow{d_0}$ is a position in the azimuthal plane (arbitrary for wide sense stationary fading processes) and $\hat{d}$ is a unit vector in the direction of travel. Durgin and Rappaport [32, Appendix II] derive the expression in (2.10) by performing the Maclaurin series expansion of (2.11) and setting the second term equal to the analogous term in a Maclaurin series expansion of a generic Gaussian function. Consequently, the spatial correlation function described by (2.10) is only valid for small values of d.

The angular spread shape factor describes the distribution of multipath components about a single azimuthal direction. It is defined as

$$\Lambda = \sqrt{1 - \frac{|F_1|^2}{F_0^2}}, \tag{2.12}$$

where F_n is the n^{th} complex Fourier coefficient of the angular distribution of multipath power, $p(\theta)$:

$$F_n = \int_0^{2\pi} p(\theta) \exp(-jn\theta)\, d\theta. \tag{2.13}$$

Λ ranges from zero to one, with zero corresponding the case where a single multipath component arrives from a single direction. When the power distribution lacks a dominant azimuthal direction, Λ is one.

Angular constriction is a concept that is similar to angular spread. It describes the distribution of multipath components about two azimuthal directions instead of just one. It is defined as

$$\gamma = \frac{|F_0 F_2 - F_1^2|}{F_0^2 - |F_1|^2}. \tag{2.14}$$

Once again, γ varies between zero and one. It is unity when there are exactly two multipath components arriving from two different directions. γ is zero when there is no clear bias in the direction of arrival of the multipath components.

The azimuthal direction of maximum fading is simply defined as

$$\theta_{\max} = \frac{1}{2} \arg\left(F_0 F_2 - F_1^2\right). \tag{2.15}$$

Durgin and Rappaport [32, Appendix I] obtain the multipath shape factors by means of a complex Fourier expansion of the second moment of the envelope fading process. Only the first three coefficients are relevant since it is stated that no harmonic content with respect to θ exists for $n > 2$. This is the reason why only F_0, F_1 and F_2 appear in the expressions for the shape factors.

If the coherence distance, D_C, is defined as the distance at which the spatial correlation coefficient drops to 0.5, i.e. $\rho(D_C) = 0.5$, (2.10) can be manipulated to obtain an estimate in terms of the multipath shape factors:

$$D_C \approx \frac{\lambda_c \sqrt{\ln 2}}{\Lambda \sqrt{23(1 + \gamma \cos[2(\theta - \theta_{\max})])}}. \tag{2.16}$$

For the classical omnidirectional Rayleigh fading channel [28], $\Lambda = 1$ and $\gamma = 0$. Using these

values in (2.16) results in:

$$D_C \approx \frac{\lambda_c \sqrt{\ln 2}}{\sqrt{23}} \approx 0.174 \lambda_c. \tag{2.17}$$

This expression, obtained by means of the spatial correlation function (2.10), is consistent with the definition given previously in (2.9) since

$$\frac{9\lambda_c}{16\pi} \approx 0.179 \lambda_c. \tag{2.18}$$

2.1.2 Existing methods of mobile speed estimation

It has been stated previously in Section 2.1.1 that the movement of a wireless receiver unit with respect to a transmitter results in a Doppler shift in the frequency of the signal. The most obvious manifestation of this phenomenon is the fast fading of the signal envelope at the receiver – as demonstrated in Figure 2.1. Most of the algorithms described in literature therefore attempt to estimate the amount of Doppler frequency shift in the signal and hence obtain an estimate of the speed of the receiver.

Majority of the existing algorithms for Doppler (i.e. speed) estimation utilise one of two methods: autocorrelation/covariance of the signal [33–38] and level crossing rate [39–42]. A brief comment on the utility of the latter for speed estimation was made previously in Section 2.1.1. A critical review of some current techniques for speed estimation found in the literature follows.

One of the first methods of extracting the maximum Doppler frequency, and hence the speed, from the received radio frequency (RF) signal envelope by means of its covariance function is described by Sampath and Holtzman in [33]. When the envelope process is assumed to be Rician, the estimator for $f_{D_{\max}}$, $\hat{f}_{D_{\max}}$ is defined as

$$\hat{f}_{D_{\max}} = \frac{1}{2\pi\tau} \sqrt{\frac{E(V)}{\mathrm{Var}\left(x_i^2\right)}} \tag{2.19}$$

where τ is the sampling interval, $\mathrm{Var}\left(x_i^2\right)$ is the variance of the squared signal envelope at time i and $E(V)$ is

$$E(V) = 2\left[\mathrm{Var}\left(x_i^2\right) - \mathrm{Cov}\left(x_{i+1}^2, x_i^2\right)\right]. \tag{2.20}$$

$\mathrm{Cov}(\cdot)$ is the covariance function. When additive white Gaussian noise (AWGN) is present

in the channel, the covariance function is given as

$$\begin{aligned}\mathrm{Cov}\left(x_{i+1}^2, x_i^2\right) &= 4\sigma^4 J_0^2\left(2\pi f_{D_{\max}}\tau\right) + 4\alpha^2\sigma^2 J_0\left(2\pi f_{D_{\max}}\tau\right)\cos(\omega_0\tau) \\ &\quad + N_0^2\frac{\sin^2(\pi B\tau)}{(\pi\tau)^2} + 2\alpha^2 N_0\cos(\omega_0\tau)\frac{\sin(\pi B\tau)}{\pi\tau} \\ &\quad + 4\sigma^2 N_0 J_0\left(2\pi f_{D_{\max}}\tau\right)\frac{\sin(\pi B\tau)}{\pi\tau}\end{aligned} \tag{2.21}$$

where σ^2 is the average signal power, $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind, α is the amplitude of the specular component, $\omega_0 = 2\pi f_{D_{\max}}\cos(\theta)$ where θ is the AoA of the specular component, $N_0/2$ is the double sided noise power spectral density and finally B is the bandwidth. Similarly, the variance is described as

$$\mathrm{Var}\left(x_i^2\right) = 4\sigma^4 + 4\alpha^2\sigma^2 + N_0^2B^2 + 2\alpha^2 N_0 B + 4\sigma^2 N_0 B. \tag{2.22}$$

It is therefore clear from (2.21) and (2.22) that the performance of the estimator depends on the signal to noise ratio (SNR) and the strength and AoA of the LoS component. This is supported by the results shown in the paper which indicate that uncertainties in the value of the Rician parameter and the AoA of the specular component can lead to significant errors (larger than 25%) in the Doppler frequency estimates and hence the speed. Such error levels are clearly unacceptable for positioning applications since accuracy would be lost rapidly. It can also be seen from the results that a very high SNR (greater than 40 dB for a bandwidth of 1 kHz) is needed to provide reliable estimates of the Doppler frequency; such SNR levels are unlikely to be available in reality.

A variation on the above is proposed by Sheng and Bolstein in [34]. The key contribution is a formulation of the algorithm that does not require the SNR to be known and can be applied to the in-phase and quadrature components of the signal in addition to samples from the modulated envelope. For a Rayleigh fading channel, the autocorrelation function (ACF) of the in-phase component at time index k is given as

$$\begin{aligned}\phi_k &= E\left[x_I(i+k)\,x_I(i)\right] \\ &= \sigma^2 J_0\left(2\pi f_{D_{\max}}k\right) + \sigma_n^2\mathrm{sinc}\left(kB_nT_s\right)\end{aligned} \tag{2.23}$$

where $x_I(i)$ is the i^{th} sample of the in-phase component, σ^2 is the average signal power, $\sigma_n^2 = 0.5N_0B_n$ is the in-band noise power, $N_0/2$ is the double sided AWGN power spectral

density, B_n is the bandwidth and T_s is the sampling interval. Two constants in terms of the ACF are also defined:

$$\begin{aligned} c_1 &= \frac{\phi_1}{\phi_0} = \frac{\Gamma J_0\left(2\pi f_{D_{\max}}\right) + \operatorname{sinc}\left(B_n T_s\right)}{1+\Gamma} \\ c_2 &= \frac{\phi_2}{\phi_0} = \frac{\Gamma J_0\left(4\pi f_{D_{\max}}\right) + \operatorname{sinc}\left(2B_n T_s\right)}{1+\Gamma} \end{aligned} \tag{2.24}$$

where $\Gamma = \frac{\sigma^2}{\sigma_n^2}$ is the SNR. Therefore, if the SNR is known, the Doppler frequency and hence the speed can be estimated from (2.23) and (2.24). If however, knowledge of the SNR is unavailable, $f_{D_{\max}}$ can still be estimated by solving

$$c_2 \operatorname{sinc}\left(B_n T_s\right) - c_1 \operatorname{sinc}\left(2B_n T_s\right) = c_3 J_0\left(4\pi f_{D_{\max}}\right) - c_4 J_0\left(2\pi f_{D_{\max}}\right) \tag{2.25}$$

where $c_3 = \operatorname{sinc}\left(B_n T_s\right) - c_1$ and $c_4 = \operatorname{sinc}\left(2B_n T_s\right) - c_2$. Once an estimate of $f_{D_{\max}}$ has been obtained, the SNR can be estimated using

$$\Gamma = \frac{\operatorname{sinc}\left(2B_n T_s\right) - c_2}{c_2 - J_0\left(4\pi f_{D_{\max}}\right)}. \tag{2.26}$$

A similar derivation for the Doppler and SNR estimators in terms of the signal envelope ACF is also provided in the paper. While the proposed algorithm no longer requires the SNR to be known in order to estimate the Doppler frequency, the performance is still dependent on the quality of the signal. This is reflected by the results provided in the paper which show that a relatively high SNR is necessary for accurate estimates of the speed. Furthermore, it is observed that estimates obtained from the in-phase component are significantly better than those obtained directly from the envelope of the signal. Regardless, the results presented show an error margin (up to 2.5 km/h) that is too large to be useful for pedestrian speed estimation. There are two other points of criticism: first, the algorithm requires inversion of the Bessel function and is generally computationally expensive. Second, the estimators are based around the assumption that the channel is Rayleigh fading and therefore has a particular ACF – this is an assumption that may not hold in practice.

Yet another speed estimation algorithm utilising the autocorrelation and autocovariance functions of a signal envelope received through a Rician fading channel is presented by Zheng and Xiao in [35]. A discrete time model of the system is presented where the channel filter is of

length L. The Doppler estimator function is given as

$$f_{D_{\max}} \approx \frac{1}{\pi m T_s} \sqrt{\frac{V_{|y|^2|y|^2}(L) - V_{|y|^2|y|^2}(m)}{2V_{|y|^2|y|^2}(L)}}; \quad m > L \tag{2.27}$$

where $V_{|y|^2|y|^2}(m)$ is the autocovariance of the received signal power at a lag of m. The channel is assumed to be wide-sense stationary uncorrelated scattering (WSSUS) [29, 43]. Since only the autocovariance of the received signal power is needed to estimate the speed, knowledge of the SNR and channel characteristics such as the Rician parameter and AoA is unnecessary. The results presented in the paper indicate that on average, the algorithm is capable of providing accurate estimates of Doppler frequencies below 100 Hz when the SNR is above 5 dB. Once again, assumptions made regarding the channel model in order to develop the algorithm may not be valid in practice.

Xiao *et al.* propose a simple speed classifier algorithm for time division multiple access (TDMA) based systems [44] in [36]. The estimator is described as

$$\text{speed} = \begin{cases} \text{slow}, & \frac{R(1)}{R(0)} \geq T_L \\ \text{fast}, & \frac{R(1)}{R(0)} \leq T_H \\ \text{indeterminate}, & \text{otherwise} \end{cases} \tag{2.28}$$

where $R(n)$ is the autocorrelation function of the signal power at a lag equivalent to the duration of n slots and T_L and T_H are appropriate thresholds determined through empirical methods. Although the channel is assumed to be fast fading, no further assumptions are made regarding the underlying fading process. It is clear from the estimator that the algorithm only indicates whether the motion of the mobile terminal is fast or slow – an arbitrary definition that depends on the values chosen for T_L and T_H. While this may be useful for the application envisaged in the paper – i.e. minimisation of handoff rates in a hierarchical cellular system, it is clearly inadequate for applications such as indoor positioning where an absolute value of the speed is needed.

A maximum likelihood (ML) estimator for the Doppler frequency is described by Krasny *et al.* in [45]. The algorithm is described in context of a TDMA system where $\hat{h}_q(i)$ corresponds to the channel estimate at the i^{th} pilot location for the q^{th} slot. $i \in [1, N]$ where N is the number of channel estimates available in each slot. The optimal ML estimate for the

Doppler is the $f_{D_{\max}}$ value that minimises the function

$$F_{\text{opt}}\left(f_{D_{\max}}\right) = \ln\left\{\det\left[\mathbf{K}\left(f_{D_{\max}}\right)\right]\right\} + \sum_{i=1}^{N}\sum_{l=1}^{N}\hat{K}(i,l)K^{-1}\left(i,l;f_{D_{\max}}\right) \tag{2.29}$$

where $\det\left[\mathbf{K}\left(f_{D_{\max}}\right)\right]$ is the determinant of the *assumed* correlation matrix $\mathbf{K}\left(f_{D_{\max}}\right)$ and

$$\hat{K}(i,l) = \frac{1}{Q}\sum_{q=1}^{Q}\hat{h}_q(i)\hat{h}_q^*(l) \tag{2.30}$$

is an *estimate* of the channel correlation matrix over Q slots. Although not explicitly shown, $\mathbf{K}\left(f_{D_{\max}}\right)$ is a function of the SNR. Since the ML estimator in (2.29) requires inversion of the channel correlation matrix and knowledge of the SNR, it is both computationally expensive and impractical. In order to surmount these difficulties, a sub-optimal estimator is also proposed in the paper:

$$F\left(f_{D_{\max}}\right) = \frac{1}{Q}\sum_{q=1}^{Q}\sum_{m=1}^{N-1}\left|\frac{\hat{K}_q(m)}{\hat{K}_q(0)} - \frac{K_h\left(m;f_{D_{\max}}\right)}{K_h\left(0;f_{D_{\max}}\right)}\right|^2, \tag{2.31}$$

where

$$\hat{K}_q(m) = \frac{1}{N-m}\sum_{i=1}^{N-m}\hat{h}_q(i)\hat{h}_q^*(i+m) \tag{2.32}$$

is an estimate of the channel correlation matrix for the q^{th} slot and $K_h\left(m;f_{D_{\max}}\right)$ is the assumed ideal (i.e. noise-free) channel correlation matrix. The results presented for a SNR level of 17 dB show that on average, the estimates are accurate. However, a large variance is also present in the estimates obtained from the sub-optimal version of the algorithm.The overall computational complexity and reliance on a particular channel correlation model are the main disadvantages that may prevent the algorithm from being useful in practice.

The use of covariance based methods for speed estimation is discussed extensively by Tepedelenlioğlu and Giannakis in [37] and Anim-Appiah in [38]. The impact of a non-isotropic scattering channel on the performance of algorithms that assume an isotropic scattering environment is also investigated. Additionally, Tepedelenlioğlu and Giannakis provide a method of estimating the Rician parameter from the empirical autocovariance function for speed estimation algorithms that require it.

Level crossing rate of the fading envelope is another measure that can be used to estimate the speed of a mobile device (see Section 2.1.1). This relationship is exploited by Zhao and Mark in [39]. Assuming that the channel is Rayleigh fading, the Doppler frequency estimator is given as

$$f_{D_{\max}} \approx \frac{0.3308}{L\tau} \tag{2.33}$$

where L is the number of sampling intervals within a positive or negative fading envelope slope and τ is the sampling interval. L is the *average fade slope duration* and defined in terms of the the zero crossing rate of the first derivative of the fading envelope, $\mathrm{ZCR}_{\dot{R}}^{-1}$:

$$2L\tau = \mathrm{ZCR}_{\dot{R}}^{-1}. \tag{2.34}$$

Therefore by simply counting the average number of sample points present in the positive and/or negative going slopes of the fading envelope, the Doppler frequency and hence the speed can be estimated. The results presented in the paper indicate that the algorithm performs better at speeds typically associated with motorised vehicles. The accuracy drops considerably at pedestrian velocities where the variance of the speed estimates is increased significantly.

No less than five different methods based on LCR for systems supporting diversity combining [46] are presented by Zhang and Abdi in [40]. The first of these is applicable to selection combining (SC) systems with L branches:

$$f_{D_{\max}}^{\mathrm{IZCR}} = \frac{\sqrt{2}N_x(0,T)}{TL\sum_{k=0}^{L-1}\binom{L-1}{k}\frac{(-1)^k}{\sqrt{k+1}}}, \tag{2.35}$$

where $N_x\left(x_{\mathrm{th}},T\right)$ is the number of times the in-phase component of the fading envelope crosses the threshold x_{th} in the positive or negative going direction within some time period T. This is the *in-phase zero crossing rate* estimator for Doppler frequency. The *in-phase rate of*

maxima estimator for SC systems is defined as:

$$
\begin{aligned}
f_{D_{\max}}^{\text{IROM}} &= \frac{M_x(T)}{\Omega}, \\
\Omega &= \frac{TL}{4\pi}\sum_{k=0}^{L-1}\binom{L-1}{k}(-1)^k \\
&\times\left[\frac{2\pi}{\left(k^2+4k+3\right)^{\frac{1}{2}}}+\frac{2\sqrt{2}\pi}{(k+1)^{\frac{3}{2}}}-\frac{4}{3}\int_0^{\frac{\pi}{2}}\frac{{}_2F_1\left(\frac{3}{2},2,\frac{5}{2},-\frac{k+1}{2}\cos^2\theta\right)}{\cos^2\theta}d\theta\right]
\end{aligned}
\tag{2.36}
$$

where $M_x(T)$ is the number of maxima of the in-phase component within a period T, θ is the phase at the output of the combiner and ${}_2F_1(\cdot)$ is the hypergeometric function [47]. The much simpler *instantaneous frequency zero crossing rate* based estimator is

$$
f_{D_{\max}}^{\text{FZCR}} = \frac{2N_{\dot{\theta}}(0,T)}{T}, \tag{2.37}
$$

where $\dot{\theta}$ is the first derivative of the phase at the output of the combiner. A pair of estimators based on the *envelope level crossing rate* are also presented for systems utilising either selection combining or maximal ratio combining:

$$
\begin{aligned}
f_{D_{\max}}^{\text{ELCR,SC}} &= \frac{N_r(1,T)}{\sqrt{2\pi}TL\sum_{k=0}^{L-1}\binom{L-1}{k}(-1)^k e^{-(k+1)}}, \\
f_{D_{\max}}^{\text{ELCR,MRC}} &= \frac{e\Gamma(L)N_{r^{\text{MRC}}}(1,T)}{\sqrt{2\pi}T}.
\end{aligned}
\tag{2.38}
$$

The symbol for the signal envelope from the selection combiner is r while that from the maximal ratio combiner is r^{MRC}. $\Gamma(\cdot)$ is the gamma function [47]. All five estimators assume that the channel is noise free and isotropic scattering.

Despite the high complexity associated with evaluating the estimators, the results presented in [40] show that the performance is sub-optimal. Even with a SNR of 10 dB, estimation errors greater than 50% are common in the results presented in the paper.

A method of estimating the speed from a wavelet analysis [48] of the fading envelope is described by Narasimhan and Cox in [49]. For a Rayleigh fading channel with isotropic

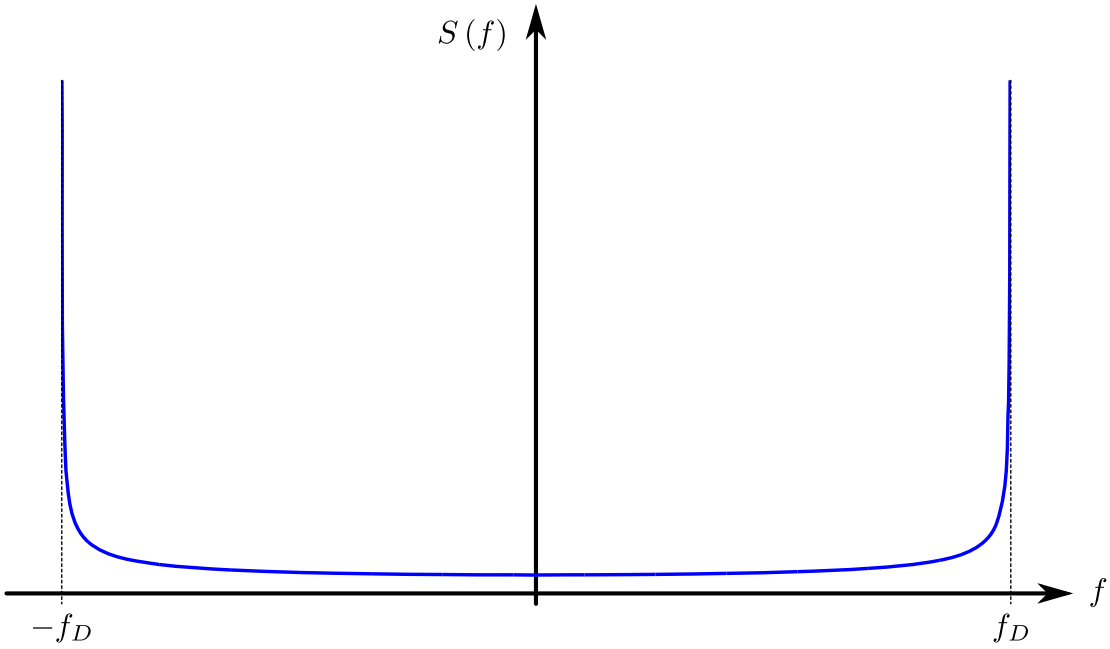


Figure 2.6: *Classical Doppler power spectrum for a Rayleigh fading multipath channel with isotropic scatterers.*

scattering, the estimator is

$$v(t) \approx \frac{0.662\lambda_c}{\Delta T(t)} \tag{2.39}$$

where $0.662\lambda_c$ is the average spatial distance between successive local minima of the fading process (Figure 2.1) described in terms of the signal wavelength, λ_c. $\Delta T(t)$ is the average temporal separation between the local minima in a neighbourhood of time t and extracted from the envelope by means of the continuous wavelet transform (CWT). Analysis of the signal envelope using CWT reveals the points of discontinuity in the first derivative and hence can be used to obtain an average value for $\Delta T(t)$. Since CWT is inherently a multiresolution analysis technique, the described method does not require an adjustable observation window – unlike most LCR and covariance based algorithms. Although the results show that the algorithm is capable of providing accurate estimates, the complexity associated with the CWT is a major disadvantage.

In a Rayleigh fading channel with isotropic scattering, most of the energy is concentrated around the frequencies $f_c \pm f_D$ where f_c is the original carrier frequency and f_D is the Doppler frequency. This is the classical U-shaped Jakes Doppler power spectrum [50, 51]. The nor-

malised spectrum is described analytically as

$$S(f) = \frac{1}{\pi f_D \sqrt{1-(f/f_D)^2}}, \quad |f| \leq f_D \tag{2.40}$$

and illustrated in Figure 2.6. This behaviour is exploited for speed estimation in the paper by Zhang and Abdi [52]. For a narrowband channel, the estimator is simply

$$f_{D_{\max}} = \left| \arg\max_f \hat{S}(f) \right| \tag{2.41}$$

where $\hat{S}(f)$ is the estimated spectrum. For a wideband channel where each of the L individual path may be resolved, the estimator is

$$f_{D_{\max}} = \max \left\{ \hat{f}_D^l \right\}_{l=1}^{L}, \tag{2.42}$$

where $\hat{f}_D^l = \left| \arg\max_f \hat{S}_l(f) \right|$ and $\hat{S}_l(f)$ is the spectrum associated with the l^{th} path. Although the algorithm itself is simple to implement, results provided in the paper show that errors larger than 10% can be expected even at low Doppler frequencies (41 Hz) – making the algorithm unsuitable for pedestrian positioning applications.

Finally, a novel method of estimating the Doppler frequency from the local oscillator adjustments made at the mobile device is presented by Smith and Cox in [53]. It is a closed-loop Doppler estimation algorithm where the base station's oscillator serves as the reference frequency source. If the actual downlink frequency is $f_{c,\text{dl}}$, Doppler shift causes the mobile terminal to track to and use an uplink frequency of $f_{c,\text{ul}} = f_{c,\text{dl}} + f_D$. Assuming channel reciprocity, the uplink signal subsequently received by the base station is at a frequency of $\hat{f}_{c,\text{dl}} = f_{c,\text{ul}} + f_D = f_{c,\text{dl}} + 2f_D$. Since $f_{c,\text{dl}}$ is the reference frequency, the base station can estimate the Doppler shift using

$$f_D = \left| \frac{f_{c,\text{dl}} - \hat{f}_{c,\text{dl}}}{2} \right|. \tag{2.43}$$

A LoS channel is assumed and at the absence of AoA information, only a lower bound on the speed of the mobile device can be obtained from the estimated Doppler frequency.

Review of the current literature on speed estimation algorithms reveals that while there are a

large number of methods available, most have a heavy computational burden and/or produce estimates that are not sufficiently accurate to be usable for positioning applications. Almost all of the algorithms are also based around specific models of the wireless channel – an assumption that may not hold under realistic conditions. In order to overcome these disadvantages associated with existing methods, a pair of simple speed estimation algorithms that do not rely on any particular channel model are presented in Chapter 3.

2.2 Anomaly detection

Anomaly detection, also known as novelty or outlier detection, is a rich field of research with a very large body of work that exists in literature. The existence of multiple survey-type papers such as [22, 54–60] is a testament to the true extent of the subject of anomaly detection. Nevertheless, it is still very much an active area of research lacking generic algorithms that can be applied universally to anomaly detection problems. The difficulty in designing a unified solution arises from the fact that it is very challenging to develop a methodology that can describe all types of anomaly detection problems equally well. As a result, most of the methods encountered in literature are based on tightly constrained frameworks that apply to very specific classes of problems.

Existing techniques of anomaly detection can be separated into a handful of categories depending on the underlying approach. *Classification* based methods [22,56,58,61–63] utilise supervised machine learning techniques to categorise nominal and anomalous behaviour while *clustering* and *nearest-neighbour* based techniques [22, 54–56, 64–66] rely on measures of the distance between points of data. Both of these approaches rely on the assumption that nominal events can be separated into distinct clusters or classes that do not overlap with anomalous events. In contrast, *Statistical* techniques [54, 57, 60, 67–69] detect anomalies by comparing the test data points against stochastic models of nominal behaviour. *Information theoretic* methods [22, 70–72] employ measures of information such as Kolmogorov complexity and entropy [73] and work under the assumption that anomalies lead to a change in the information content. The algorithms presented in Chapter 4 employ techniques that are jointly statistical and information theoretic in nature.

A critical review of a number of current anomaly detection algorithms follows. It is, however, difficult to perform direct comparisons between algorithms due to differences in the appli-

cation areas and the measures employed for performance evaluation. It has been noted by Patcha and Park in [54] that there is currently no globally acceptable standard or metric for evaluating the performance of anomaly detection algorithms. Furthermore, evaluations of performance based on classical methods such as the receiver operating characteristic (ROC) curve [74] are often misleading and/or incomplete. Development of a general systematic methodology and a set of metrics that can be used to fairly evaluate anomaly detection systems is still an open challenge.

A statistical method for detecting outliers in sensor data streams is proposed by Basu and Meckesheimer in [75]. Relying on the assumption that the input data stream is continuous and locally smooth, the method exploits the fact that correlation between neighbouring data points is expected to be higher than between points separated by a relatively long duration of time. The described algorithm employs windowing to compare the local median against the data point under test. If the difference between the two is observed to be larger than a predefined threshold, the tested data point is labelled as an outlier. The low complexity and windowed nature of the algorithm allows processing of non-stationary signals in real-time. Performance evaluation is done using *Jaccard coefficients* [76] and is observed to be dependent on the threshold and the window length – as expected. Due to the rapid local envelope fluctuations typically exhibited by modulated RF signals, the algorithm is unsuitable for the detection of complex anomalies in communications systems.

A similar approach based on regression modelling of the underlying time series is described by Ma and Perkins in [77]. The algorithm requires that a technique called support vector regression (SVR) (described within the paper) is used on training data to construct a model of the sequence. A margin ($\pm\epsilon$, "tolerance width") to the regression line is used as a threshold for the detection of unusual events. In operation, the algorithm compares events against the model by calculating the difference between the actual event $x(t_0)$ at time t_0 and the model $M(t_0 - 1)$ at time $t_0 - 1$. If the value of the difference is found to be outside the range $(-\epsilon, \epsilon)$, the event is labelled as "surprising". A temporal sequence of a given length is then considered as novel if it contains a predefined minimum number of surprising events. After testing event $x(t_0)$, it is integrated into the model to yield $M(t_0)$. In addition to this basic approach, the authors present a windowed version of the algorithm where the regression model consists of only a fixed number historical samples. Due to the complexity involved in computing the regression function, the algorithm is costly to implement. Additionally, the algorithm

requires the specification of six different parameters, at least two of which are non-intuitive. While results obtained from test data sets appear to show the effectiveness of the algorithm, no measures of performance are provided in the paper.

In [78] and [21] Keogh *et al.* describe a simple and intuitive method of detecting unusual sequences ("discords") in a finite time series. In addition to being of finite length, the time series must also be periodic. The described algorithm works by splitting the time series in a number of smaller, equal-size sequences ("subsequences") that are compared against one another. Using Euclidean distance as the measure of similarity, exhaustive search across all possible pairings of subsequences yields the one that is most dissimilar. Although heuristics for significantly reducing the search complexity compared to a brute force approach is described in the paper, the method still remains computationally expensive. Additionally, due to its very nature, i.e. the discovery of unusual sequences in *finite* and *periodic* time series, it cannot be used to analyse modulated signal envelopes in real-time. It is also useful to have *a priori* knowledge of the length of the discord since it dictates the choice for the length of the subsequences and the detection performance. Once again, the authors demonstrate the effectiveness of the algorithm by comparing the results obtained from test data sets against expert opinions from a human and no formal evaluation of the algorithm's performance is given.

A hypothesis test approach to the problem of anomaly detection is described by Yeung and Chow in [79]. The first step of the algorithm is to generate a probabilistic generative model $\mathscr{M}$ of the nominal sample space using the Parzen windows method [80] of non-parametric probability density estimation. The model is subsequently used to generate a representative sample set $\mathscr{V}$. Given $\mathscr{M}$, a random sample $y \in \mathscr{V}$ and the test sample x, the decision is based on the probability inequality

$$\Pr(L(y) \leq L(x)) < \psi; \quad 0 < \psi < 1 \tag{2.44}$$

where $L(x) = \log p(x|\mathscr{M})$ is the log-likelihood of x with respect to the model $\mathscr{M}$ and ψ is a threshold that determines the algorithm's false alarm rate. If the condition in (2.44) is satisfied, x is deemed to be anomalous with respect to the model. The algorithm is employed for the detection of network intrusion attempts and the performance is quantified using measures of the true acceptance rate and the true detection rate. Results indicate that the algorithm is capable of providing true detection rates higher than 90% with a false alarm rate of 1% for

certain types of intrusion attempts. The main disadvantage of the technique is complexity – Parzen windows method of density estimation and the testing phase of the algorithm are both costly operations that make real-time implementations difficult.

Yet another statistical technique of anomaly detection is described by Desforges *et al.* in [69]. Once again, the Parzen windows method (i.e. kernel method of density estimation) is employed to derive a smooth, non-parametric probabilistic model of the nominal, underlying sample space. Unlike the aforementioned method by Yeung and Chow [79], the test is between two models rather than a single sample against a reference model. In order to perform a test, the probability density estimate of the test set conditioned on the reference model is compared against a probability density estimate of clean data set conditioned on the same reference model. If a mismatch is observed between the density estimates, the test set is labelled as anomalous. In the examples provided in the paper, the authors simply perform a visual comparison and do not specify a robust procedure that can be used to quantify the difference between the density estimates. Additionally, due to the use of kernel method for density estimation, the computational burden of the algorithm is too high for an efficient, real-time implementation.

A large number of model based anomaly detection techniques, including one proposed in this thesis, assume the availability of clean training data to establish the point of reference. However, this is an assumption that is often not valid in practice. Given a noisy data set that contains a large number of a nominal events and a few anomalous events, a method of extracting the anomalous events is described by Eskin in [81]. The algorithm assumes that the distribution of the entire data set, $\mathbf{D}$, can be explained by means of a mixture model

$$\mathbf{D} = (1-\lambda)\mathbf{M} + \lambda\mathbf{A}, \tag{2.45}$$

where $\mathbf{M}$ is the distribution associated with the nominal (majority) events, $\mathbf{A}$ is the distribution associated with anomalous events and λ is the probability is that a given data point is from $\mathbf{A}$. The goal of the algorithm is to separate the data set D into two subsets: M_t is the majority subset and A_t is the anomalous subset at time t, after processing element x_t. Initially, $M_0 = D$ and $A_0 = \emptyset$. The log-likelihood function of $\mathbf{D}$ is crucial for the test and is

given as

$$LL_t(\mathbf{D}) = |M_t|\log(1-\lambda) + \sum_{x_i \in M_t} \log\left(P_{M_t}(x_i)\right) + |A_t|\log(\lambda) + \sum_{x_j \in A_t} \log\left(P_{A_t}(x_j)\right) \tag{2.46}$$

for time t. To test the hypothesis, LL_t is computed with event x_t temporarily removed from M_{t-1} and added to A_{t-1}. If the difference $LL_t - LL_{t-1}$ is larger than some predefined threshold, x_t is deemed to be anomalous and left in A_t. The probability distributions P_{A_t} and P_{M_t} are also updated at every step to reflect the change. Results obtained from applying the algorithm to data sets of UNIX system calls containing various proportions of anomalous events show that the it is possible to obtain perfect intrusion detection rate (100%) with a very low false positive rate (much less than 1%). The performance is seen to degrade with an increase in the proportion of anomalous events in the data set. Despite its excellent overall performance, the main criticism of the method is its reliance on an *a priori* knowledge of the probability of anomalous events, λ – this information is generally unknown for real test data. An indication of the degradation in performance that can be expected when λ is not known exactly is not given. The algorithm is also unsuitable for detecting signal anomalies in real-time since it is designed to work offline on finite data sets.

An interesting method of detecting anomalous packets in a network in real-time is described by Krügel *et al.* in [19]. A composite anomaly score, AS, is computed from statistical measurements of three different characteristics of the packets:

$$AS = 0.3AS_{\text{type}} + 0.3AS_{\text{len}} + 0.4AS_{\text{pd}} \tag{2.47}$$

where the sub-scores AS_{type}, AS_{len} and AS_{pd} are based on the packet type, length and payload distribution respectively. The sub-scores the weighted according to their relative importance. AS_{type} is inversely proportional to the relative frequency of the packet type and is calculated using the expression

$$AS_{\text{type}} = -\log_2\left[p(type)\right], \tag{2.48}$$

where $p(type)$ is the probability associated with the given *packet type*. AS_{len} is derived from the length l of the test packet and the mean μ and variance σ^2 of the packet lengths observed

in clean training data:

$$AS_{\text{len}} = 1.5^{\frac{(l-\mu)}{2.5\sigma}}. \tag{2.49}$$

It is assumed that malicious behaviour produces packets that are longer than usual. Finally, AS_{pd} is calculated from a χ^2-test [82] performed on the distribution of ASCII characters in the payload, normalised by the packet length:

$$AS_{\text{pd}} = \chi^2\left(\frac{15}{l}\right). \tag{2.50}$$

Results presented in the paper show that the approach is capable of successfully detecting a number of a network intrusion attempts. However, extended evaluation of the performance is not provided. Additionally, the algorithm is unsuitable for detecting anomalous events in a signal envelope due to the fact that information regarding the length, type and payload distribution is only available after the signal has been demodulated.

Finally, a discussion on the use of information theoretic measures for anomaly detection is presented by Lee and Xiang in [70]. Despite the title, the focus of the paper is on developing suitable models for anomaly detection and not the process itself. Information theoretic measures such as entropy can be used to determine the "regularity" and hence the quality of the reference models. The authors argue that higher anomaly detection performance is obtained from reference models that are regular (i.e. possess low entropy). The use of relative entropy (i.e. Kullback-Leibler divergence (KLD)) is suggested for testing the reference model against data obtained during testing. Once again, the motivation for this is to validate the quality of the reference model by comparing the regularity of the training set against the test set.

It is evident from this review of the existing anomaly detection literature that there is currently no algorithm that can be readily used to detect anomalies in a RF signal envelope in real-time with limited system resources. The main deficiencies that have been identified are high computational complexity and the requirement for *a priori* knowledge that is often not available in reality. The complexity in particular is a disadvantage associated with most existing algorithms – as also pointed out by Chandola *et al.* in [22]. Since hand-held communications devices have limited processing power and energy, it is necessary to employ algorithms that are effective yet simple. The novel algorithms presented in Chapter 4 aim to address these issues by means of methods that are computationally simple, effective and require no *a priori* knowledge regarding the test signal.

2.3 Chapter Summary

A critical review of the exiting methods for speed estimation and anomaly detection is provided in this chapter. Strengths and weaknesses of some typical algorithms described in the literature are identified and presented. One weaknesses that is common to both anomaly detection and speed estimation algorithms is high computational complexity. It is considered to be an automatic disqualification since the target platform (mobile handset) is a device with limited processing power and energy.

Speed estimation algorithms typically utilise the second order statistics or the level crossing rate of the underlying fading envelope process. Consequently, assumptions are made regarding the channel model which may not hold in practice. Methods that rely on spectral decomposition of the signal for Doppler frequency extraction are generally complex and produce results that are strongly affected by the state of the radio environment.

A review of typical statistical and information theoretic methods of anomaly detection reveals that the described methods are often very application specific and therefore only readily applicable to the problems for which they have been originally developed. Some of the algorithms also require a significant amount of *a priori* information that may not always be available in practice. Although time series analysis algorithms do exist in the literature, they are typically for sequences of finite length and not applicable signals that the structurally complex – such as those from communications systems.

To summarise, existing speed estimation and anomaly detection algorithms are generally complex, assume a particular model for the underlying random process and require a significant amount of *a priori* information that may be difficult to obtain – challenges that the work presented in Chapters 3 and 4 attempt to address.

Chapter 3
Mobile speed estimation

3.1 Introduction

THE most widespread and popular form of positioning and navigation technology in use today is the global positioning system (GPS). Unfortunately it cannot be used indoors or in other areas such as natural and urban canyons that experience heavy shadowing. This is due to the fact that the system relies on a line-of-sight connection to multiple space based satellites simultaneously to triangulate the receiver position [83]. To overcome this deficiency in GPS coverage, a number of alternative triangulation techniques based on terrestrial signal sources such as cellular telephony radio towers [5, 84], wireless local area network (WLAN) access points [4, 85] and purpose-built ultra-wideband (UWB) beacons [6, 86] have been proposed in the literature. While the positioning accuracy achieved varies considerably depending on the technology employed, all of these techniques have one thing in common – a network of beacons (i.e. infrastructure) is required for the positioning system to function.

When triangulation based methods cannot be used due to a lack of existing infrastructure, techniques such as dead reckoning (DR) may be used to obtain an estimate of the position. Starting at a reference location with known absolute spatial coordinates, integrating the velocity (speed and heading) profile of the receiver unit over time yields an approximation of the instantaneous position [83]. Assuming that the coordinate of the starting position is accurate, performance of DR depends on two factors: frequency and accuracy of the speed and heading information used to build the velocity profile.

Accurate electronic heading information is easily obtained from sensors such as a ring laser gyro [7,87]. While relatively accurate measures of speed can also be obtained through ground-contact electro-mechanical devices such as a vehicular speedometer, such methods are clearly unsuitable for use with hand-held devices. There are two distinct classes of solutions to this problem: methods belonging the first, known as inertial navigation system (INS), employ accelerometers while the second utilise a radio frequency (RF) front-end to estimate the speed

from characteristics of received signals. The main disadvantages of INS arise from accelerometer related errors. The sensors are adversely affected by a wide range of physical phenomena such as vibration, shock, thermal fluctuations, angular acceleration and – under certain circumstances – gravity. Frequent updates of the absolute position are also required with INS as bias errors tend to accumulate [88]. Speed estimation devices based on analysis of radio signals are immune to the physical phenomena affecting INS due to the fact that the system (RF front-end & processor) does not contain any moving parts that are sensitive to motion.

Most of the speed estimation algorithms described in existing literature rely on the signal strength fluctuations caused by a time-varying multipath channel. While methods based on the level crossing rate (LCR) of the received fading envelope typically provide good results for high speeds, the accuracy drops considerably at speeds associated with pedestrians and other low velocity entities [39, 89]. As an alternative to LCR based methods, continuous wavelet transform (CWT) is employed in [49] to extract the speed information from the signal. While the procedure is capable of providing a good level of accuracy, its shortcoming is the computational complexity associated with computing the CWT of the signal. Methods based on signal envelope auto-covariance and autocorrelation function (ACF) [34, 38] can provide reasonable results but generally require the signal to noise ratio (SNR) to be high. In order to overcome these challenges, a novel pair of fast and effective algorithms have been developed and are presented in this chapter.

The algorithms presented herein exploit the spatiotemporal correlation between unique signal signatures to estimate the speed of the receiver unit [8, 9]. In its simplest form, the first algorithm requires a pair of antenna elements with a fixed separation distance aligned in the direction of motion. An estimate of the time required by the receiver to traverse a distance equivalent to the inter-antenna separation distance is then obtained by correlating the instantaneous signal signature observed at the trailing antenna against those previously observed at the leading antenna. With an estimate of the time delay, the speed can be easily calculated since the distance travelled is equal to the known inter-antenna separation distance. The correlation frequency is adjusted adaptively to provide accurate estimates at both high and low speeds. The second speed estimation algorithm is based around an equation that describes the theoretical spatial correlation function. Once the environment specific parameters of the equation have been estimated, the correlation value between two successive signal signatures can be used to obtain an estimate of the distance from the spatial correlation function. Since

both of the channel signatures can be obtained from the same antenna element, a receiver with only a single antenna may be sufficient. The speed of the device is once again easily calculated since the time delay between the channel signatures is assumed to be known in this case.

The proposed algorithms are simple and require little in terms of supporting infrastructure – a single transmitter broadcasting a wideband RF signal is sufficient. It is not necessary to install new infrastructure since it is possible to utilise existing terrestrial transmitter networks such as those used for terrestrial digital video broadcasting (DVB-T). Dedicated receiver units are also unnecessary since mobile communications equipment incorporating multiple-input multiple-output (MIMO) technology and software defined radio (SDR) can be easily reconfigured to utilise the signal from any transmitter within range of the device.

The proposed speed estimation algorithms are described in detail in Section 3.2. Results obtained from software simulations are provided in Section 3.3 while Section 3.4 concludes the chapter with a summary of the contributions made.

3.2 Speed estimation algorithms

3.2.1 Spatiotemporal fading channel

As a result of the multitude of propagation paths that typically exist between a radio transmitter and a receiver, multipath fading is a relatively common phenomenon in wireless channels. The spatiotemporal impulse response of such a channel can be described using (see Section 2.1.1 and [29,30]):

$$\mathbf{h}(\tau,t)=\sum_{l=0}^{L-1}\alpha_l \exp\left[j\left(\phi_l+2\pi f_{D_l}t\right)\right]\mathbf{a}\left(\theta_l\right)\delta\left(\tau-\tau_l\right) \tag{3.1}$$

where L is the total number of multipath components observed at the receiver. For the l^{th} multipath component, α_l is the relative gain, ϕ_l is the carrier phase shift, f_{D_l} is the Doppler shift, θ_l is the angle of arrival (AoA) and τ_l is the time delay. $\delta(\cdot)$ is the unit impulse function and $\mathbf{a}\left(\theta_l\right)$ is the array response vector. When the signal and the antenna array (containing m antennae) at the receiver are restricted to a two-dimensional (2D) space, the array response

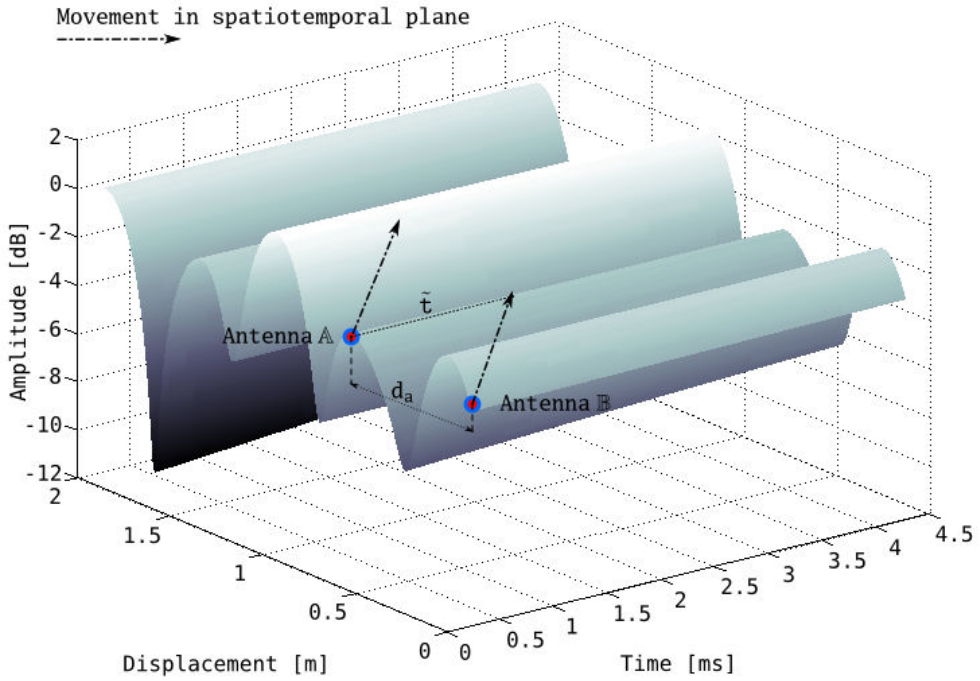


Figure 3.1: *Rayleigh fading spatiotemporal multipath channel for a carrier frequency of* 474*MHz (DVB-T) and a typically pedestrian speed of* 1.5 *m/s. It shows rapid changes along the spatial dimension while remaining relatively stable in time. At any given time, different channel characteristics are observed by two sufficiently distant (d_a) antenna elements ($\mathbb{A}$ and $\mathbb{B}$) aligned in the direction of motion. However, temporal stability of the channel implies that after a certain amount of time $\tilde{t}$, $\mathbb{B}$ can experience a channel similar to that previously experienced by $\mathbb{A}$. This information can be used to estimate the speed using $|\vec{v}| = d_a/\tilde{t}$.*

vector is given by

$$\mathbf{a}(\theta_l) = \left[\exp\left(-j\Psi_{l,1}\right),\ \exp\left(-j\Psi_{l,2}\right),\ \exp\left(-j\Psi_{l,3}\right),\ \ldots,\ \exp\left(-j\Psi_{l,m}\right)\right]^T \tag{3.2}$$

where $\Psi_{l,i} = \left[x_i \cos(\theta_l) + y_i \sin(\theta_l)\right] \mathscr{B}$. The spatial coordinate pair (x_i, y_i) represents the location of antenna element i in the 2D plane. $\mathscr{B} = \frac{2\pi}{\lambda_c}$ is the wavenumber [30] and λ_c is the carrier wavelength. The number of multipath components, relative path gains, phase shifts, Doppler frequency, AoA and path delay are generally time-varying parameters; even though this fact is not explicitly clear from (3.1).

A simulated realisation of the channel described by (3.1) is shown in Figure 3.1 for a typically pedestrian speed of 1.5 m/s. The carrier frequency, f_c, of the constant envelope signal is 474 MHz – corresponding to the first channel of the DVB-T service [31]. It is assumed that no

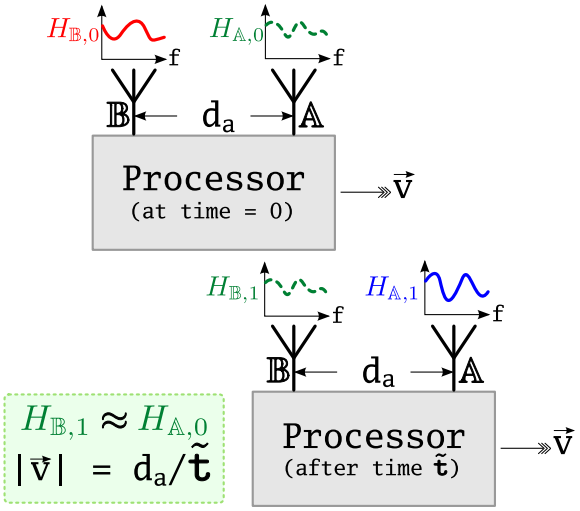


Figure 3.2: *Speed estimation algorithm using RRFSM. The time $\tilde{t}$ required by the device to traverse the fixed distance d_a can be estimated by correlating the channel response estimates obtained at the two antennae. From an estimate of $\tilde{t}$, the speed is easily calculated using $|\vec{v}| = d_a/\tilde{t}$.*

line-of-sight (LoS) paths exist and hence the channel is Rayleigh fading. The relative path gain, phase shift and delay values used for the channel realisation are identical to the reference values specified in Annex B of [31]. Since the specification does not state reference values for the AoA, it is assumed to be a uniform random variable distributed in $[0, 2\pi]$.

The plot shows that the spatiotemporal channel exhibits rapid variations across space while remaining largely unchanged over the displayed time scale. This is expected due to the relatively slow speed of 1.5 m/s which produces a maximum Doppler shift of only 1.6 Hz – leading to a lengthy channel coherence time of approximately 260 ms. This property of the multipath channel is exploited in the RF signature matching based speed estimation algorithm described in the following section.

3.2.2 Relative radio frequency signature matching algorithm

Given a RF signal source and a receiver equipped with multiple antennae, the relative RF signature matching (RRFSM) technique can be used to estimate the speed of the receiver unit. It is a relative comparison technique since the estimate of the speed is obtained by processing a number of successive channel response estimates.

It is possible to estimate the speed of a mobile wireless device equipped with multiple receiving antennae by comparing the channel frequency response observed at each of the antenna

array elements. The simplest such device consists of a receiver with a pair of antennae as illustrated in Figure 3.2. In order to illustrate the principle, movement of the device is assumed to be restricted to a single spatial dimension and the antenna array is aligned in parallel to the direction of motion.

In operation, the speed estimator device traces a diagonal path in the spatiotemporal plane as illustrated previously in Figure 3.1. Labelling the antennae by the letters $\mathbb{A}$ and $\mathbb{B}$ respectively, the starting position of the two can be denoted by the spatiotemporal coordinate pair $\left[x_{\mathbb{A},t_0}; x_{\mathbb{B},t_0}\right]$ where $x_{\mathbb{A},t_0}$ represents the position of $\mathbb{A}$ at time t_0. The radio channel sensed at the array elements, $H_{\mathbb{A}}$ and $H_{\mathbb{B}}$, can be similarly represented by $\left[H_{\mathbb{A},t_0}; H_{\mathbb{B},t_0}\right]$.

Assuming a non-zero velocity, the coordinates are $\left[x_{\mathbb{A},t_0+t}; x_{\mathbb{B},t_0+t}\right]$ and the channel response vectors are $\left[H_{\mathbb{A},t_0+t}; H_{\mathbb{B},t_0+t}\right]$ after some time t. For some value of $t = \tilde{t}$, $x_{\mathbb{B},t_0+t} = x_{\mathbb{A},t_0}$ and $H_{\mathbb{B},t_0+t} \approx H_{\mathbb{A},t_0}$ due to the relatively large coherence time of the channel. This implies that the device travels a distance $d_a = x_{\mathbb{B},t_0+t} - x_{\mathbb{B},t_0}$ in that time. Since d_a is the inter-antenna separation distance, it is constant and known *a priori*. Therefore, the speed of the device is easily estimated from an estimate of $\tilde{t}$ using

$$\left|\vec{v}\right| = \frac{d_a}{\tilde{t}}. \tag{3.3}$$

It is possible to obtain an estimate of the time delay, $\tilde{t}$ by comparing the channel responses observed at the antenna array elements. Accurate channel estimates or knowledge of the underlying fading process is not required as the proposed method only searches for a relative match between channel responses. This also implies that a reasonably low amount of uncertainty and errors in the estimates can be safely ignored as long as all the estimates are affected in a similar fashion.

Clearly, a history of the channel responses observed at the foremost antenna ($\mathbb{A}$ in this case) must be maintained. Figure 3.3 illustrates the structure of a database that stores the current and $N_{\mathbb{A}}$ previous channel response estimates (CREs) and associated timestamps for $\mathbb{A}$. The instantaneous CRE of $\mathbb{B}$ is also stored for the duration of the signal correlation computation.

The operation used to locate the $H_{\mathbb{A}}$ leading to maximum correlation is:

$$\ell = \arg\max_k \left(H_{\mathbb{B}} \star H_{\mathbb{A},k}\right);\ k = 1, \ldots, N_{\mathbb{A}} \tag{3.4}$$

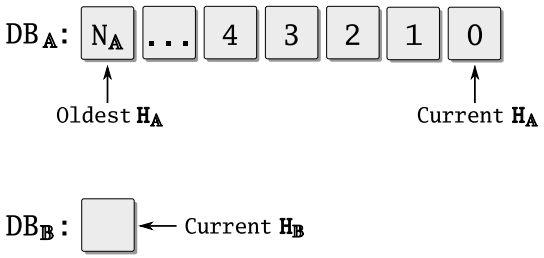


Figure 3.3: *Structure for the database used to store the channel response estimates.*

where $H_{\mathbb{A},k}$ is the k^{th} entry in the database. Then ℓ is the index that corresponds to the CRE from $\mathbb{A}$ yielding maximum correlation (i.e. $H_{\mathbb{A},\ell}$) and it can be used to determine the elapsed time from the associated timestamp. The binary operator "$\star$" denotes cross-correlation at a lag of zero. It is unnecessary to compute $H_{\mathbb{B}} \star H_{\mathbb{A},0}$ since it implies a time delay of zero and hence infinite speed. The maximum correlation value obtained must also be above a threshold to ensure that erroneous speed estimates are not obtained when the device is at a standstill or simply moving too slowly. Therefore, if the correlation threshold is not met, the speed is assumed to be zero. It is important to pick a suitable threshold as a value too low may result in erroneous results while a value too high will reduce the sensitivity of the algorithm. The threshold is lower bounded by the correlation expected between two points in space separated by the inter-antenna distance d_a. A spatial correlation function such as the one described by (3.14) can be employed to obtain this minimum value.

The cross-correlation of two signals at a lag of zero is identical to their *dot-product*. As a result, (3.4) can be simplified to a series of normalised dot-product calculations followed by maximum detection:

$$\ell = \arg\max_{k} \left(\frac{H_{\mathbb{B}} \cdot H_{\mathbb{A},k}}{||H_{\mathbb{B}}||\,||H_{\mathbb{A},k}||} \right);\ k = 1,\ldots,N_{\mathbb{A}}. \tag{3.5}$$

The range of speeds detectable with such a device ultimately depends on the CRE estimation rate and size of the $H_{\mathbb{A}}$ history buffer. With $v_{\min}$ and $v_{\max}$ as bounds on the detectable speed range, the time delay between CRE estimates must be less than $\frac{d_a}{v_{\max}}$ for some inter-antenna separation distance d_a. On the other hand, correct estimation of $v_{\min}$ requires the ability to store CRE estimates spanning a time of at least $\frac{d_a}{v_{\min}}$. Assuming that the CRE estimation interval is short enough to detect $v_{\max}$, the total number of CRE vectors that need to be kept

in memory at any given time is

$$N_{\mathbb{A}} = \left\lceil \frac{\left(\frac{d_a}{v_{\min}}\right)}{\left(\frac{d_a}{v_{\max}}\right)} \right\rceil = \left\lceil \frac{v_{\max}}{v_{\min}} \right\rceil . \tag{3.6}$$

Since the time delay $\tilde{t}$ is estimated from correlation between CREs, the instantaneous channel response must be estimated periodically. For a wireless system with symbol duration T_s, the CRE estimation interval $\hat{i}$ can be defined as the number of complete symbols that elapse between successive CRE estimates:

$$\hat{i} = \left\lfloor \frac{\varsigma}{\mu} \right\rfloor , \tag{3.7}$$

where ς is a constant defined as

$$\varsigma = \frac{d_a}{T_s \, v_{\max}} \tag{3.8}$$

and μ is an adaptive scaling parameter. By adjusting the CRE estimation interval in accordance with the estimated speed of the device, it attempts to provide higher accuracy in the time delay estimates.

The parameter μ is a real number bounded by the constant ς and 1, allowing $\hat{i}$ to assume integer values within the range defined by $\hat{i}_{\min} = 1$ and $\hat{i}_{\max} = \lfloor \varsigma \rfloor$. A CRE estimation interval lower than $\hat{i}_{\min}$ is impossible as it implies an interval shorter than a symbol. An interval longer than $\hat{i}_{\max}$ is unnecessary as it is only needed for speed estimates lower than $v_{\min}$ for a given value of $N_{\mathbb{A}}$.

Initially the adaptive parameter μ is set to unity, resulting in $\hat{i} = \hat{i}_{\max}$. This allows for a rough estimate, v_{est}, to be obtained for any speed between $v_{\min}$ and $v_{\max}$. Given the rough speed estimate, linear interpolation is employed to obtain the new value of μ and hence $\hat{i}$:

$$\mu = \begin{cases} \varrho \cdot (v_{\text{est}} - v_{\min}) + 1, & v_{\min} \leq v_{\text{est}} \leq (v_{\max} + \epsilon) \\ 1, & \text{otherwise} \end{cases} \tag{3.9}$$

where

$$\varrho = \frac{\varsigma - 1}{v_{\max} - v_{\min}} \tag{3.10}$$

and ϵ is a small number required to allow the adaptive procedure to converge at the upper

bound of the detectable speed range. Defining the maximum speed detectable with $\mu = 1$ as

$$\tilde{v}_{\max} = \frac{d_a}{T_s \hat{i}_{\max}}, \tag{3.11}$$

ϵ is defined as

$$\epsilon \geq |\tilde{v}_{\max} - v_{\max}|. \tag{3.12}$$

While the first case in (3.9) is useful for adapting the value of μ while the device is in motion, the second case is necessary to reset the value of μ to unity in case of erroneous estimates (e.g. $v_{\text{est}} \gg v_{\max}$) or when the device is stationary ($v_{\text{est}} = 0$).

Due to the integer nature of $\hat{i}$, any time delay estimated is generally an integer multiple of T_s and therefore the speed estimates between $v_{\min}$ and $v_{\max}$ are not continuous. The estimates come in discrete steps defined by

$$v_{\text{est}} = \frac{d_a}{\ell T_s \hat{i}}. \tag{3.13}$$

As an illustration, consider the following set of parameters: $v_{\min} = 0.1$ m/s, $v_{\max} = 15$ m/s, $T_s = 224$ μs, $\hat{i} = \hat{i}_{\max} = 14$, $d_a = 0.05$ m, and $N_{\mathbb{A}} = 150$. The finite set of speeds detectable with this parameter combination are 15.94 m/s, 7.97 m/s, 5.31 m/s, …, 0.106 m/s – corresponding to $\ell = 1, 2, 3, \ldots, N_{\mathbb{A}}$. It is also clear from this that the performance of the speed estimator is poor in the vicinity of $v_{\max}$ with $\hat{i} = \hat{i}_{\max}$. This is due to the fact that the CRE estimation interval is then not short enough to provide the time resolution needed at higher speeds. Table 3.1 compares the effect of using $\hat{i} = \hat{i}_{\max}$ and $\hat{i} = \hat{i}_{\min}$ on the speed estimates obtainable.

The tabulated data indicates that a longer estimation interval is necessary for speeds in the vicinity of $v_{\min}$ while higher speeds require shorter estimation intervals for more accurate estimates. However, the improved accuracy in the vicinity of $v_{\max}$ comes at the cost of the minimum speed detectable – it increases to 1.4881ms^{-1} when $\hat{i} = \hat{i}_{\min}$. It is clear from (3.9) that the adaptive parameter μ strives to provide the most suitable $\hat{i}$ for a given database size ($N_{\mathbb{A}}$) – higher values of v_{est} result in higher values of μ which in turn lead to a reduction in the CRE estimation interval. The opposite is true for lower values of v_{est} where μ is adjusted to yield a longer estimation interval.

While it is obvious that the setting $\hat{i} = \hat{i}_{\min}$ yields the finest time resolution possible and hence

$\hat{i} = \hat{i}_{\max} = 14$		$\hat{i} = \hat{i}_{\min} = 1$	
ℓ	Speed (m/s)	ℓ	Speed (m/s)
1	15.944	1	223.21
2	7.9719	…	…
3	5.3146	14	15.944
4	3.9860	15	14.881
…	…	16	13.951
148	0.10773	…	…
149	0.10701	149	1.4981
150	0.10629	150	1.4881

Table 3.1: *Effect of the CRE estimation interval on speeds detectable.*

the most accurate results, it is inefficient in practice as the number of past CRE estimates that must be stored in memory to detect $v_{\min}$ is prohibitively large (e.g. $N_{\mathbb{A}}$ is 4465 for a DVB-T receiver with $T_s = 224$ μs, $d_a = 10$ cm, $v_{\min} = 0.1$ m/s). Dynamically adjusting $\hat{i}$ to the estimated speed allows the RRFSM algorithm to be more memory efficient ($N_{\mathbb{A}} = 150$ for the aforementioned DVB-T receiver with $v_{\min} = 0.1$ m/s and $v_{\max} = 15$ m/s).

Table 3.1 also shows that with the initial value of $\hat{i} = 14$ and hence $\mu = 1$, a true speed of 15 m/s is most likely to be detected as 15.944 m/s. At the absence of the parameter ϵ, μ will always remain at that initial value (since the condition $v_{\text{est}} \leq (v_{\max} + \epsilon)$ in (3.9) would not be satisfied) and hence the speed will continue to be estimated as 15.944ms^{-1}. With $\epsilon = 1$, however, that condition will be satisfied and μ will be increased accordingly – reducing the estimation interval and yielding an estimate that is more precise than the last.

Since is T_s is usually fixed for a given system, the minimum time resolution available is also fixed. As a result, the speeds that can be detected using the RRFSM approach not only depend on $\hat{i}$ but also d_a – as indicated by (3.13). An illustration is provided in Figure 3.4 for three different antenna separation distances: 1 cm, 5 cm and 10 cm. The plot shows that for a given antenna separation distance, a wide ranges of true speeds can produce exactly the same estimated speed; with the effect becoming more apparent at smaller values of d_a. Even with the smallest possible CRE estimation interval ($\hat{i} = 1$) and ideal conditions, a significant mismatch is observed at higher speeds. This discrepancy between the true and estimated speed is due to the fact that when d_a is small, the total number of discrete speed levels obtainable using (3.13) is dominated by the minimum time resolution T_s. Therefore, for a given RF

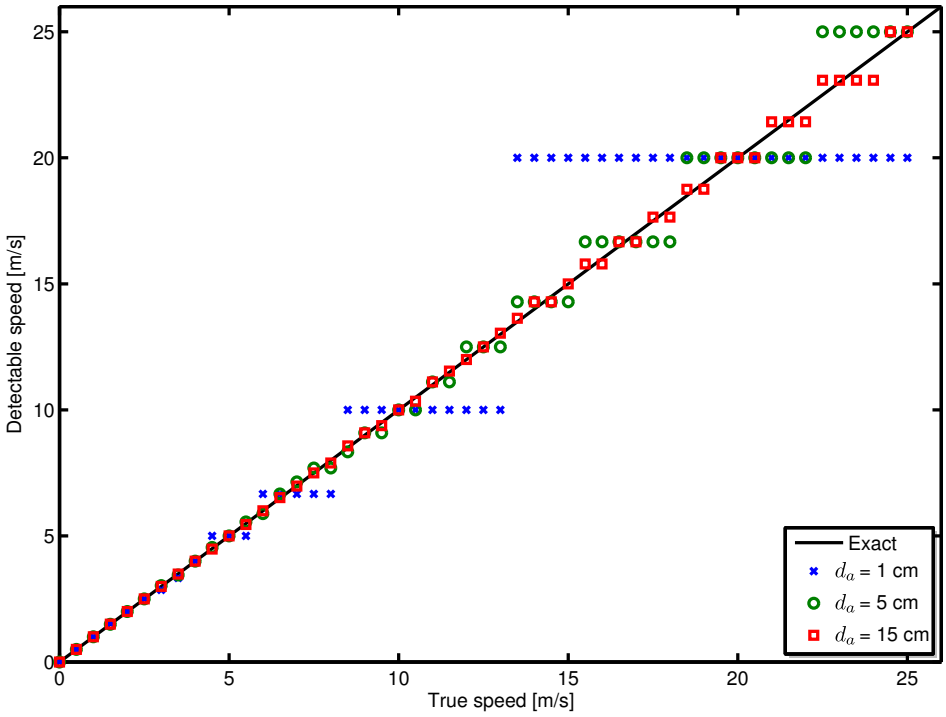


Figure 3.4: *Speeds detectable using the RRFSM based algorithm under ideal circumstances. The results are obtained from* (3.13) *for* $T_s = 224$ *µs,* $\hat{i} = 1$ *and* $d_a = 1$ *cm,* 5 *cm and* 10 *cm. It can be seen that the precision drops considerably at higher speeds for smaller antenna separation distances.*

receiver system, the precision of the speed estimates obtained may be improved by increasing the antenna separation distance.

Operation of the RRFSM algorithm for a receiver equipped with a pair of antennae aligned in the direction of motion is illustrated by means of the flowchart in Figure 3.5. The CREs from both antennae are acquired simultaneously and stored in memory along with the current time. It is necessary to store the time associated with each $H_{\mathbb{A}}$ in order to overcome the difficulty in establishing the time delay between a pair of CREs that otherwise arises from an adaptive and therefore variable channel estimation rate. It has been stated previously that the resolution used to store the timestamps has a significant impact on the accuracy of the algorithm – with a higher resolution (finer increments) generally providing better speed estimates. Although an infinitesimally small time interval between CRE estimates is desirable, the resolution achievable in practice is limited by the wireless standard used by the receiver. For an orthogonal frequency division multiplexing (OFDM) based system such as DVB-T, the minimum resolution possible is equivalent to the duration of a single OFDM symbol.

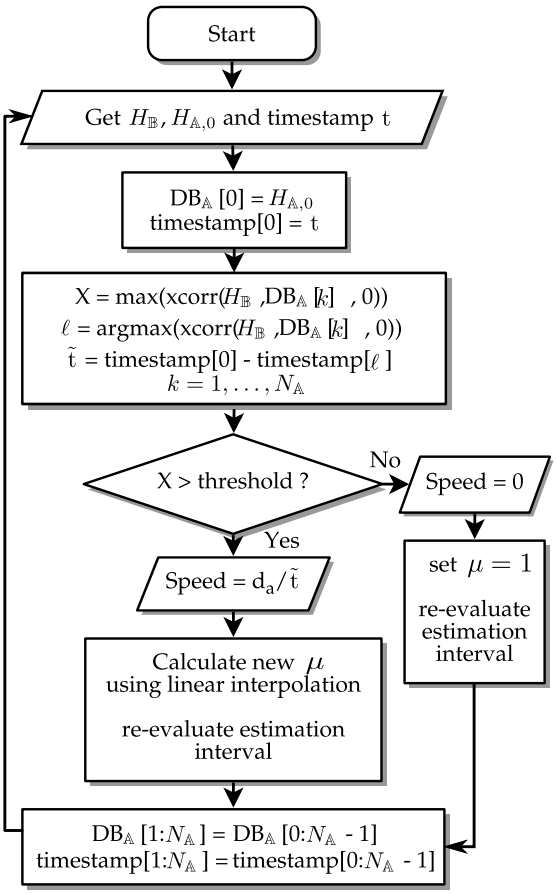


Figure 3.5: *Flowchart illustrating the RRFSM approach to speed estimation. $DB_{\mathbb{A}}$ is the CRE database and t is the time when $H_{\mathbb{A},0}$ and $H_{\mathbb{B}}$ are acquired. The function* `xcorr(`$\vec{x}$`,`$\vec{y}$`,`l`)` *provides the normalised cross correlation product of the vectors $\vec{x}$ and $\vec{y}$ at a lag of l; i.e. it performs the operations necessary to compute the parenthesised expression in* (3.5).

Once the CREs and the associated timestamps have been recorded, a normalised cross correlation between $H_{\mathbb{B}}$ and the database of CREs from $\mathbb{A}$ as prescribed by (3.5) yields a maximum correlation value at some database index ℓ. The index is used to retrieve the timestamp associated with $H_{\mathbb{A},\ell}$ and hence estimate the time delay $\tilde{t}$. If the correlation value is larger than a predefined threshold, the speed is estimated from d_a and $\tilde{t}$; otherwise, the speed is assumed to be zero. The estimated speed is then used to adjust the adaptive parameter μ which dictates the time that must pass before the next set of CREs are obtained. The timestamp and $H_{\mathbb{A}}$ databases are moved by a single index position to accommodate the new set of estimates.

In order to test the performance of the proposed speed estimation algorithm, a simulated speed profile representing the movement of a pedestrian over a short distance is used. The results obtained are shown and discussed in detail in Section 3.3.1.

3.2.3 Estimating speed using the spatial correlation function

The RRFSM algorithm described in Section 3.2.2 produces discrete speed estimates as shown by (3.13) and therefore is of limited use when estimates across a continuous range of speeds are required. In order to overcome this limitation, it is possible to develop an alternative approach for the speed estimation algorithm that relies on a mathematical description of the spatial correlation function. Previously discussed in Section 2.1.1, the function is presented once again in the following paragraphs for convenience.

It is shown in [28, 32] that the spatial correlation function of a channel can be described in terms of three quantities collectively referred to as *multipath shape factors*. The spatial correlation, $\rho(d,\theta)$, between signal envelopes separated by a distance d, along some azimuthal direction θ is approximated by

$$\rho(d,\theta) \approx \exp\left[-23\Lambda^2\left(1+\gamma\cos\left[2\left(\theta-\theta_{\max}\right)\right]\right)\left(\frac{d}{\lambda_c}\right)^2\right] \tag{3.14}$$

where Λ , γ and $\theta_{\max}$ are the multipath shape factors known as *angular spread*, *angular constriction* and *azimuthal direction of maximum fading* respectively.

The angular spread shape factor describes the distribution of multipath components about a

single azimuthal direction. It is defined as

$$\Lambda = \sqrt{1 - \frac{|F_1|^2}{F_0^2}}, \tag{3.15}$$

where F_n is the n^{th} complex Fourier coefficient of the angular distribution of multipath power, $p(\theta)$:

$$F_n = \int_0^{2\pi} p(\theta) \exp(-jn\theta)\, d\theta. \tag{3.16}$$

Λ ranges from zero to one, with zero corresponding the case where a single multipath component arrives from a single direction. When the power distribution lacks a dominant azimuthal direction, Λ is unity.

Angular constriction is a concept that is similar to angular spread. It describes the distribution of multipath components about two azimuthal directions instead of just one. It is defined as

$$\gamma = \frac{|F_0 F_2 - F_1^2|}{F_0^2 - |F_1|^2}. \tag{3.17}$$

As stated before, γ varies between zero and one. It is unity when there are exactly two multipath components arriving from two different directions. However, when neither of the two directions dominate, γ is zero.

Finally, the azimuthal direction of maximum fading is simply defined as

$$\theta_{\max} = \frac{1}{2} \arg\left(F_0 F_2 - F_1^2\right). \tag{3.18}$$

Durgin and Rappaport arrive at the expressions for the multipath shape factors by means of a complex Fourier expansion of the second moment of the envelope fading process. Details of the derivation are available in [32, Appendix I].

If the motion of the RF receiver equipped speed estimation device is once again assumed to be along a single constant azimuthal direction θ_0, it is possible to re-write (3.14) as:

$$\rho(d) \approx \exp\left[-23\Lambda^2 \left(1 + \gamma \cos\left[2\left(\theta_0 - \theta_{\max}\right)\right]\right) \left(\frac{d}{\lambda_c}\right)^2\right] \tag{3.19}$$

where $\rho(d)$ signifies the fact that the correlation function is now valid for a predefined azimuthal direction only. Additionally, if $p(\theta)$ is assumed to be time invariant, Λ, γ, and $\theta_{\max}$ are all constant and (3.19) can be further simplified to:

$$\rho(d) \approx \exp\left[-23K\left(\frac{d}{\lambda_c}\right)^2\right], \tag{3.20}$$

where the constant K is defined as

$$K = \Lambda^2\left(1 + \gamma\cos\left[2\left(\theta_0 - \theta_{\max}\right)\right]\right). \tag{3.21}$$

Therefore, using the inverse of (3.20):

$$d \approx \lambda_c\sqrt{-\frac{\ln(\rho)}{23K}}, \tag{3.22}$$

it is possible to estimate the distance d between two signal envelopes (and hence CREs) given the mutual correlation value ρ, the signal wavelength λ_c, and the channel dependent constant K. If the time delay Δt between the CREs is known, an estimate of the speed can be obtained from the estimate of d using (3.3) by substituting d for d_a and Δt for $\tilde{t}$:

$$\left|\vec{v}\right| = \frac{d}{\Delta t} \approx \frac{\lambda_c\sqrt{-\frac{\ln(\rho)}{23K}}}{\Delta t}. \tag{3.23}$$

The concept is illustrated using Figure 3.6. If the value of K is assumed to be known, it is possible to perform speed estimation using a device equipped with just a single antenna element.

Unfortunately, the parameter K is directly related to the angular distribution of multipath power ($p(\theta)$) which is highly dependent on the geometry of the radio environment. Therefore it is virtually impossible to obtain an analytical expression for the instantaneous value of K. Hence it must be estimated before the proposed speed estimation algorithm can be applied. Simple manipulation of (3.22) provides an approximation

$$K \approx -\frac{\ln(\rho)}{23}\left(\frac{\lambda_c}{d}\right)^2 \tag{3.24}$$

in terms of the correlation ρ and the distance d. At this point, it is possible to exploit the

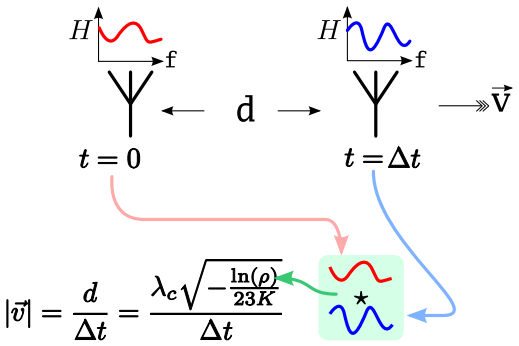


Figure 3.6: *Speed estimation using the spatial correlation function described by* (3.20)*. Knowledge of the channel parameters (jointly represented by K) and the correlation ρ between two CREs acquired Δt seconds apart allows the distance traversed, d, to be estimated using* (3.22)*. The speed, $|\vec{v}|$, can then be estimated using $|\vec{v}| = \frac{d}{\Delta t}$.*

speed estimation scheme described in Section 3.2.2. While evaluating (3.5), $H_{\mathbb{B}}$ is correlated against all the $H_{\mathbb{A}}$ vectors stored in the CRE database to yield a set of correlation factors $\left[\rho_1, \rho_2, \rho_3, \ldots, \rho_{N_{\mathbb{A}}}\right]$. This set of values can be used together with the rough speed estimate $v_{\text{est}}^{(\text{RRFSM})}$ to map each ρ_k to a distance d_k using

$$d_k = d_a - \left(\Delta t_k \cdot v_{\text{est}}^{(\text{RRFSM})}\right);\ k = k_{\min}, k_{\min}+1, \ldots, k_{\max} \tag{3.25}$$

where d_a is the antenna separation distance, Δt_k is the time delay associated with ρ_k and $k_{\min}$ and $k_{\max}$ define a subset of the coefficients that meet a suitability criterion described later in this section. Given this subset of correlation coefficients and their associated distance estimates, a set of estimates for K can be obtained using (3.24). The final estimate, K', is then obtained by taking the arithmetic mean over that set:

$$K' = \frac{1}{k_{\max} - k_{\min} + 1} \sum_{k=k_{\min}}^{k_{\max}} K_k. \tag{3.26}$$

Since K' is an estimate of the true value of K, it is necessary to evaluate its accuracy. Given the procedure employed to obtain the estimate, the most likely source of errors are the correlation coefficients ρ_k and $v_{\text{est}}^{(\text{RRFSM})}$. As the estimated speed already represents the best-effort of the algorithm in Section 3.2.2, an attempt is made to pick a subset of the correlation values that lead to the least error in the estimate of K. The relative error e_K in the estimate of K due

to an erroneous correlation value $\widehat{\rho}$ is given by:

$$e_K = \left| \frac{K' - K}{K} \right| = \left| \frac{\ln(\widehat{\rho})}{\ln\left(\widehat{\rho} - e_\rho\right)} - 1 \right|, \tag{3.27}$$

where e_ρ is the error in $\widehat{\rho}$. The derivation is provided in Appendix A.1.

As it is not possible to determine the absolute error present in each correlation coefficient computed, e_ρ is assumed to be the same for all coefficients. Figure 3.7 shows plots of (3.27) for $e_\rho = -0.05, -0.02$ and $+0.05$. Although it appears that errors in the coefficients can lead to relatively large errors in K – the minimum e_K for $e_\rho = 0.05$ is approximately 0.15 – the error remains close to the minimum over a wide range of values of the correlation coefficient; as indicated by the "valley" in each of the plots. Figure 3.7 indicates that this range is approximately between $\rho = 0.2$ and 0.7 for e_ρ up to 5%. At higher values of e_ρ the range becomes smaller, with larger values of ρ leading to higher e_K than smaller values of ρ.

The reason for the existence of a range of values of ρ that lead to lower e_K is clear from Figure 3.8 where the behaviour of the spatial correlation function is investigated under three different channel conditions represented by $K = 0.33, 0.50$ and 1.00. The figure shows that the rate of change of the distance with respect to the correlation coefficient is the smallest when ρ is between 0.2 to 0.7 – implying that a large uncertainty in the value of ρ leads to relatively little error in the value of d, thereby minimising the relative error in K. As a result, $k_{\min}$ and $k_{\max}$ in (3.25) are chosen such that only coefficients with values within a certain range $[\rho_{\min}, \rho_{\max}]$ are considered for the estimation of K. It is important to minimise this error since the accuracy of the proposed speed estimation algorithm depends only on K and Δt which is assumed to be known exactly.

Once an estimate of K is available, subsequent correlation coefficients ρ_k computed at intervals of Δt, where $\Delta t \ll T_c$, can be used to obtain the associated set of distances d_k using (3.22). The corresponding set of speed estimates can then be obtained using

$$v_{\text{est},k} = \frac{d_k}{k \times \Delta t};\ k = k_{\min}, k_{\min} + 1, \ldots, k_{\max}. \tag{3.28}$$

The final speed estimate is then obtained by taking the arithmetic mean of the set of estimated

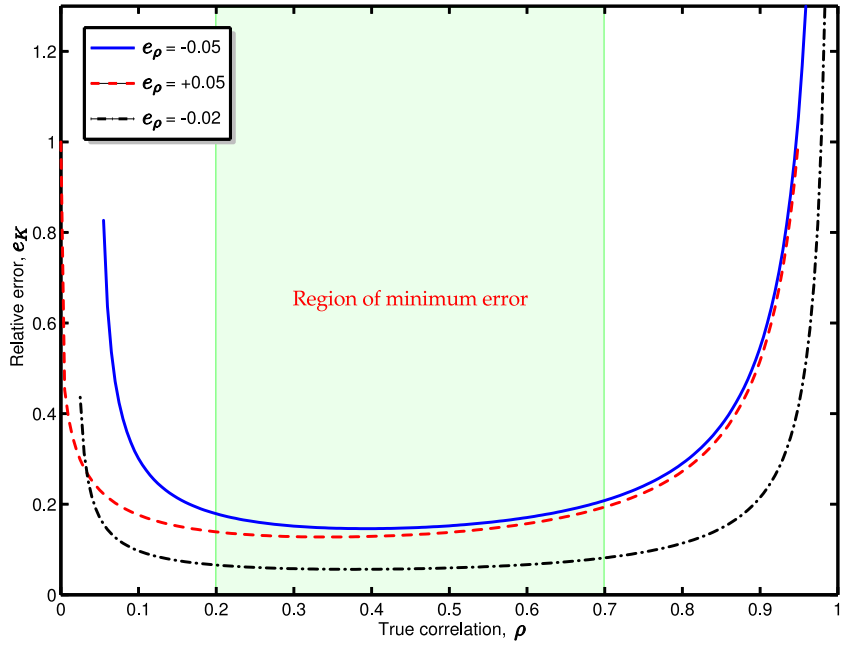


Figure 3.7: *Relative error in the estimate of K for $e_\rho = -0.05, -0.02$ and $+0.05$. The error is at its lowest (below 30%) when the true correlation is between 0.2 and 0.7.*

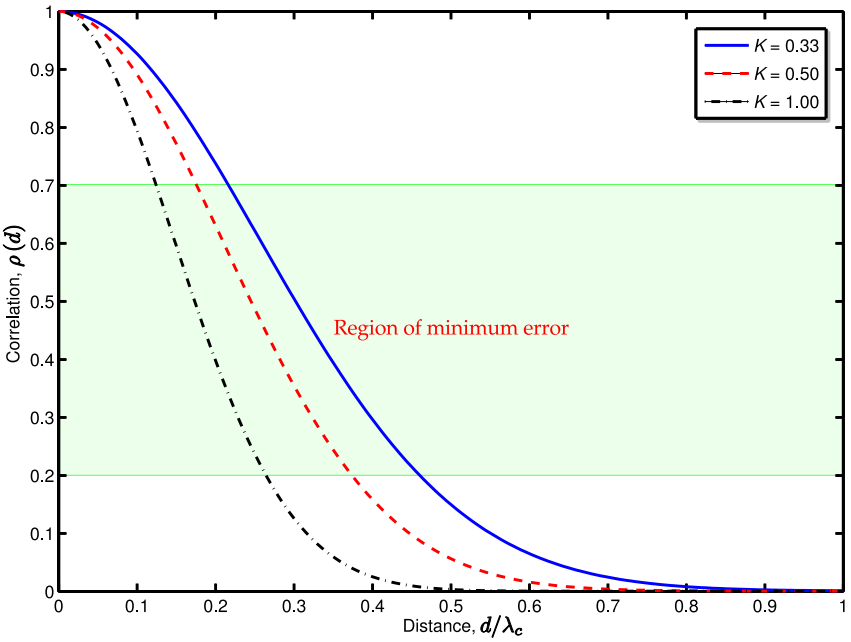


Figure 3.8: *Plot of the spatial correlation as a function of distance for various channel conditions. Between $\rho = 0.7$ and 0.2 the gradient is at its steepest and therefore the rate of change of the distance with respect to the correlation coefficient is at its lowest. This implies that large errors in the value of ρ leads to relatively small errors in the estimate of d.*

speeds:

$$v_{\text{est}} = \frac{1}{k_{\max} - k_{\min} + 1} \sum_{k=k_{\min}}^{k_{\max}} v_{\text{est},k}. \tag{3.29}$$

For the proposed speed estimation scheme to remain accurate and effective, the estimate of K must be periodically updated. A flowchart of the algorithm is shown in Figure 3.9. Naturally, the first step of the algorithm is to estimate the channel dependent parameter K since it is required by the spatial correlation function. Then a reference CRE, H_0, is acquired against which N_ρ of the subsequent CREs are compared. When a sufficient number of correlation coefficients (N_ρ) have been obtained, those satisfying the admissibility criterion defined by $\rho_{\min}$ and $\rho_{\max}$ are used to obtain an estimate of the speed. Since the channel is time varying in practice, a fresh estimate of K must be obtained at the start of each iteration of the proposed speed estimation scheme.

Once again, simulated pedestrian movement is used to investigate the effectiveness of the algorithm. The results obtained are presented and discussed in Section 3.3.2.

3.3 Results

In order to test the performance of the proposed algorithms, MATLAB based implementations are used to estimate the speed profile illustrated in Figure 3.10. It represents the behaviour of a pedestrian over a period of 60 seconds. Starting from rest, the pedestrian moves at an average speed of 1.5 m/s while crossing a road. On reaching a traffic-free zone, the speed drops to 0.5 m/s. This speed is maintained until the second road where the average speed increases to 2 m/s for a quick crossing. On reaching the second pedestrian zone, the average speed once again reduces to 0.5 m/s before the pedestrian finally stops at the destination.

For simulation purposes, the RF signal source is assumed to be a DVB-T transmitter with a carrier frequency of 474 MHz and 1705 subcarriers (2K mode of DVB-T). The spatiotemporal multipath channel is Rayleigh fading with relative gain (α_l), phase offsets (ϕ_l) and time delays (τ_l) obtained from [31]. For simplicity, the AoAs (θ_l) are assumed to be time-invariant and uniformly distributed in $[0, 2\pi]$. Furthermore, it is also assumed that the channel response estimates are error free and displacement is along a single dimension only. For the RRFSM based algorithm, the correlation threshold used is 0.95. The use of such a high threshold is justified since the CRE estimates are assumed to be perfectly known and error-free.

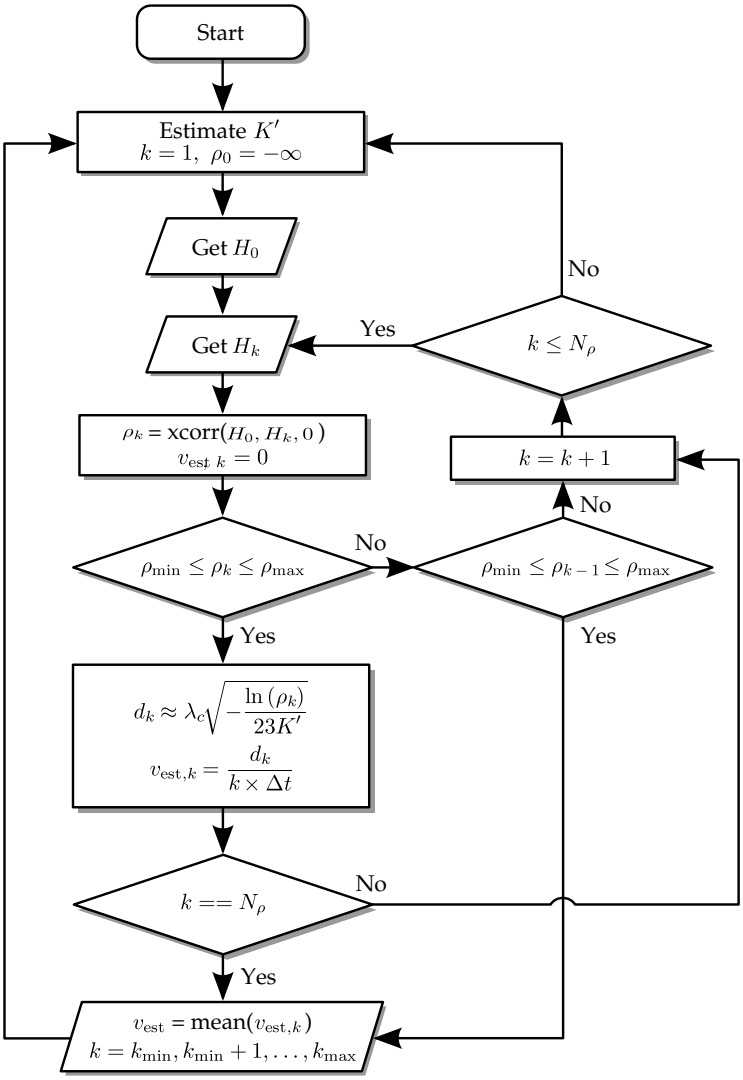


Figure 3.9: *Speed estimation algorithm utilising spatial correlation. H_k represents the k^{th} CRE. $\rho_{\min}$ and $\rho_{\max}$ define the coefficient range considered for speed estimation. N_ρ is the total number of CREs to be used and serves as a stopping condition for each iteration of the algorithm.*

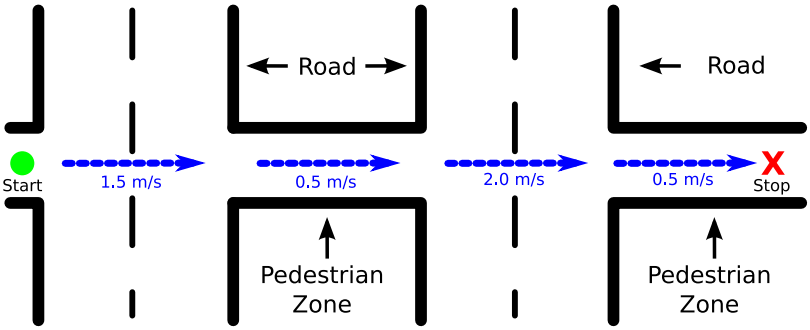


Figure 3.10: *Illustration of the behaviour of a pedestrian over roads and traffic-free zones.*

3.3.1 Method utilising relative RF signature matching

The result of using the RRFSM algorithm for estimating the speed of the pedestrian illustrated in Figure 3.10 is shown in Figure 3.11. The plot shows the estimates obtained using antenna separation distances (d_a) of 10 cm and 30 cm. A moving average filter is applied to the raw estimates in order to smoothen the output. With an antenna separation distance of 10 cm, a good match between the real and estimated speeds is obtained. The performance is visibly poorer at the larger separation distance of 30 cm. The observed difference is largely due to the time-varying nature of the channel. As an illustration, consider the case where the device is moving at a constant speed of 0.5 m/s. With $d_a = 30$ cm, the time required by $\mathbb{B}$ to traverse the inter-antenna separation distance is 0.6 s; however, at that speed, the coherence time of the channel is only 0.23 s. Since the travel time required is then larger than the coherence time of the channel, the relevant CREs are no longer strongly correlated and a reliable speed estimate cannot be obtained. It should also be noted that for a given speed, the maximum Doppler spread and hence the coherence time of the channel is dictated by the carrier frequency of the wireless system in use – with a higher f_c leading to a shorter T_C. Therefore, receivers operating at a higher frequency will require a shorter antenna separation distance to maintain an acceptable level of performance.

The estimated speed profiles show that the algorithm tends to momentarily lose track of the speed when the device undergoes a sudden change in speed – i.e. while accelerating. The reason for this behaviour is a drastic change in the channel model itself. At constant speeds (zero acceleration), the channel response is linear a function time, t. However, when the device is experiencing acceleration, the response changes from being a function of linear time to quadratic time (t^2) as shown in Appendix A.2. As a result, any CRE recorded at some time t_0 while accelerating is in fact similar to a CRE at time t_0^2. This fundamental change in the CREs acquired invalidates the fundamental principle upon which the speed estimator is designed and hence leads to a disruption in the operation of the algorithm. Once the speed stabilises, however, the algorithm is quick to recover and continue tracking the real speed with a low margin of error.

Figure 3.12 shows CDFs of the relative errors in estimating speeds of 0.5 m/s, 1.5 m/s and 2.0 m/s at an antenna separation distance of 10 cm. The percentage error values are calculated

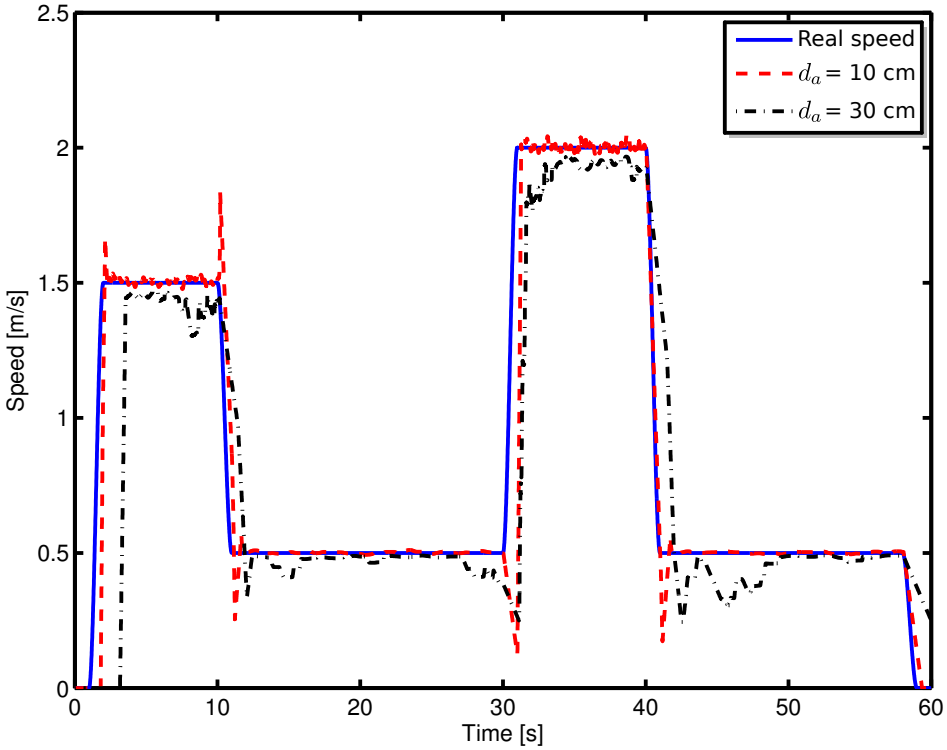


Figure 3.11: *Simulation of the relative RF signature matching algorithm for antenna separation distances (d_a) of* 0.10 *m and* 0.30 *m. The estimates are passed through a moving average filter for smoothing. The speed profile emulates the behaviour of a pedestrian as illustrated in Figure 3.10.*

using

$$\text{Error} = \left(\frac{v_{\text{est}} - v_{\text{real}}}{v_{\text{real}}} \right) 100\%. \tag{3.30}$$

From the figure it is evident that the performance of the algorithm is independent of the actual speed of the device for a given d_a; as long as the CREs can be estimated with sufficient frequency and the antenna separation distance is small enough to avoid errors due to the coherence time of the channel. On average, the relative error is approximately 2.67% with a standard deviation of 5%. The quantisation of detectable speeds due to time resolution available (Section 3.2.2) also leads to discretisation of the relative error. This is the reason behind the stair-like CDF.

It is clear from CDF that the average error is positive – indicating a tendency to overestimate the speed. Since d_a is fixed, this implies that the time delay is generally underestimated. This is once again due to the time-varying nature of the channel and a difference in the rate at which the spatial and temporal components of the channel evolve. As in illustration of the effect, Figure 3.13 shows how the spatial and temporal correlation changes for a device with $d_a = 10$

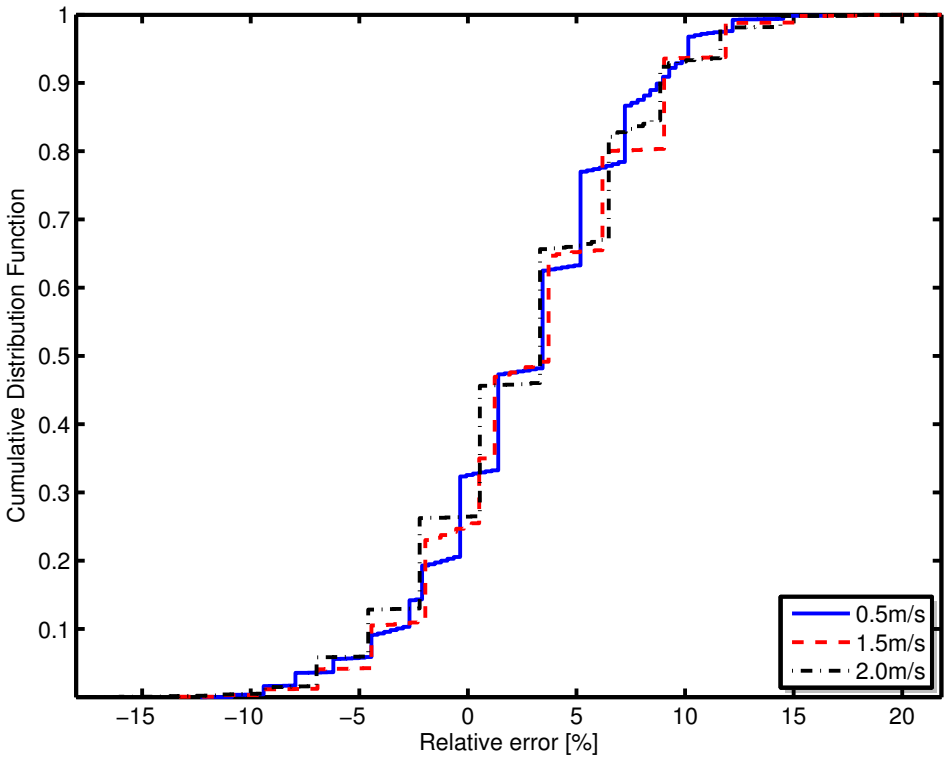


Figure 3.12: *CDF of relative errors in estimating speeds of* 0.5 *m/s,* 1.5 *m/s and* 2.0 *m/s at an antenna separation distance of* 10 *cm. The issue of limited time resolution described in Section 3.2.2 manifests itself in the form of discrete steps in the error CDF.*

cm moving at a speed of 15 m/s. Although a time axis, t, is not explicitly shown, it is related to the displacement axis, x, by the simple equation $t = \frac{x}{15} + t_0$, where t_0 is some constant. It can be seen that as $\mathbb{B}$ approaches the desired spatial location, x_A, the spatial correlation rapidly increases while the temporal correlation gradually decreases in accordance with the channel's coherence time. As a result, the joint maximum of the correlation functions is not at the desired location but shortly before it – leading to an underestimate of the time delay and hence an overestimate of the speed.

The results shown appear to indicate that a smaller antenna separation distance may lead to smaller errors by counteracting the degradation of CRE correlation caused by the coherence time of the channel. However, it has been shown previously in Section 3.2.2 (and Figure 3.4) that smaller values of d_a also lead to a rapid loss of *precision* in the speed estimates. As a result of this conflict, there must be a trade-off between the accuracy of the estimates and overall performance.

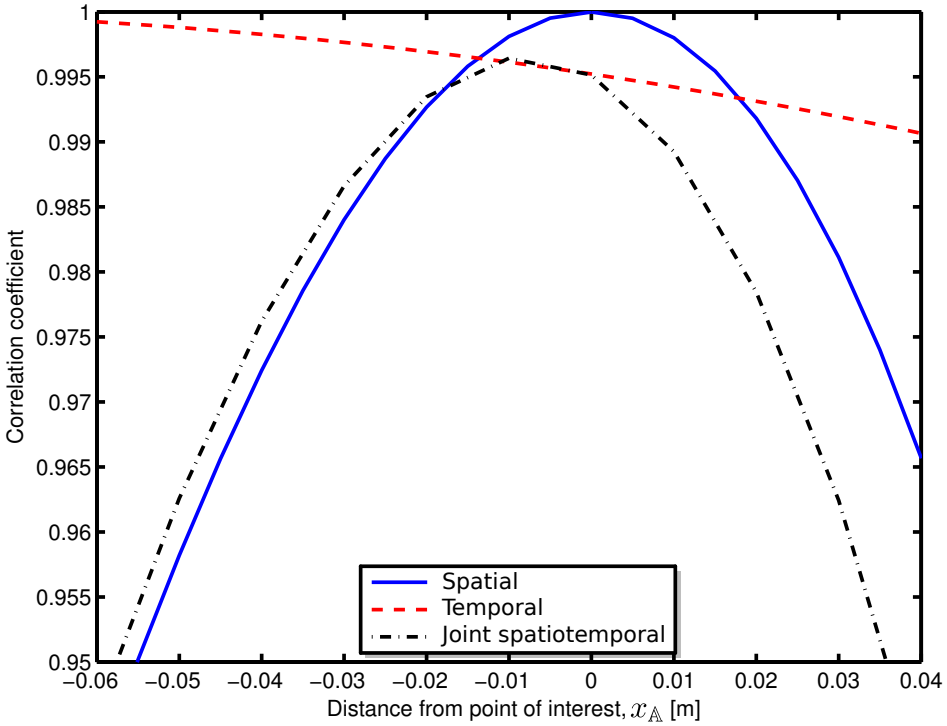


Figure 3.13: *Spatial and temporal correlation coefficients for a device with $d_a = 10$ cm moving at a speed of 15 m/s. As $\mathbb{B}$ approaches the position previously occupied by $\mathbb{A}$ (the spatial coordinate of interest, $x_{\mathbb{A}}$), the spatial correlation increases while the temporal correlation decreases. This leads to a joint maximum at a spatial coordinate shortly before $x_{\mathbb{A}}$ – leading to an underestimate of the time delay and hence an overestimate of the speed.*

3.3.2 Method utilising spatial correlation function

The pedestrian speed profile (Figure 3.10) is utilised once again to provide a common basis of comparison between the proposed speed estimation algorithms. It is assumed that an estimate of the channel dependent parameter K is available for each instance of the simulation. The estimates obtained from the proposed speed estimation scheme under various channel conditions represented by $K = 0.33, 0.50$ and 1 are shown in Figure 3.14. It is further assumed that perfect knowledge of the channel is available and hence the correlation coefficients ρ and the estimates of the parameter K are error free. From the plot, it is immediately clear that there is a near-perfect match between the real speed profile and that estimated by the algorithm – showing that the method described works exactly as expected under ideal circumstances. The lack of any substantial difference between the estimates indicates that the results are not affected by the channel condition simulated. This is due to the assumption that both ρ and K are error free.

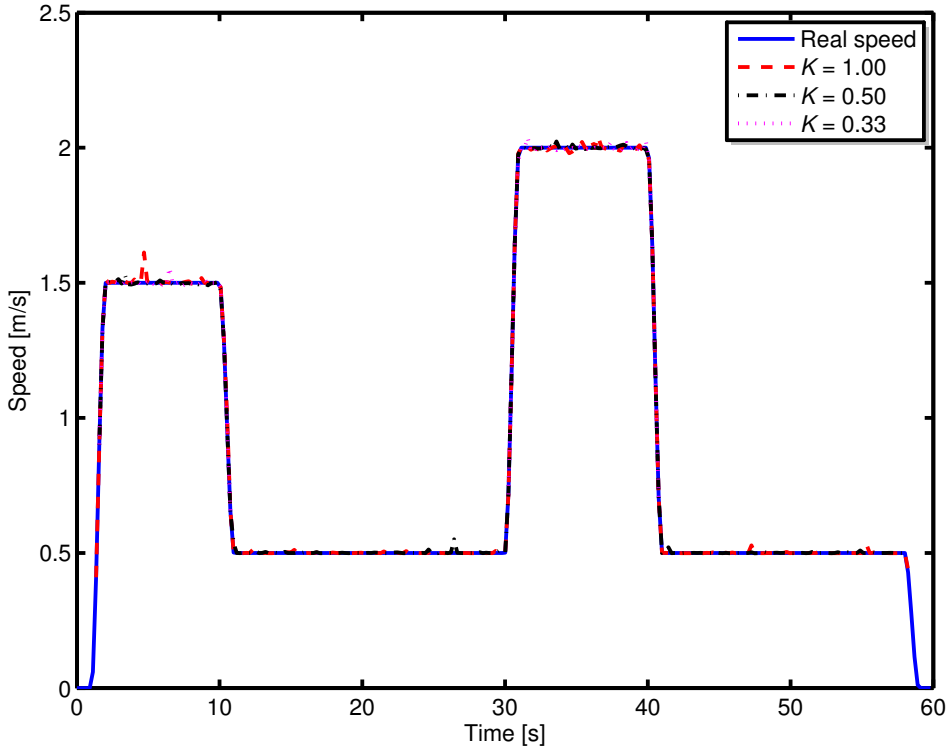


Figure 3.14: *Estimating the pedestrian speed profile in Figure 3.10 using the spatial correlation function based algorithm. Different channel conditions are simulated using* $K = 0.33, 0.5$ *and* 1. *The estimates obtained are nearly exact since perfect knowledge of the channel conditions is assumed.*

Since the assumption of a perfect knowledge of channel conditions is unrealistic, the simulation is repeated for a scenario where uncertainties exist in the estimate of the channel dependent parameter K. Due to the high level of complexity associated with implementing a statistical channel model that supports empirical measurements of the angular power distribution and hence shape factors, the estimator for K is not included in the simulator. Instead, a uniformly distributed random error between $\pm 10\%$ is added to the true value of K to model uncertainties in the parameter. The estimates obtained are shown in Figure 3.15. Once again, a set of three distinct channel conditions are emulated by setting $K = 0.33, 0.50$ and 1. The degradation in speed estimation performance due to the errors in K is immediately apparent in the form of increased noise. However it should be noted that although the estimates are noisy, on average it still shows a good match against the real speed profile.

The plots (Figure 3.14 and Figure 3.15) also reveal that no speed estimates are obtained below approximately 0.3 m/s. In order to further explore this observation, comparisons of the minimum speed detectable using the algorithm under various (exact) values of the parameter K

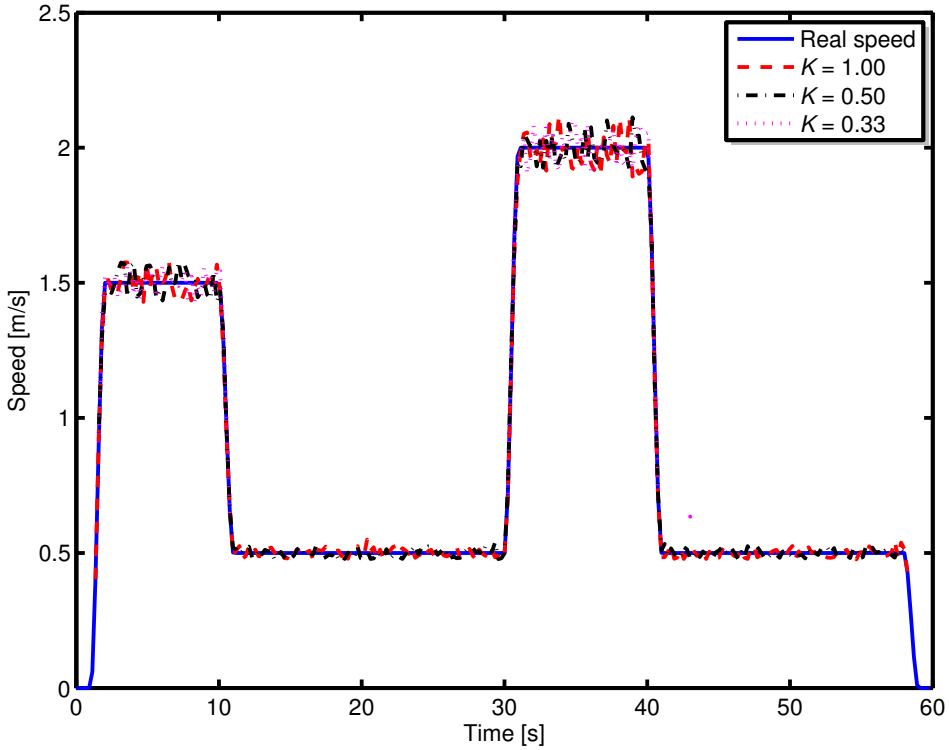


Figure 3.15: *Simulation of the spatial correlation algorithm for various channel conditions. The estimate of the parameter K is not exact and contains a random relative error uniformly distributed between* $\pm 10\%$.

are performed and the results obtained are shown in Figure 3.16. The general trend visible in the figure is that the higher the value of K, the lower is the speed detectable. This relationship that is observed between the speed and the channel dependent parameter can be explained by exploring the connections that exist between K, $\rho_{\max}$ and the time length over which the set of correlation coefficients are acquired.

The implementation of the algorithm is such that the time over which successive CREs are acquired is fixed – regardless of the speed. As a result, at very low speeds, all of the recorded correlation coefficients may be very high and therefore fail to satisfy the condition $\rho_k \leq \rho_{\max}$. Since no coefficients are then available, it is impossible to obtain an estimate of the speed. For a fixed value of d, the spatial correlation function described by (3.20) shows that $\rho \approx \exp[-CK]$ where C is a constant greater than zero. This implies that at a given distance, d, the correlation coefficient, ρ, is small for a large value of K and vice versa. Consequently, for a given speed, it maybe possible to obtain an estimate from a channel with a large value of K since a number of the correlation coefficients acquired may indeed be smaller than $\rho_{\max}$.

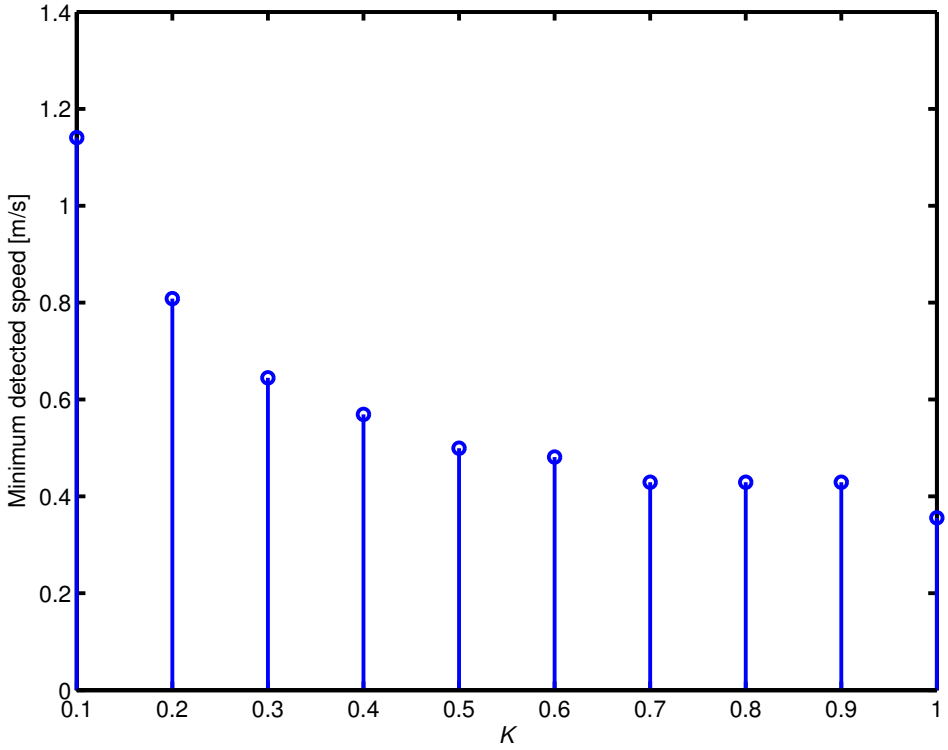


Figure 3.16: *Minimum speed detectable using the spatial correlation function based speed estimation algorithm under various channel conditions.*

It is clear that the minimum speed detectable ultimately depends on implementation specific parameters such as $\rho_{\max}$ and the time over which the set of correlation coefficients are recorded. Since the acquisition time cannot be arbitrarily increased due to channel coherence time constraints, the only option available is to relax the bounds on admissible values of correlation coefficients. With a larger $\rho_{\max}$, estimates (perhaps with a higher margin of error – see Section 3.2.3) may still be obtainable at low speeds.

To investigate the error performance of the algorithm, simulations are performed at typical pedestrian speeds of 1.0 m/s, 1.5 m/s and 2.0 m/s. The percentage relative error in the estimates (where available) are once again computed using (3.30). Figure 3.17 shows the CDF of the errors when the estimate of the parameter K contains a random error uniformly distributed between $\pm 10\%$. As the error CDFs are very similar to one another, it is reasonable to conclude that the relative error in the estimates is independent of both the speed and the actual channel characteristics for this particular implementation. It can be seen that even with a 10% uncertainty in the value of the parameter K, the average error in the speed estimates is less than 0.15% with a standard deviation of approximately 3%.

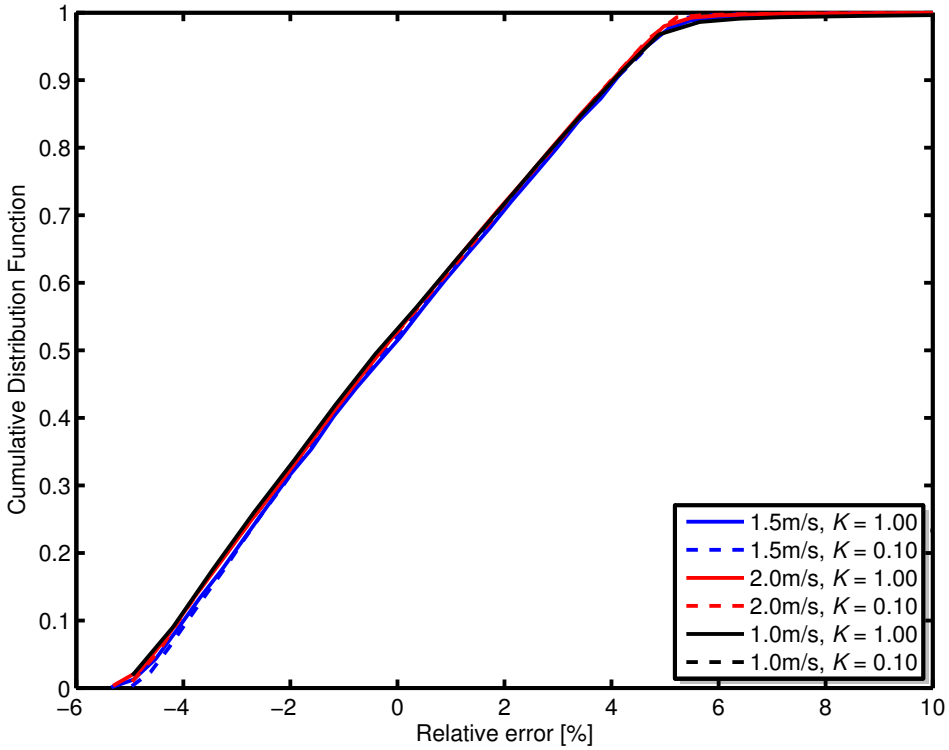


Figure 3.17: *CDF of relative errors in estimating speeds of* 1.0 *m/s,* 1.5 *m/s and* 2.0 *m/s using the spatial correlation function based method. The estimate of the parameter K contains a random error uniformly distributed between* ±10%.

It is clear from the error CDFs provided in Figure 3.12 and Figure 3.17 that the spatial correlation function based method can provide better performance than the RRFSM based method when the channel dependent parameter K can be reliably estimated. Due to its superior performance, it is an ideal candidate for use in position estimation. The effect of errors in the speed estimates for a device moving according to the model in Figure 3.10 on the estimate of the displacement is shown in 3.18. Once again, the parameter K is assumed to contain an error uniformly distributed between ±10%. After 60s, the error in the displacement estimate is only 24.5 cm. The reason for such a low error in the estimate is the fact that overestimates and underestimates of the speed tend to cancel out (as seen in Figure 3.15) – resulting in a mean error that is very close to zero. This is particularly beneficial in context of dead reckoning based method of position estimation where the accumulation of error is a common problem.

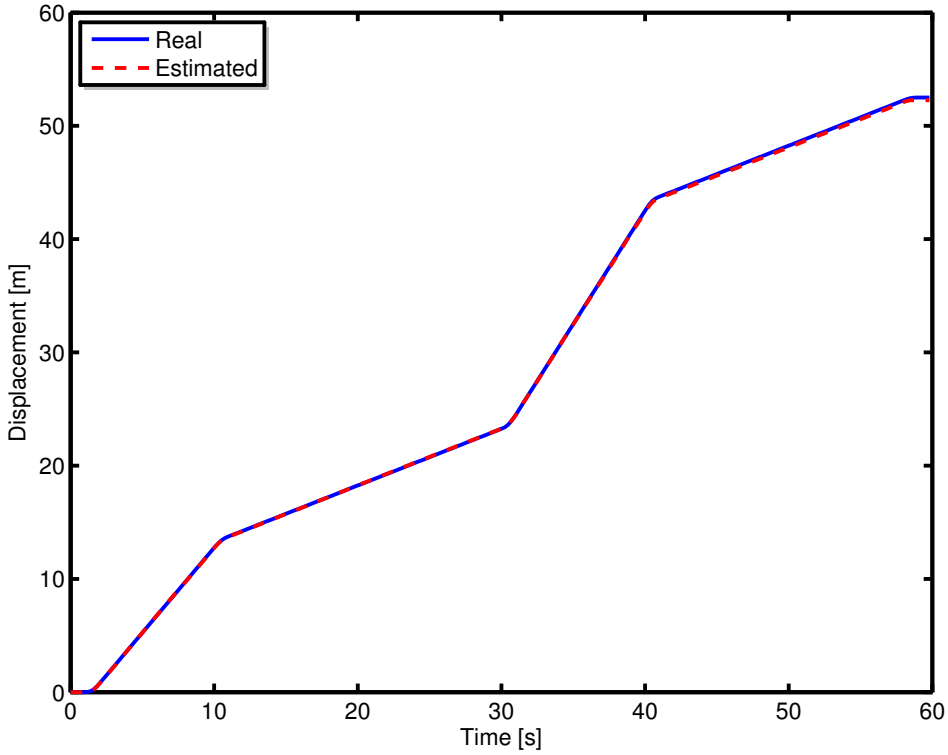


Figure 3.18: *Displacement (real and estimated) of a device moving according to Figure 3.10 using the spatial correlation function based algorithm with a* 10% *uncertainty in the parameter K. Despite the errors in K, the estimate is still excellent (error of* 24.5*cm after* 60*s) due to the fact that overestimates and underestimates in the speed cancel each other (as seen in Figure 3.15).*

3.4 Chapter summary

A pair of novel speed estimation algorithms exploiting spatiotemporal correlation properties of the channel are presented in this chapter. If in addition to the speed estimates, heading information is available (e.g. through an electronic compass), indoor pedestrian dead reckoning can be performed for position estimation – allowing mobile navigation in places where GPS otherwise fails to work.

It is demonstrated that the RF signature matching algorithm based on two antennae separated by a known distance provides speed estimation errors of less than 2.67% on average at speeds typical of pedestrian entities. A by-product of this algorithm is a set of correlation coefficients which can be used to characterises the propagation environment by means of channel dependent metric. This information can be exploited by the second algorithm which utilises a deterministic spatial correlation function to perform speed estimation. When accurate estimates of the channel dependent parameter are available, the algorithm can yield highly

accurate estimates of the speed (relative error of 0.15% or less on average for a pedestrian speed profile) using a single antenna element. Additional simulations also show that when the second method is used for speed and subsequently displacement estimation, the error accumulated is very low. After 60s, the estimate differs by only 24.5cm for a typical pedestrian speed profile. This indicates that the algorithm is ideal for use in dead reckoning applications where a high rate of accumulation of error is a common problem.

As no assumptions are explicitly made regarding the radio environment, the schemes are completely self-sufficient. The only requirement is a source of RF signal that leads to a fading channel at the receiver. Since the methods do not rely on the properties of any particular wireless transmission standard, any wideband system capable of effectively capturing the effect of a fading channel is usable. In this work, DVB-T is chosen for its wideband nature (8 MHz channels) and widespread availability. Software reconfigurable MIMO handheld devices are an ideal platform for the proposed algorithms since the computational complexity is low and channel estimation performed as a part of data detection can be simply recycled for speed estimation.

Chapter 4
Detection of signal anomalies

4.1 Introduction

ANOMALY detection refers to the process of locating unusual and unexpected events that may exist alongside nominal samples in a dataset. It is a process that is already utilised in large number of diverse application domains. Typical examples include the detection of: unauthorised access to computer systems [19], irregularities in vital signs such as electrocardiogram (ECG) traces [21], fraud in financial services [20], etc. An extensive survey of current anomaly detection techniques and application domains is provided in [22].

The aforementioned survey reveals that while there are a number of different approaches to solving the anomaly detection problem, the burden of complexity is shared by many. The computational effort required makes it difficult to adapt these techniques for real-time and online processing of the input signal. This is unfortunate since any algorithm employed on an interactive communications system such as a cognitive radio platform must be capable of real-time operation to maintain a seamless user experience. To overcome this challenge, two complementary anomaly detection algorithms based on simple information theoretic measures have been developed [23–25, 27]. The first method utilises Kullback-Leibler divergence (KLD) [73] while the latter uses the information content [90] of individual signal events [90]. The algorithms are easy to generalise and broadly define anomalies as events that lead to changes in the nominal probability distribution of the radio signal. As a result it is possible to employ the techniques for the detection of a wide range of disruptive events such as interference, timing errors, transmitter malfunction and so on.

KLD is a convenient and robust method of measuring the difference between two data sets in a statistical sense. Due to its versatility and general appeal, it finds use in fields as diverse as economics [91] and computational neuroscience [92]. As a statistical comparison tool, KLD can also be employed for the automatic and real-time detection of unusual (anomalous) data segments. The proposed KLD based technique utilises two data windows to perform

a statistical comparison of neighbouring segments of signals with periodic structures (e.g. systems utilising time division multiple access [93]). Since segments separated by the signal period are expected to be analogous and hence have similar statistical characteristics, any deviation can be taken to imply the presence of an anomaly. Although anomaly detection is the primary purpose of the algorithm, it can also be used to estimate the signal's timing structure with a simple and straightforward modification.

Unlike the KLD based method, the information content analysis (ICA) algorithm can also be applied to signals lacking any kind of periodic features. Information content is a quantity that is directly related to the probability of an event: the lower the probability, the higher the information content. Since anomalies are, by definition, rare (low probability), the associated information content is high. The proposed anomaly detection algorithm exploits this fact by analysing the signal for high information content events.

The proposed algorithms are described in detail in the following section (4.2). Results from applying the techniques to the test signals and measures of performance are provided in Section 4.3. Section 4.4 concludes the chapter with a summary of the contributions made.

4.2 Detection algorithms

The detection algorithms utilise Kullback-Leibler divergence and information content analysis respectively to determine the presence of anomalies. Both quantities are ultimately calculated from estimates of the statistical probabilities of events in the signal.

Given two data sets P_n and Q_n, at time n, that contain samples from domain X, it is possible to obtain empirical estimates of the associated probability mass functions (PMFs) p_n and q_n from a non-parametric model such as a histogram. Once the PMF estimates are available, the KLD between them can be calculated using [73]

$$D\left(p_n \,||\, q_n\right) = \sum_{x \in X} p_n(x) \log_2 \frac{p_n(x)}{q_n(x)}, \tag{4.1}$$

where $x \in X$. Since base-2 logarithm is used, the divergence is measured in *bits*. KLD between two PMFs is generally asymmetric: i.e. $D\left(p_n \,||\, q_n\right) \neq D\left(q_n \,||\, p_n\right)$ and the triangle inequality is not satisfied. When $p_n = q_n$, the KLD is zero; otherwise it is a positive real number ($\mathbb{R}_+$). For brevity and convenience, $D\left(p_n \,||\, q_n\right)$ will also be referred to as D_n.

KLD belongs to a class of distance measures known as *f-divergence* (or *Ali-Silvey distances*). Some of the other distance measures that belong to the same class are: variational distance (symmetric), Hellinger distance (symmetric), and Chernoff distance (generally asymmetric) [94]. While they are all equally suitable for quantifying the statistical difference between two probability distributions, KLD and variational distance are the least complex and therefore easiest to implement. Variational distance is defined as

$$\begin{aligned} V\left(p_n \,||\, q_n\right) &= \frac{1}{2}\sum_{x\in X}|p_n(x)-q_n(x)| \\ &= \frac{1}{2}||p_n-q_n||_1 \end{aligned} \tag{4.2}$$

where $||p_n-q_n||_1$ is commonly known as the $\mathscr{L}_1$ distance (L1D) between the PMFs p_n and q_n. Furthermore, KLD and L1D (and hence the variational distance) are related by the inequality [73]

$$D\left(p_n \,||\, q_n\right) \geq \frac{1}{2\ln 2}||p_n-q_n||_1^2. \tag{4.3}$$

Crucially, it states that $D\left(p_n \,||\, q_n\right)$ is bounded by $||p_n-q_n||_1^2$ and not $||p_n-q_n||_1$. It is an important distinction as it implies that for certain PMF pairs the KLD may in fact be *smaller* than the L1D. For a pair of largely dissimilar PMFs*, as is generally the case when comparing an anomalous data set against a nominal reference, larger distance magnitudes can be obtained from KLD rather than L1D. However, when both PMFs are similar (e.g. a nominal data set and the reference) it can lead to L1D values that are larger compared to KLD. Since distance measures obtained from nominal events can be considered as "noise", it is beneficial to pick the metric that minimises this noise while maximising the distance for anomalous events. For the radio frequency (RF) signals analysed in this thesis, the aforementioned behaviour is exhibited by KLD. The difference in performance between the two distance measures is illustrated by the results provided in Section 4.3 where performance of a KLD based algorithm is compared against one based on L1D. The algorithm for anomaly detection using KLD is described in Section 4.2.2.

Information content analysis is another technique based on an information theoretic quantity that can be utilised for the detection of anomalies. The amount of information, $I_n(x)$, conveyed by any discrete random event, x_n, at time n, is directly related to its probability of

*Differences that are large enough to produce a $\mathscr{L}_1$ distance of $2\ln 2$ or greater, to be precise.

occurrence, $p_n(x)$ [90]:

$$I_n(x) = -\log_2\{p_n(x)\}; \quad n = 1,2,\ldots \tag{4.4}$$

Since base-2 logarithm is used once again, information is also measured in *bits*. The equation implies that an event with a very high probability of occurrence carries very little information while a large amount of information is conveyed by the occurrence of rare events (i.e. $I_n(x) \to \infty$ as $p_n(x) \to 0$). Information is always real, positive ($\mathbb{R}_+$) and monotonically increasing with decreasing values of event probability. ICA is essentially a non-linear scaling function that favours the unusual.

4.2.1 Histogram and PMF estimation

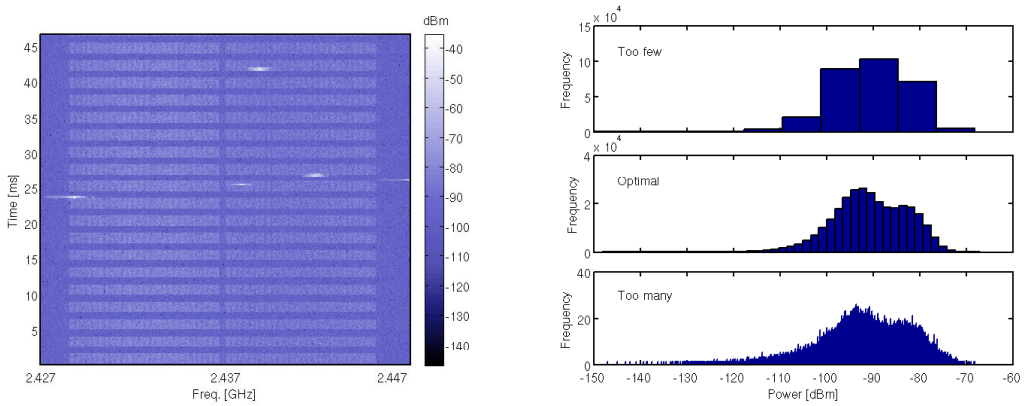
It is clear from (4.1) and (4.4) that both KLD analysis and ICA require estimation of empirical event probabilities. One approach to obtaining the necessary estimates is via event histograms. In addition to being simple to implement, histograms are non-parametric – implying that no assumptions need to be made regarding the underlying distribution of the sample data.

For samples that originate from domain X, the histogram is obtained by first partitioning X into bins B such that

$$X = \bigcup_{l=1}^{\beta} B_l$$

and then counting the number of samples that belong each bin. β is the total number of bins used to construct the histogram. Once the histogram is available, the empirical PMF of the sample set is easily obtained by simply dividing the histogram values by the cardinality of the set.

Given a statistically significant sample size, it is clear that the only parameter that affects the quality of the PMF estimate obtained is the bin allocation B. If the partitions are then assumed to be equidistant for simplicity, the only variable that remains is the number of bins utilised: β. The effect of β on the histogram of a random process is shown in Figure 4.1. The random process in question is the instantaneous power density at any time-frequency point of the signal shown in Figure 4.1(a). It is a wireless local area network (WLAN) signal experiencing bursts of interference from a Bluetooth (BT) device. Histograms of the power



(a) Signal with interference

(b) Effect of bin count

Figure 4.1: *(a) Spectrogram of a WLAN signal experiencing interference from a Bluetooth device. The regular pattern is a single WLAN frame repeatedly transmitted by a signal generator. The frequency hopping nature of the Bluetooth transmission is clearly visible in the plot. (b) Impact of the number of bins utilised, β, on a histogram of the instantaneous power density of the WLAN signal. When β is too small, the histogram is insensitive to small changes and does not effectively capture the subtleties of the process. At the other extreme, it is too sensitive and therefore susceptible to noise. The optimal β is with respect to some minimum error criterion [95] and provides a good balance between resolution and sensitivity.*

density obtained using three different values of β are shown in Figure 4.1(b).

When a small number of bins are utilised, i.e. β is small, the histogram is insensitive to small scale variations in the input. As a result of the poor resolution, the estimated model fails to adequately capture the subtleties in the behaviour of the underlying random process. On the other hand, when the value of β utilised is too large, the resolution is too high and the histogram is overly sensitive – resulting in an estimate that is noisy. The optimal value of β yields a good balance between resolution and sensitivity.

A method of computing the optimal bin size (and hence the optimal β) for constructing a histogram, subject to some minimum mean squared error criterion when the underlying distribution is assumed to be due to a time-varying Poisson process, is provided by Shimazaki and Shinomoto in [95]. While the algorithm described therein is conceptually simple, it unfortunately requires the use of exhaustive search to iteratively minimise a certain cost function – making it too computationally expensive to be evaluated in real-time on a hand-held mobile device with limited energy and processing power.

The impact of β on the performance of each of the anomaly detection algorithms has been

investigated and the results are presented in Section 4.3.2.4.

Depending on the choice of B, there may be zeros in the estimated PMFs due to the presence of empty histogram bins. Under such circumstances, calculation of the KLD using (4.1) can be a problem as it leads to instances where $0\log_2\{0/q_n(x)\}$ or $p_n(x)\log_2\{p_n(x)/0\}$ have to be evaluated. While it is certainly possible to handle these as special cases by setting them to 0 and ∞ respectively through continuity arguments, it may be better to simply avoid zeros in the PMFs. It is possible to avoid empty histogram bins and hence zeros in PMFs by adding a small number, ε, to every bin of the histogram. As pre-loading of histogram bins in this manner undoubtedly distorts the estimate of the true PMF, the pre-load value must be carefully chosen. According to the work done by Krichevsky and Trofimov [96, 97], $\varepsilon = 0.5$ is a good choice.

4.2.2 Based on Kullback-Leibler divergence

The capability of KLD to quantise the difference, in a statistical sense, between two data sets to single real value is ideal for use in anomaly detection since it provides a convenient detection metric. A general description of the algorithm is provided here while a discussion of the optimisations needed for an efficient hardware implementation is provided in Chapter 5.

The flowchart in Figure 4.2 shows the proposed algorithm. At time n, the process starts with the acquisition of the two data sets to be compared using KLD. One of the data sets is a reference (Q_n) while the other is the one under test (P_n). If the samples in the data sets do not directly represent the parameter of interest, they must be processed. Once the data sets have been suitably transformed, the associated PMFs $p_n(x)$ and $q_n(x)$ are estimated and used to compute the KLD, D_n. If D_n is then observed to be larger than some predefined KLD threshold, D_{th}, the test data set may be anomalous.

This general approach to detecting anomalies using KLD can be easily adapted for use with signals containing periodic structures. One example of such a signal is IEEE 802.16e wireless broadband (WiBro) which utilises time division duplex (TDD) [98]. The procedure adopted for the analysis of periodic signals is shown in Figure 4.3. Periodic signals are expected to have statistics that are also periodic – implying that segments of the signal separated by the period, T_p, should have probability distributions that are very similar under normal circumstances.

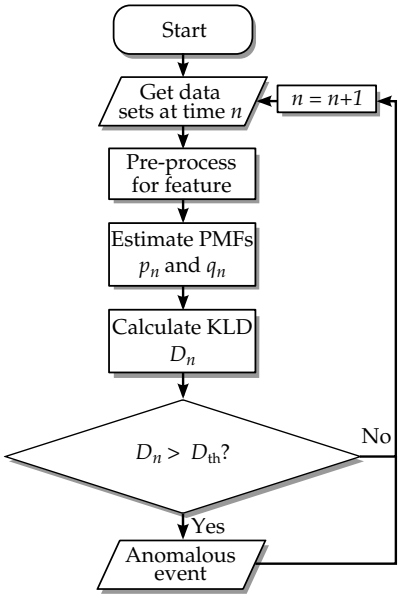


Figure 4.2: *The algorithmic flowchart. KLD is used to compare the statistical distribution of a test data set against that of a reference. If the divergence, D_n, is greater than some predefined threshold, D_{th}, the test set may be anomalous.*

Therefore, by simply acquiring the data sets P_n and Q_n from two sliding signal windows of length T_w with centres separated by T_p, the proposed algorithm can be utilised for the detection of anomalies in periodic signals. It should be noted that T_w must be less or equal to T_p to prevent overlap between the two signal windows. An appropriate detection threshold for the algorithm may be determined by simply observing the KLD values obtained when an anomaly-free segment of the signal under test is analysed using the algorithm. As KLD analysis can be performed on the signal envelope itself, demodulation is unnecessary and the only *a priori* information required by the algorithm is the signal period T_p.

The anomaly detection algorithm described can also be used for the detection of signal features such as frame structure and timing. No *a priori* information regarding the signal is needed and the modification required is simple: eliminate the condition that the centre of PMF windows must be separated by T_p and set the inter-window gap to zero. Figure 4.4 illustrates the method. It exploits the fact that the signal probability distribution changes drastically at frame boundaries. When the mutual centre of the adjoining pair of PMF estimation windows coincide exactly with a frame boundary, one of the windows is over the data bearing signal frame while the other is in the inter-frame space. As a result, the windows typically encompass two very different data sets – leading to a pair of dissimilar PMFs and

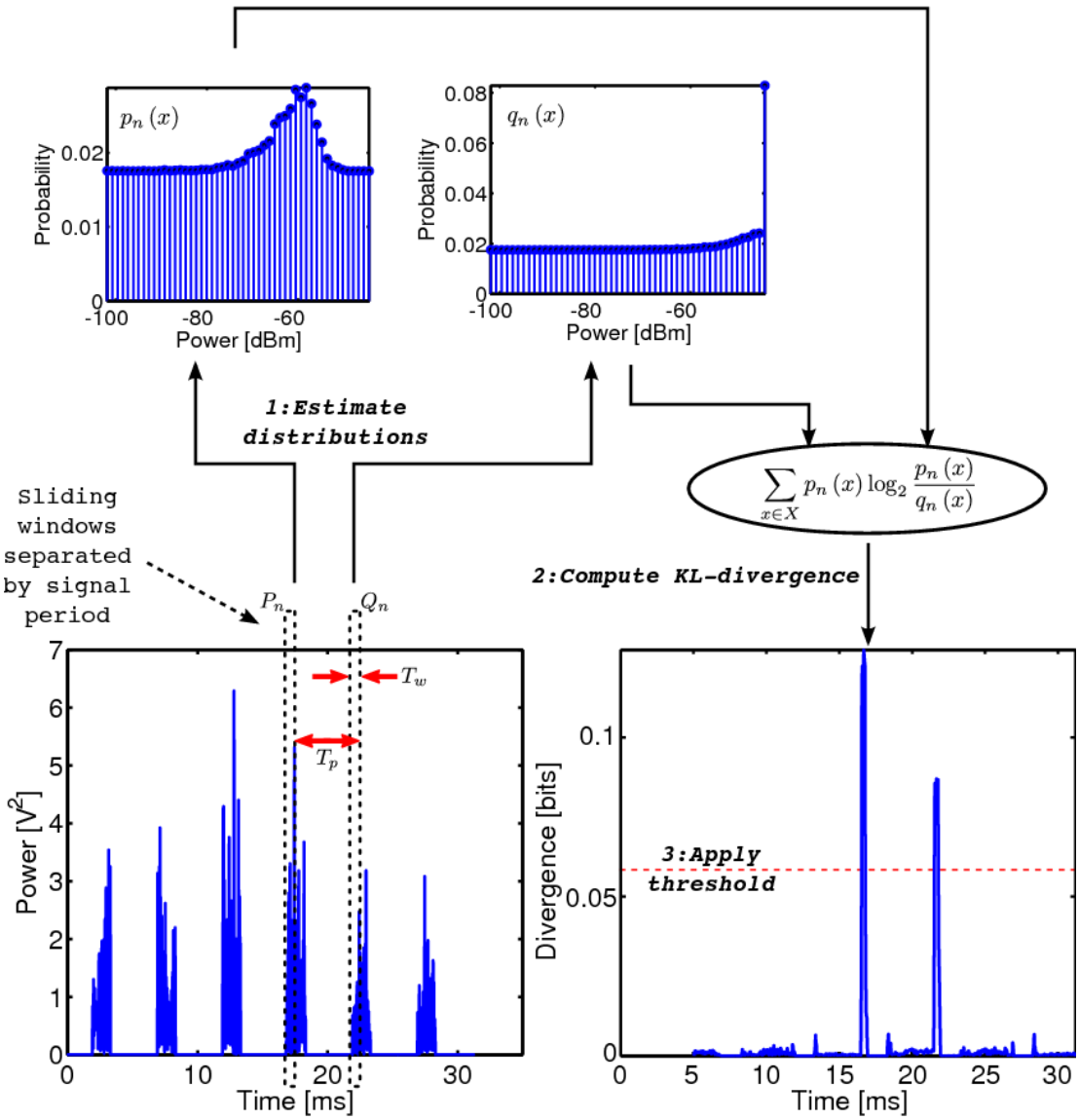


Figure 4.3: *Illustration of the KLD analysis algorithm for periodic signals. PMF estimates obtained from two sliding windows of length T_w separated by the signal period T_p are used to compute the KLD at each time step. If the KLD exceeds the predefined threshold, an anomaly may be present.*

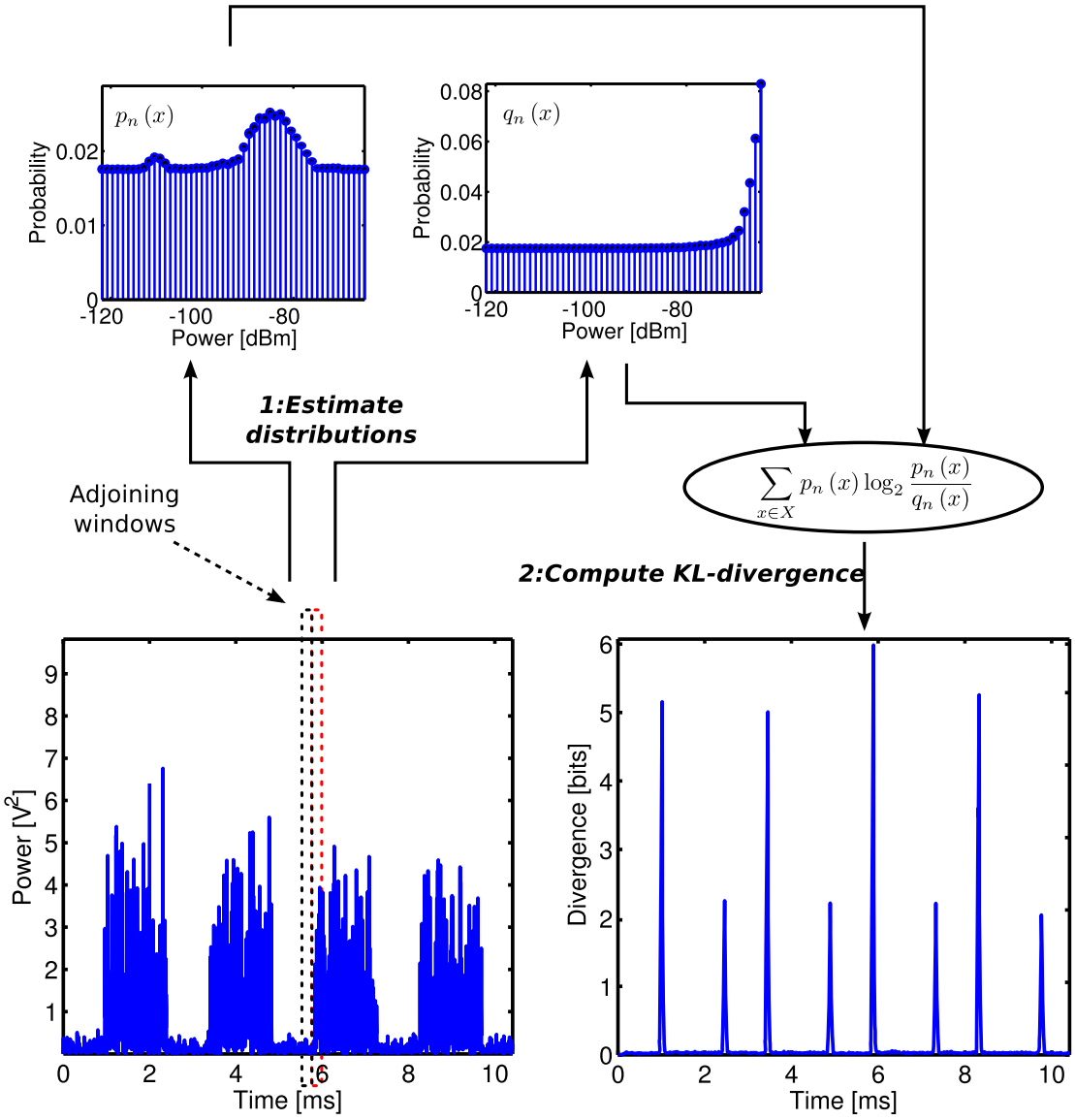


Figure 4.4: *Illustration of the KLD based algorithm for signal timing structure detection. Two adjoining windows are utilised for PMF estimation and KLD computation. Signal probability distributions change suddenly at frame boundaries – leading to peaks in the divergence when they coincide with the centre of the window pair. The peaks then define the detected signal timing structure.*

	Σ	×	÷	log	Total
Operations	$2\|P_n\|+\beta$	β	3β	β	$2\|P_n\|+6\beta$
Memory		$\|P_n\|+\psi+2\beta$			

Table 4.1: *Runtime operational complexity and memory requirements of the Kullback-Leibler divergence based anomaly detection algorithm.*

hence a large KLD. Peaks in the KLD analysis then correspond to frame boundaries in the signal – revealing the underlying timing structure. This information is useful as it can be used to estimate the signal period required by the aforementioned anomaly detection algorithm in an autonomous fashion.

While the steps required to compute the KLD are all simple and straightforward, the storage (data buffers) and the number of arithmetic operations required grow linearly with the size of input data sets. As these data sets can be very large when analysing high speed signals, it can easily lead to scenarios where it may not be possible to provide for the resources required by the algorithm. Analysis of the algorithm's complexity and memory requirements follows and is summarised in Table 4.1.

The input data sets P_n and Q_n themselves require a buffer capable of holding at least $|P_n|+\psi$ elements, where ψ is the number of samples corresponding to the signal period and $|P_n|$ $(=|Q_n|)$ is the size of the data windows in samples. Only a single buffer is required for the input data since one of the data sets is essentially just a ψ-delayed version of the other in this case. Computing the frequency count over the bins (B), for the purpose of estimating the histograms, requires up to $|P_n|$ additions for each of the two windows. Once the histograms are available, the PMFs are obtained by dividing the frequency count in each of the β bins by $|P_n|$. Two buffers of size β each are then required to store the resulting PMFs. Computation of the KLD from the PMFs then require a further β divisions, logarithms, multiplications and additions respectively.

4.2.3 Based on information content analysis

Unlike the KLD based anomaly detection algorithm just described, the ICA based method analyses individual input samples rather than aggregate sets of data. The information con-

veyed by the events is the detection metric utilised. The following is a general description of the algorithm.

First and foremost, it is necessary to establish the type of event that is under observation. This can be any property that is associated with the signal under test (e.g. instantaneous amplitude, phase, or power). If the event type chosen is measurable directly from the signal envelope, demodulation is unnecessary for anomaly detection. The ICA algorithm utilises supervised learning to establish a reference histogram (and hence probability) of events; therefore, some anomaly-free signal is required for training. Once the reference histogram is obtained, online analysis of the signal under test can commence. Events from the test signal are extracted and used to update the reference histogram. This yields updated event probabilities and hence the associated information content. If the information content $I_n(x)$ of any event x_n, at time n, is above some predefined threshold I_{th}, an anomaly may be present.

Once again, it is clear that the event histogram plays a central role in the anomaly detection algorithm. It has been stated previously in Section 4.2.1 and illustrated by Figure 4.1 that the number of bins utilised, β, has a significant impact on the sensitivity of the histogram and hence the effectiveness of the detection algorithms. When β is too small, anomalous events may not be detected due to poor sensitivity – leading to missed detections. On the other hand, when β is too large, even nominal events will appear to have low probability – leading to a large number of false positives. It is therefore necessary to find a β that offers a good balance between sensitivity and probability of detecting false positives.

The event histograms shown in Figure 4.1(b) reveal another potential challenge for the ICA algorithm. It can be seen that the histograms have long tails with numerous low probability (i.e. high information content) events even when the signal is behaving nominally. Although this is expected for any analogue signal transmitted over a lossy physical channel, it raises the possibility that numerous false positives are observed at a detector that employs a simple information content threshold. It is certainly possible to reduce the number of tail events by using a smaller number of bins, but that leads to reduction in sensitivity and hence an increase in the probability of missed detections.

Examination of the interference scenario in Figure 4.1(a) reveals an important distinction between anomalous events and the underlying signal – anomalies tend to appear in clusters while nominal low-probability signal events are decidedly "singular". This difference is the

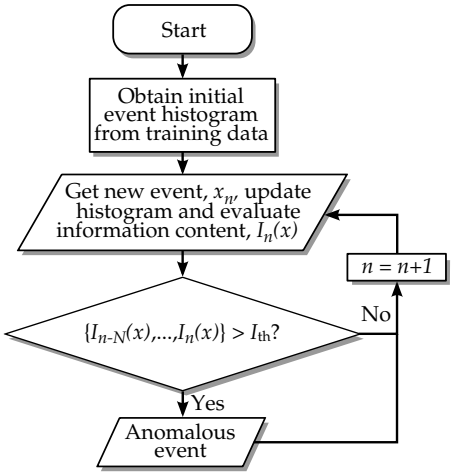


Figure 4.5: *Flowchart of information content analysis algorithm with clustered anomaly detection. First, clean data is used to initialise the reference event histogram and event probabilities. Then, events in the signal under test are used to update the reference histogram and event probabilities. The updated values are used to estimate the information content of the events. If the information content of $N+1$ contiguous events exceed a predefined threshold, I_{th}, an anomaly may be present in the signal under test.*

key to reducing the number of false positives while still maintaining a low rate of missed detections. The proposed algorithm is easily augmented to benefit from this insight: instead of triggering on individual high information content events, the detector must search for contiguous groups of events that exceed the predefined information content threshold.

A flowchart of the algorithm with simplified clustering is shown in Figure 4.5. The general approach is as before, with the exception of the last step. With the simple clustering extension, detection of an anomaly is signalled only when a contiguous sequence of N previous events and the current event exceeds a predefined information content threshold . Sequence detection is used rather than full two-dimensional clustering to minimise the complexity of the algorithm. This is permissible since a sequence can be considered as a one-dimensional cluster. While a larger value of N will certainly yield a smaller number of false positives due to spurious nominal events, it will also make it impossible to detect temporally brief anomalous events. Due to this trade-off it is necessary to select an appropriate sequence length. The effect of N on the detector performance is examined in Section 4.3.2.5.

The discrimination threshold is an important aspect of any detector. While the optimum threshold is problem and cost function specific, it is generally chosen to minimise missed detections while still maintaining a low rate of false positives. For the proposed anomaly

Operations	Σ	$\times$	$\div$	log	Total
	2	0	1	1	4
Memory	$\beta + N + 2$				

Table 4.2: *Runtime operational complexity and memory requirements of the information content analysis based anomaly detection algorithm.*

detection algorithm, it is not possible to define a single information content threshold, I_{th}, that is suitable for use with any arbitrary signal. I_{th} is signal specific and may be set automatically using information obtained from the clean training data. After the reference event histogram and probabilities have been estimated, the reference information content associated each event type can be easily computed using (4.4). For β bins, the standard deviation, $\sigma_{\text{I}\langle\beta\rangle}$, of the reference information content provides a measure of the spread and may be used to obtain the threshold:

$$I_{\text{th}} = m\sigma_{\text{I}\langle\beta\rangle}. \tag{4.5}$$

m is a multiplicative factor greater than 1. The effect of I_{th} on detector performance is investigated in Section 4.3.2.3.

Due to the simplicity of the ICA algorithm, its runtime operational complexity and memory requirements are negligibly small. On completion of the initial training phase, a small buffer capable of holding just $\beta + 1$ elements is required to store the event histogram and the total events count. At runtime, analysis of an event requires 2 additions to increment the relevant bin count and the total events count. Division of the incremented bin count by the total is then needed to obtain the event probability. After the probability is computed, a single base-2 logarithm is needed to calculate the event's information content. An additional buffer capable of holding $N+1$ elements is also needed to accommodate information content clustering. A summery of the complexity analysis is provided in Table 4.2. It reveals that in addition to being negligibly small, the fixed runtime operational complexity is independent of any algorithmic parameter (e.g. histogram resolution) – suggesting that fast and efficient implementations for power limited hand-held devices are possible.

4.3 Results

In order to evaluate the performance of the proposed anomaly detection schemes, signals with different classes of abnormalities are employed as test cases. All of the signals under test are actual radio frequency transmissions captured using spectrum analysis hardware and therefore represent scenarios likely to be encountered by real world wireless devices.

Analysis of the test signals are provided in the following section while detailed performance analysis of the algorithms based on parameters such as histogram bin resolution, data window size, sampling rate and cluster length are provided in Section 4.3.2.

4.3.1 Data analysis

Of the four data sets available, the first three are used to demonstrate the PMF divergence analysis (KLD/L1D) based technique while the last is used to demonstrate the ICA based technique.

4.3.1.1 Test signal A

The signal is shown as a time series in Figure 4.6(a). It consists of a single WLAN frame that repeats with a period of 2.45 ms and a burst of interference from a Bluetooth device that is visible at 11.5 ms. The signal is similar to that shown earlier in Figure 4.1.

Both KLD and L1D are used to analyse the signal for the purpose of obtaining results that can be directly compared. Two windows with a duration of 256 μs each are employed to process the time series signal. The window centres are separated by 2.45 ms to match the WLAN frame repetition interval. The windows estimate the PMFs of the signal power. The number of histogram bins utilised is the optimal value (51 in this case) as obtained from the algorithm proposed by Shimazaki and Shinomoto [95]. In any case, it is shown in a subsequent section (4.3.2.4) that the number of histogram bins used does not have a significant impact on the outcome – therefore, an arbitrary but reasonable choice such as 32 can also be used instead.

The result of the analysis is shown in Figure 4.6(b). Both KLD and L1D based methods are successful at detecting the presence of the anomaly (BT interferer). However, it is clear that KLD is the better choice as it produces a larger peak compared to L1D when the anomaly is detected. The baseline noise level with KLD is also much lower than that obtained with L1D

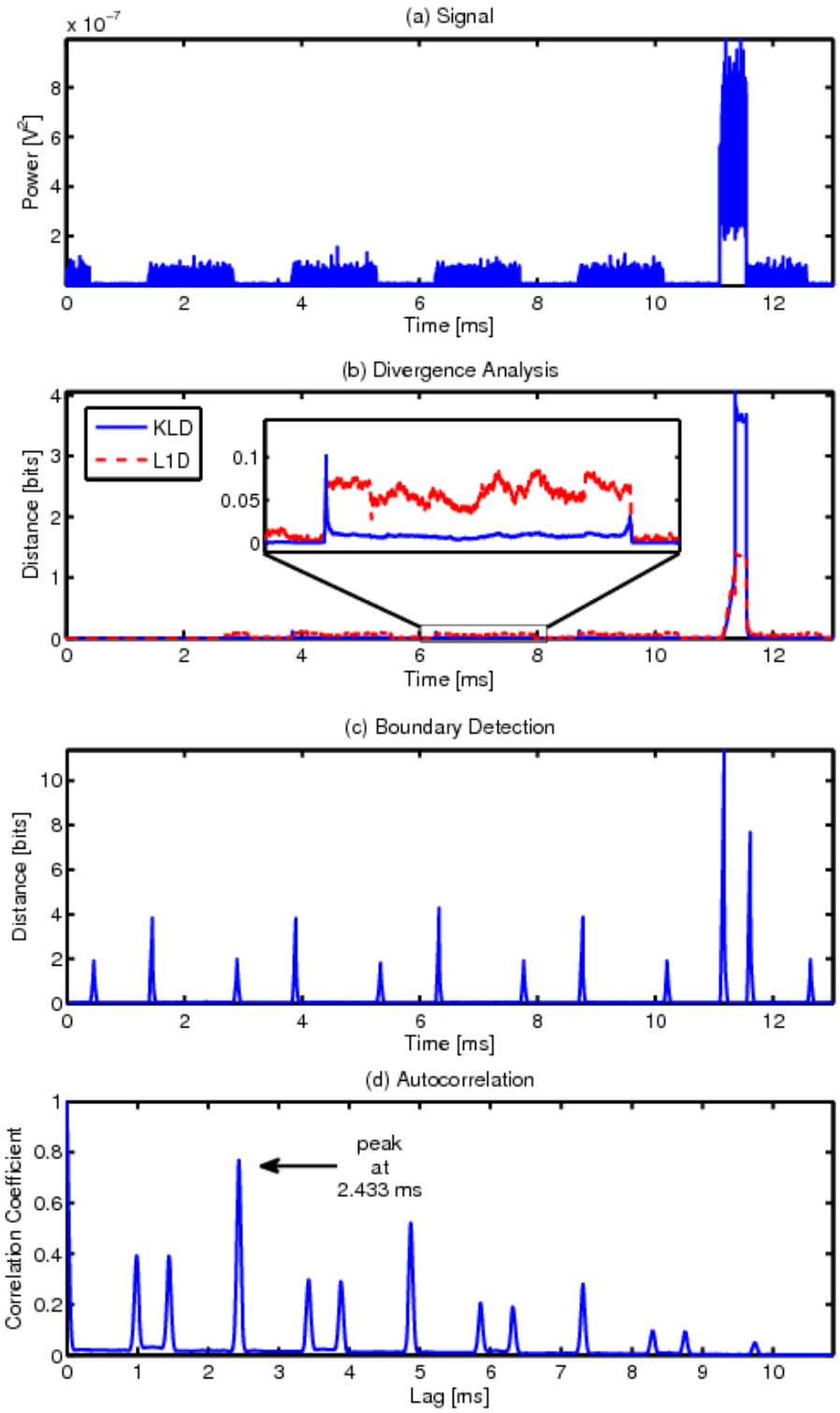


Figure 4.6: *(a) WLAN signal with interference at* 11.5 *ms from a Bluetooth device. (b) Both KLD and L1D analysis detect the anomaly. KLD is better since it produces a larger peak and lower noise. (c) Signal boundary detection using KLD. (d) Normalised autocorrelation of the first* 10.85 *ms of the KLD trace in (c). Estimate of the period is* 2.433 *ms (*0.69% *error).*

– confirming the hypothesis presented in Section 4.2.

The signal is also processed using the boundary detection algorithm described previously in Section 4.2.2. The PMF estimation window size utilised is 64 μs to provide a good resolution and peak localisation. The number of histogram bins utilised is 32. Figure 4.6(c) shows the result of the analysis. Sharp KLD peaks at every signal transition indicates that the technique is indeed successful. Since the underlying signal is periodic, the peaks are also periodic within the clean portion of the time series (0 – 10.85 ms). Given such a trace, it is then possible to apply signal processing techniques such as autocorrelation to obtain an estimate of the signal period T_p. It is necessary to only use clean reference data for period estimation since anomalies lead to changes in the structural and statistical properties of the signal.

The result of performing normalised autocorrelation on the anomaly-free segment of Figure 4.6(c) is shown in Figure 4.6(d). Only positive lags are considered since the function is symmetric. As expected, the largest autocorrelation coefficient is obtained at a lag of zero. However it is the second largest coefficient that is of interest. It occurs at 2.433 ms – providing an excellent estimate (error of 0.69%) of T_p which is known to be 2.45 ms.

4.3.1.2 Test signal B

The second test vector is a WiBro signal. The time domain signal is shown in Figure 4.7(a). Due to the proximity of the recording equipment to the mobile terminal (MT), the uplink (UL) sub-frames show a higher power level than the downlink (DL) sub-frames. From the plot, it can be seen that the UL sub-frame at 16 ms is longer than any of the other UL sub-frames. In context of this particular signal snapshot, this behaviour is unusual and hence can be considered to be anomalous. Once again, two windows with a duration of 256 μs each are employed to estimate the signal power PMFs. The window centres are separated by 5 ms – corresponding to the frame period of the signal. The optimal histogram bin allocation scheme in [95] is once again used to determine the number of bins utilised ($\beta = 51$).

Processing the test signal using the divergence based anomaly detector produces the result shown in Figure 4.7(b). A sharp peak in the divergence at 16 ms reveals the presence of the unusual UL sub-frame. A second peak is obtained when the signal returns to normal in the following UL sub-frame – this is due to the twin-windowing nature of the anomaly detection algorithm. Once again, the superiority of KLD over L1D as a divergence metric is

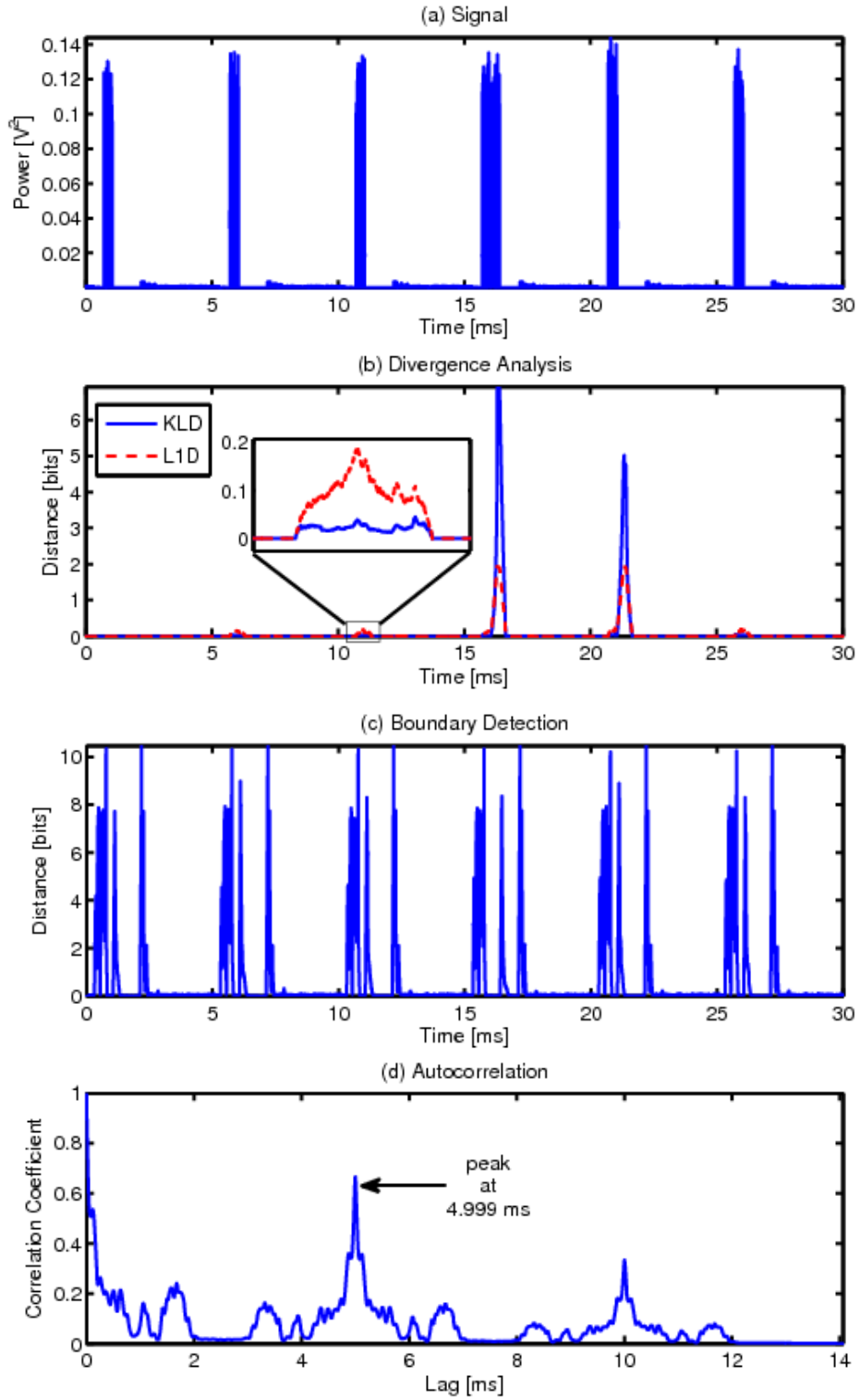


Figure 4.7: *(a) WiBro signal with an anomalous UL sub-frame at* 16 *ms. (b) Both KLD & L1D analysis reveal the anomalous UL segment. A second peak is obtained when the signal returns to normal. Again, KLD analysis results in a larger peak and lower noise. (c) Signal boundary detection using KLD. (d) Normalised autocorrelation of the first* 14 *ms of the KLD trace in (c). Estimate of the period is* 4.999 *ms (*0.02% *error).*

demonstrated by the larger peaks and lower baseline noise levels.

Boundary detection is then applied to the test signal using the same parameter combination as that used in Section 4.3.1.1. Figure 4.7(c) shows the KLD plot obtained. In addition to the UL and DL frame boundaries, some sub-frame level features such as the preamble and payload borders are also identified. Autocorrelation is then performed on the first 14 ms of the trace to obtain an estimate of the period. The result is shown in Figure 4.7(d). The second largest coefficient is obtained at a lag of 4.999 ms, providing an estimate of the period that is once again highly accurate (error of 0.02%).

4.3.1.3 Test signal C

The third and final signal used to test the detection capabilities of the divergence based algorithm is shown in Figure 4.8(a). It depicts communication between a MT and base station (BS) using the WiBro standard. Since the recording is made at the MT, there is significantly more power in the UL sub-frames. Although unnoticeable in the time series, the UL sub-frame at 20 ms contains an additional command sequence that triggers the subsequent change in the transmit power observed at 25 ms. As a result, there are effectively two unusual events in the signal: the extra command and the subsequent change in power level. The parameters utilised for analysis of the signal are identical to those used in Section 4.3.1.2.

The divergence analysis plot in Figure 4.8(b) shows that both anomalies can be successfully detected using KLD and L1D. Since the width of a KLD peak corresponds to the temporal duration of the anomaly responsible, the first peak at 20 ms is very sharp as it is due to the extra command sequence in the UL sub-frame. Since the subsequent change in power at 25 ms affects the entire UL sub-frame, the second peak is much broader and spans the entire sub-frame.

The plot also reveals that for the second anomalous event (power change), KLD is smaller than the associated L1D. This is not unexpected since it has been hypothesised in Section 4.2 that for differences that lead to a L1D of $2\ln 2$ or smaller, L1D can be larger than KLD. It is also the reason why L1D generally leads to larger baseline noise levels compared to KLD.

Application of the boundary detection algorithm to the test signal produces the result shown in Figure 4.8(c). Once again, the parameters used are identical to those utilised for the signal in Section 4.3.1.1. Although the test signal contains both UL and DL frames, only the UL

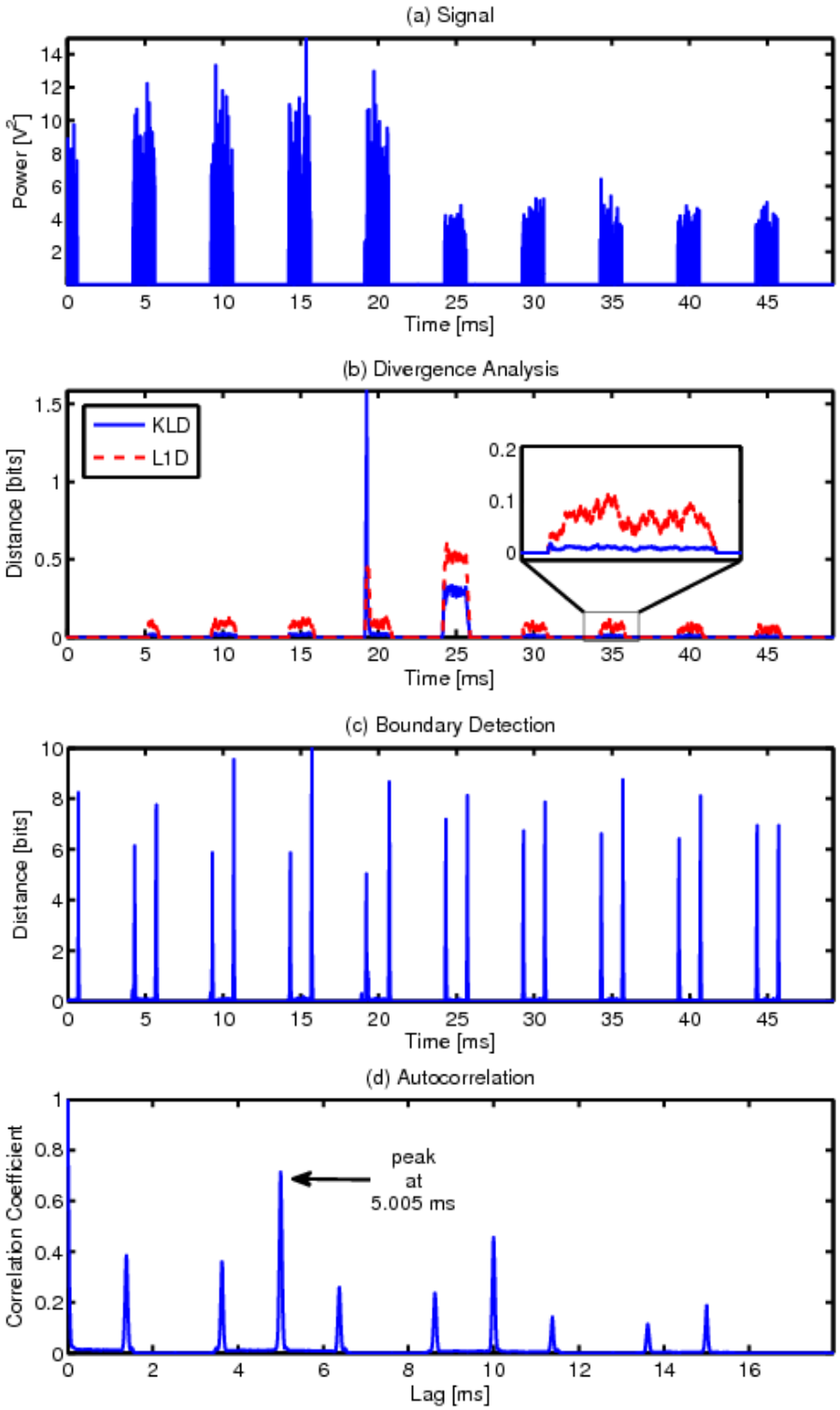
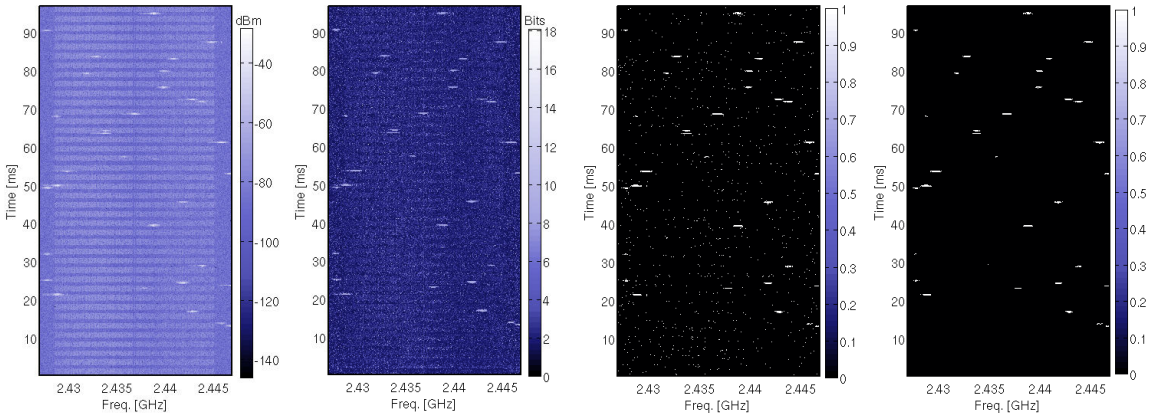


Figure 4.8: *(a) WiBro signal. An extra command in the UL sub-frame at* 20 *ms initiates the power control loop. (b) The extra command sequence (*20 *ms) and subsequent power change (*25 *ms) are both revealed by KLD/L1D analysis. Larger KLD is obtained from the anomalous command but not from the change in power. L1D leads to larger noise levels compared to KLD. (c) Signal boundary detection using KLD. (d) Normalised autocorrelation of the first* 17.95 *ms of the KLD trace in (c). Estimate of the period is* 5.005 *ms (*0.1% *error).*



(a) Test signal (b) Information content (c) Anomalies ($N = 0$) (d) Anomalies ($N = 2$)

Figure 4.9: *Analysis of a WLAN signal with interference from a Bluetooth device. (a) Spectrogram of the original signal. Data bursts from the Bluetooth device are clearly visible as high power, lightly shaded patches. (b) Information content of events (power density at any given time-frequency point) in the signal. (c) Anomalies detected (light patches) using a threshold of* $1.25\sigma_{\mathrm{I}\langle 16\rangle}$ *(7.15 bits) and a cluster length of 1. The result is noisy and there are a lot of false positives (appearing as singular, lightly shaded spots). (d) Anomalies detected when the threshold is left unchanged at* $1.25\sigma_{\mathrm{I}\langle 16\rangle}$ *and the cluster length is increased to 3. The outcome is now much cleaner with virtually zero false positives.*

frame boundaries are detected. The reason for this is the 50 dB difference in power level of the UL and DL frames. Consequently DL frames are indistinguishable from noise when only 32 histogram bins are used for PMF estimation.

In order to estimate the signal period, autocorrelation is applied to the first 17.95 ms of the KLD trace in Figure 4.8(c). The second largest coefficient is observed at 5.005 ms – providing an estimate that contains an error of only 0.1%.

4.3.1.4 Test signal D

The final test signal is used to evaluate the ICA algorithm. It is similar in nature to the signal shown in Figure 4.1(a). Spectrogram of the test signal is shown in Figure 4.9(a). It is a much longer signal with numerous interference events to provide a statistically significant sample size. The plot depicts a real WLAN signal with Bluetooth interference captured over the air-interface. The WLAN signal consists of a single frame that is repeated periodically by a vector signal generator. The characteristic frequency hopping pattern of the Bluetooth device marks the locations of the interference (anomalous) events.

The signal spectrogram is estimated from the time series using non-overlapping Hamming windows that are 64 μs long. A 1024 point fast Fourier transform (FFT) is used to obtain a frequency resolution of approximately 20 kHz. The signal event under observation is the instantaneous power density at any given time-frequency coordinate. The first 10 ms of the signal is assumed to be free from interference and is therefore used for training purposes. 16 equally spaced histogram bins divide the range between the maximum and minimum power densities observed in the training data. The $\sigma_{\mathrm{I}\langle 16\rangle}$ for the training data is 5.72 bits.

Figure 4.9(b) shows the information content of events in the test signal. As expected, the anomalous events have a higher information content and there are highlighted while the regular underlying structure is suppressed. The plot also shows that there is a lot of noise (tiny spots of high information content) from individual low-probability signal events that are otherwise nominal. The reason for this behaviour has been outlined in Section 4.2.3.

Anomalies detected using a threshold of $1.25\sigma_{\mathrm{I}\langle 16\rangle}$ and a cluster length of 1 (i.e. only the current event) is shown in Figure 4.9(c). It is immediately obvious from the large number of small, lightly shaded spots that there are a lot of false positives (67%). Again, singular low-probability signal events are responsible since they can potentially have higher information content than actual anomalous events. Keeping I_{th} the same and increasing the cluster length to 3 yields the result shown in Figure 4.9(d). It reveals that a simple change in the cluster length is sufficient for greatly reducing the total number of false positives (2.2%).

4.3.2 Performance analysis

The analysis of test signals presented in Section 4.3.1 show that both algorithms perform well for the parameter combinations chosen. In order to investigate and quantify the impact of other parameter choices, it is necessary to define and utilise metrics that reflect performance.

For the divergence based technique, the ratio between the anomaly detection peak and the maximum of the baseline noise level is a good indicator of performance since it is a reflection of the range over which a threshold can be applied. It can be seen from the results presented in Figure 4.6, Figure 4.7 and Figure 4.8 that KLD is an extremely effective discriminator for statistical changes in the observed data. Even with such *real test vectors captured over-the-air*, the KLD peaks produced by anomalous events are many orders of magnitude larger than baseline noise levels associated with nominal data. As a result, 100% probability of

detection can be achieved over a wide range of KLD threshold values (the anomalous peak is approximately 140 times as large as the background noise level in Figure 4.7) while still guaranteeing a 0% probability of false positives – making such classical measures of detector performance inadequate for gauging the true extent of the algorithm's performance.

Another reasons against the suitability of classical performance measures such as receiver operating characteristic (ROC) curves [74] – which show the relationship between the probability of detection and the probability of false positives – is the scarcity of available test data. Probability of detection and false positives are inherently statistical measures of performance that require a large sample size to produce meaningful results. Since the focus of this work is exclusively on practical applications of the proposed algorithm, the number of test vectors available are limited and each test signal (A, B and C) contains only 1 or 2 anomalous events. So instead of attempting to extract questionable probability measures from the limited data set, measure of ratio between the anomalous peak and background noise is utilised to quantify the observed performance.

When KLD is used as the measure of divergence, the KLD ratio (KLDR) is defined as

$$\mathrm{KLDR} = \frac{\mathrm{KLD}_{\mathrm{anom}}}{\mathrm{KLD}_{\mathrm{bg}}} \tag{4.6}$$

where $\mathrm{KLD}_{\mathrm{anom}}$ is the maximum of the detection peak and $\mathrm{KLD}_{\mathrm{bg}}$ is the maximum of the background baseline noise level. KLDR is the metric that is used to quantify the algorithm's performance.

For the ICA based algorithm, the circumstances are different. The test set (signal D) contains a sufficient amount of nominal and anomalous events to allow the use of more traditional performance metrics. Performance is measured in terms of the detector true positive rate (R_{tp}) and false discovery rate (R_{fd}). R_{tp} is defined as the ratio of the number of correctly detected anomalous events (Σ_{tp}) to the total number of anomalous events present (Σ_{ta}):

$$R_{\mathrm{tp}} = \frac{\Sigma_{\mathrm{tp}}}{\Sigma_{\mathrm{ta}}} = 1 - \frac{\Sigma_{\mathrm{md}}}{\Sigma_{\mathrm{ta}}} \tag{4.7}$$

where Σ_{md} is the number of anomalous events that missed detection. R_{fd} is the ratio of false

positives (Σ_{fp}) to the total number of anomalies detected (includes both Σ_{fp} and Σ_{tp}) [99]:

$$R_{fd} = \frac{\Sigma_{fp}}{\Sigma_{fp} + \Sigma_{tp}}. \tag{4.8}$$

R_{fd} is preferred over the more common false positive rate (R_{fp}) as it is more useful in this context. R_{fp} is defined as the ratio between Σ_{fp} and all non-anomalous events (Σ_{tn}) in the signal:

$$R_{fp} = \frac{\Sigma_{fp}}{\Sigma_{tn}}. \tag{4.9}$$

It is also known as the false alarm rate. Since Σ_{tn} is a very large number, R_{fp} is close to zero for most parameter combinations and therefore does not adequately reflect the variations observed in detector performance.

To summarise, KLDR is used to evaluate the performance of the KLD based algorithm while R_{tp} and R_{fd} are used to evaluate the ICA based algorithm.

4.3.2.1 Sampling rate

Continuous processes such as time-series must be sampled before the anomaly detection algorithms can be applied. The sampling frequency employed is crucial as it dictates the size of the input data sets, $|P_n|$, and therefore the memory utilisation of the KLD based algorithm – as indicated in Table 4.1. For a given window length, a higher frequency implies that more data samples have to be stored and sorted to construct the histograms. If the frequency is too low, small scale signal features and anomalies may be lost. According to the Nyquist sampling criterion, a signal must be sampled with a frequency at least twice as large as its bandwidth to be reconstructible. For wideband signals this leads to a very high sampling frequency and hence a prohibitively large volume of data – heavily increasing the resource requirements of the proposed scheme. Since neither of the proposed algorithms require the time-series to be reconstructible, a far lower sampling frequency can be used instead. Figure 4.10 shows how the performance of the KLD based anomaly detection scheme is affected by under-sampling of the input time series. The window length utilised is 256 μs and the histograms used to construct the PMFs are 32 bins wide. The amount by which the input time-series is under-sampled relative to the bandwidth is defined as the decimation factor (DF). Therefore, a factor of unity implies that the signal is sampled at the same frequency as the signal bandwidth.

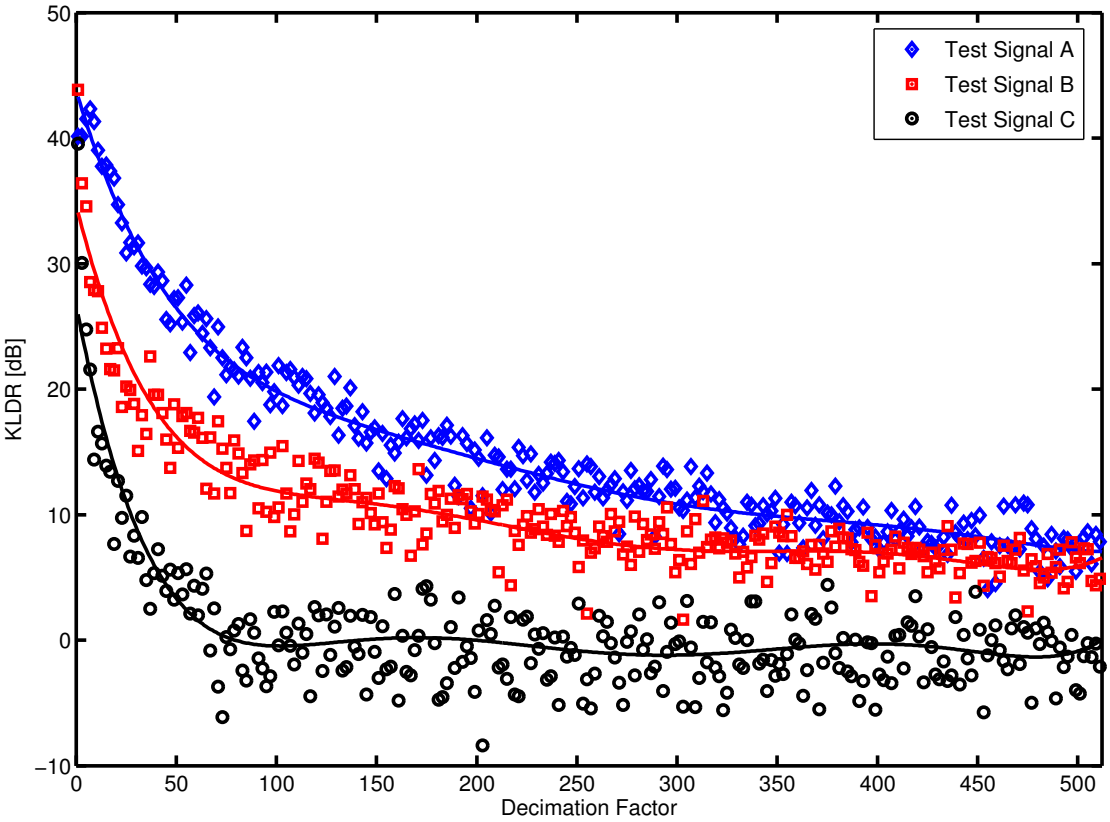


Figure 4.10: *Performance of the KLD based anomaly detection scheme under various data sampling rates. Solid lines are spline fits of the data and used to highlight trends. Decimation factor refers to the amount by which the input signal is under-sampled relative to the signal bandwidth. A factor of unity corresponds to a sampling frequency equal to the signal bandwidth. For two of the test cases (A and B), a KLDR well above* 10 *dB can be maintained for a window length of* 256 *µs, histogram bin count of* 32*, and a decimation factor of* 100*. Signal C is unable to accommodate such high decimation rates.*

The results indicate that decimation factors as large as 500 can be successfully employed depending on the type and duration of the anomaly present. For test signals A and B, a KLDR of more than 10 dB can be maintained even with a decimation factor of 100. This is an important result as it indicates that satisfactory performance levels can be maintained with little input data and hence memory-limited implementations of the algorithm. At high decimation factors, performance is poor for test signal C. This is because the first anomaly (extra command sequence) is temporally brief and is likely to be lost when the signal is heavily under-sampled. As for the second anomaly in the signal, the change in power is simply not large enough to produce a significant increase in the divergence.

Results for the ICA based method are not shown since the input sampling rate has no bearing

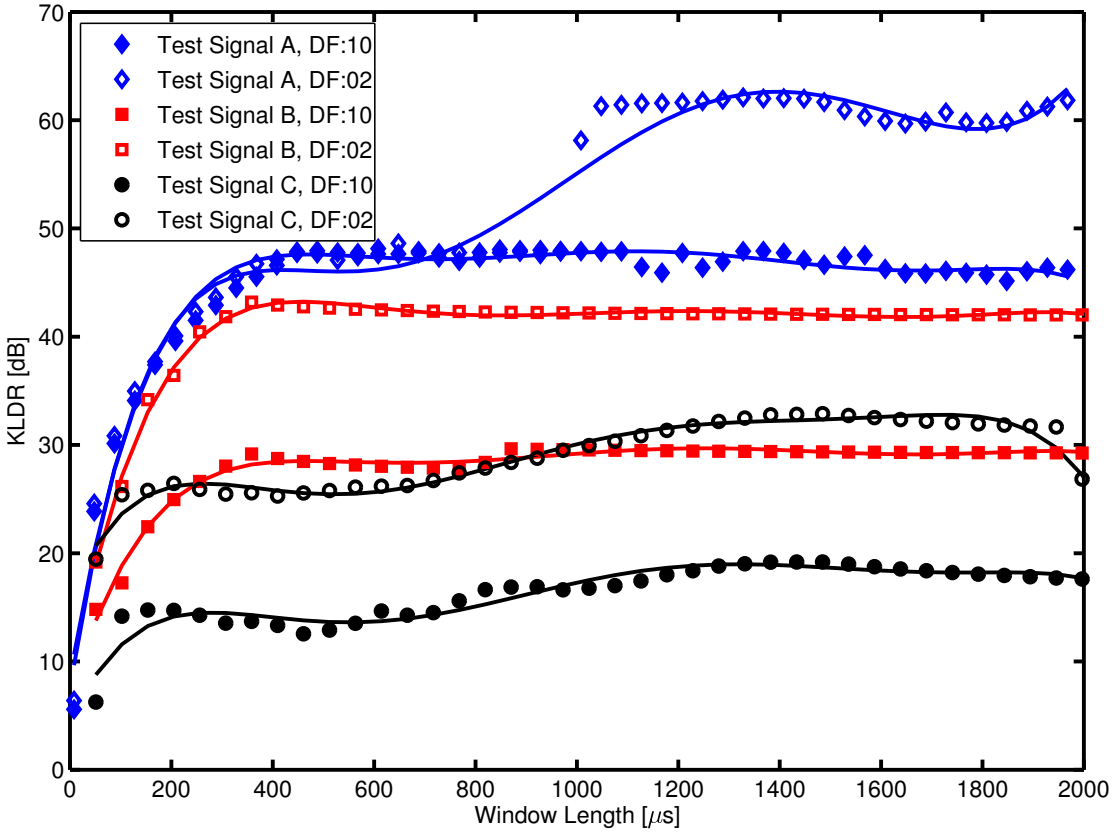


Figure 4.11: *Performance of the KLD based anomaly detection scheme under various window lengths and hence the input sample size $|P_n|$. Solid lines are spline fits of the data and used to highlight trends. DF of 2 and 10 are utilised and the number of histogram bins is 32. In all cases, an increase in the amount of data leads to a better performance. However, beyond a certain window length, the performance is no longer strongly affected.*

on the complexity or memory requirements of the algorithm – as shown in Table 4.2. The only requirement then on the sampling rate is that it must be fast enough to capture events that are suspected of being anomalous.

4.3.2.2 Window length

The window size, and hence the input data set size $|P_n|$, is another parameter that is relevant for the KLD based algorithm but not the ICA based algorithm. The effect of the PMF estimation window size on the performance of the algorithm shown in Figure 4.11. The number of histogram bins utilised is 32. Results are shown for under-sampling factors of 2 and 10. At lower decimation factors, more data is available and the KLDR improves uniformly across all window sizes for signals A and B. At smaller window sizes, performance for signal A is

unaffected by the choice of the decimation factor due to the relatively long duration of the anomaly. This is because even at a decimation factor of 10, a sufficient number of anomalous samples are represented in the PMF.

As anticipated, the performance is poor at small window sizes where the amount of data available is insufficient to adequately model the underlying PMFs. Increasing the window length leads to an improvement of the performance. However, for signals B and C, the gains become marginal for windows larger than approximately 400 μs. The transition shown by signal A at a window length of 1 ms for a decimation factor of 2 is due to a sudden reduction in the KLD noise at the frame edges (as seen in the zoomed-in segment of Figure 4.6) while the detection peak remains at approximately the same level. It is no coincidence that the duration of the inter-frame spacing for the signal is also 1 ms. It is a signal feature that is detected by the algorithm alongside the actual anomalous events. When the window size is increased beyond this feature size, it can no longer be resolved effectively by the detection algorithm and this leads to a decrease in the noise level seen at the frame edges.

The initial KLDR improvements with increasing window size is due to improvements in the PMF estimates which in turn lead to a reduction in the baseline KLD levels. At larger window sizes, the anomalous samples represent smaller fractions of the data and hence contribute less to the shape of the estimated PMF – resulting in a decrease of the KLD due to the anomaly. As the background levels are also reduced by an increase in the data size, the overall KLD ratio (i.e. the KLDR) remains relatively constant.

4.3.2.3 Information threshold

As stated previously in Section 4.2, the discrimination threshold (4.5) is an important aspect of any detector. The impact of I_{th} on R_{tp} and R_{fd} of the ICA based algorithm is investigated using a cluster length of 3 and histogram bin sizes of 8, 16, and 32. The result of the analysis is shown in Figure 4.12.

The plot shows that there is a hard I_{th} boundary for each β after which R_{tp} drops rapidly. This implies that the majority of the anomalous events share similar characteristics and convey information equivalent to that boundary. When I_{th} is increased further through the use of a larger threshold factor m, R_{tp} approaches zero due to an ever increasing number of missed detections.

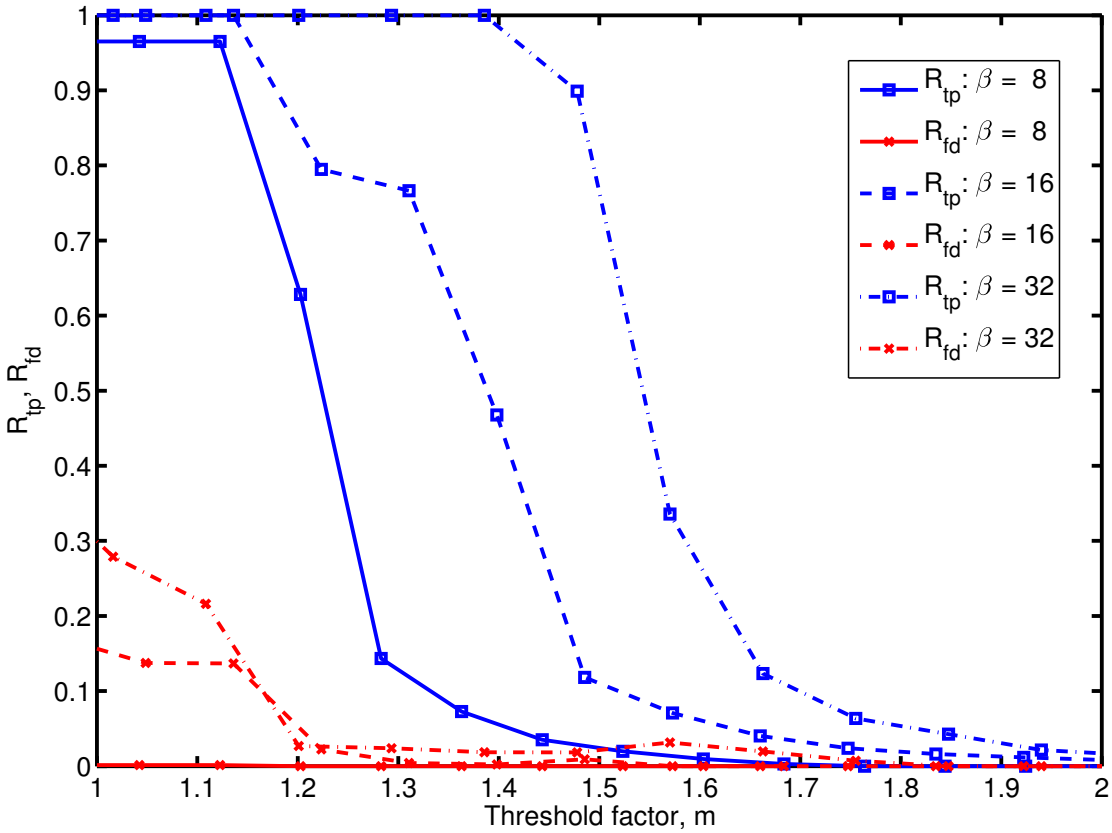


Figure 4.12: *Effect of information threshold* (4.5) *on detector* R_{tp} *and* R_{fd} *for the ICA algorithm. The cluster length utilised is 3. Histogram bin sizes of 8, 16, and 32 are used for comparison.* R_{tp} *for each* β *shows a sharp decrease when* I_{th} *(i.e.* $m \cdot \sigma_{\text{I}(\beta)}$*) is increased above a certain limit. Majority of the anomalous events have an information content less than this and miss detection.* $m = 1.35$ *and* $\beta = 32$ *yields the best performance.*

At low information content thresholds, R_{fd} is also high – specially for high values of β. As explained earlier in Section 4.2.1, a higher resolution makes the detector more susceptible to noise, leading to an increase in the number of false positives and hence the R_{fd}.

The impact of D_{th} on the performance of the KLD based method has not been investigated and therefore cannot be shown. The reasons for this are:

- The total number of anomalous events available is insufficient to investigate statistical trends.

- Detection is often guaranteed for a wide range of thresholds due to the large KLDR (greater than 30 dB) values that are observed.

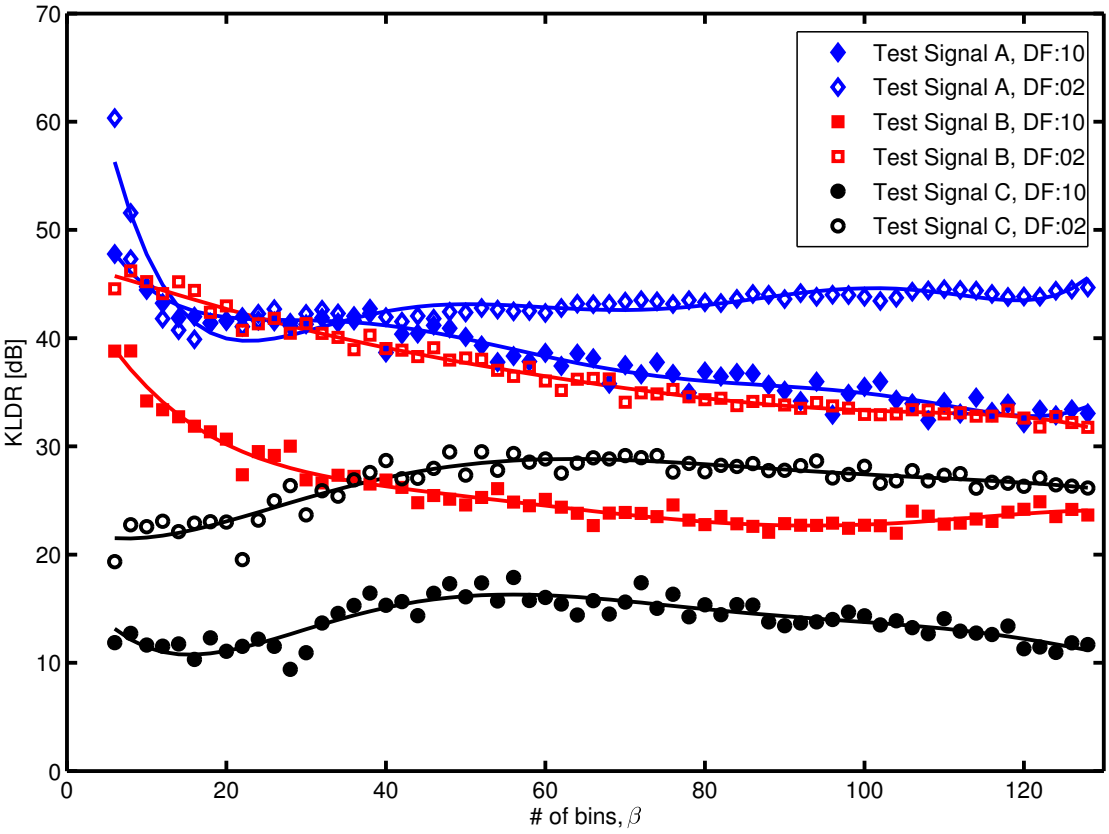


Figure 4.13: *Performance of the KLD based anomaly detection scheme under various histogram bin counts. Solid lines are spline fits of the data and used to highlight trends. The window size is set at* 256 μs *and the decimation factors used are* 2 *and* 10*. Generally, a larger number of bins lead to poorer performance due to increased noise in the estimates. However, the rate of change is small and therefore the drop in performance is insignificant over a wide range of bin resolutions.*

It is the second reason that generally makes it straightforward to choose a suitable D_{th}.

4.3.2.4 Histogram resolution

Histogram bin resolution, represented by the parameter β, is of relevance to both of the proposed algorithms. Figure 4.13 shows how performance of the KLD based algorithm is affected by the choice of the number of histogram bins used to classify the input data and estimate the PMFs. The window length is set at 256 μs and results are shown for decimation factors of 2 and 10. Once again, the smaller decimation factor provides uniformly improved performance over the entire range of β values. The only exception is signal A where the performance for smaller β values appear to be independent of the decimation factor used. The reasons for this is the relatively long duration of the anomaly – as explained previously

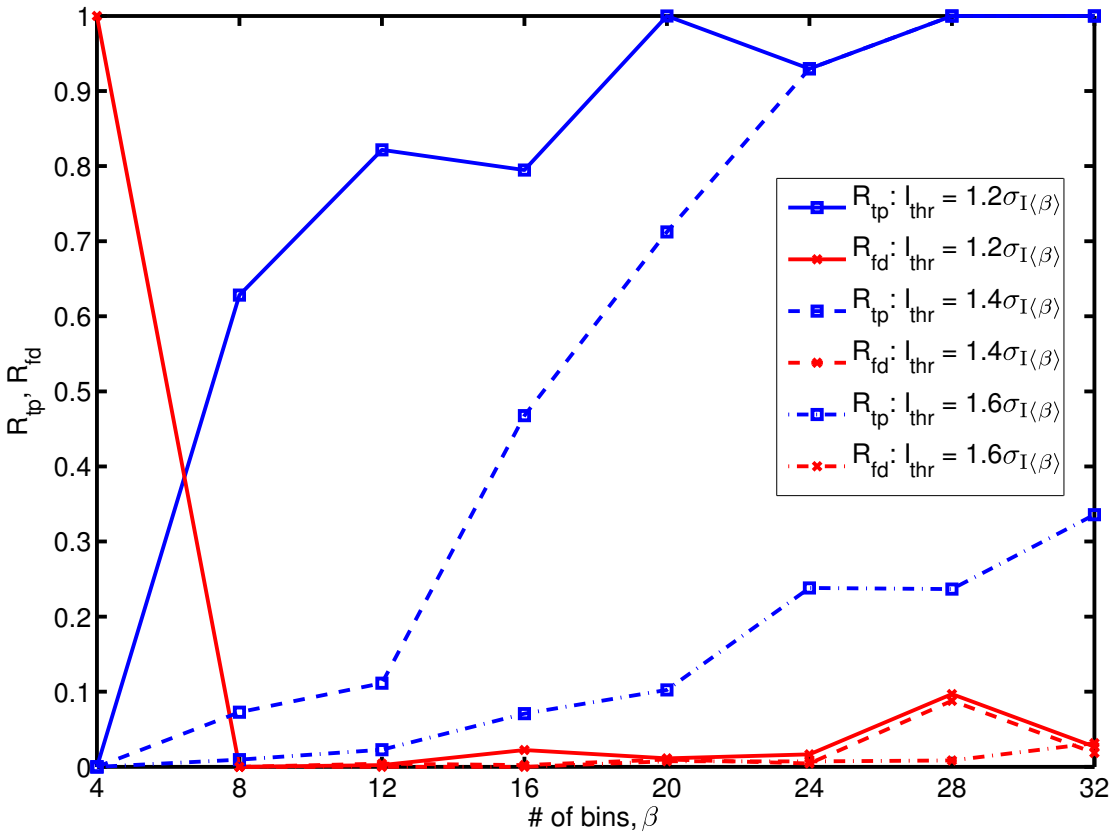


Figure 4.14: *Effect of histogram resolution on detector R_{tp} and R_{fd}. The cluster length is* 3 *and thresholds are* $1.2\sigma_{I\langle\beta\rangle}$, $1.4\sigma_{I\langle\beta\rangle}$ *and* $1.6\sigma_{I\langle\beta\rangle}$. *$R_{tp}$ improves with resolution while R_{fd} deteriorates. $\beta = 20$ with a threshold of $1.2\sigma_{I\langle 20\rangle}$ yields the best performance.*

in Section 4.3.2.2.

The only trend common to all three signals is that the performance changes little with increasing bin numbers, with signal C showing an optimum in the vicinity of $\beta = 55$. This indicates that the behaviour observed is specific to the type of anomaly present in a signal. While the number of bins utilised does not appear to have a significant impact on the performance of the scheme for a fixed amount of data, the decrease observed is due to noisier PMF estimates that are obtained for larger values of β. Noisy PMFs lead to larger background KLD values and hence a reduced KLDR.

Figure 4.14 shows how the R_{tp} and R_{fd} varies for the test signal (Figure 4.9(a)) with the number of histogram bins utilised. The cluster length utilised is 3 and I_{th} of $1.2\sigma_{I\langle\beta\rangle}$, $1.4\sigma_{I\langle\beta\rangle}$ and $1.6\sigma_{I\langle\beta\rangle}$ are used for comparison. The plot reveals that when $\beta = 4$, R_{tp} is zero and R_{fd} is unity for all thresholds tested. This is because the sensitivity is very low and no anomalies can

be detected ($R_{tp} = 0$). Events exceeding the threshold are low-probability signal events and hence are all false positives ($R_{fd} = 1$). As β is doubled to 8, the resolution improves and there is a corresponding increase in the R_{tp}. The R_{fd} also drops to a negligibly small value. As β is increased further, the R_{tp} increases due to better detector resolution. The R_{tp} improvements come at a cost however – the detector is more susceptible to noise at higher resolutions. This is evident from the gradual increase in the R_{fd}.

Comparison between the three detection thresholds reveals that a higher R_{tp} is achieved with a lower threshold. Unfortunately, this also leads to a higher R_{fd}. This behaviour is in accordance with the explanation provided in Section 4.3.2.3.

4.3.2.5 Cluster size

In order to investigate the impact of the information cluster length, N, on the ICA based detector, β is set at 16 and the analysis is performed for I_{th} of $1.2\sigma_{I\langle 16\rangle}$, $1.4\sigma_{I\langle 16\rangle}$ and $1.6\sigma_{I\langle 16\rangle}$ on the test signal shown in Figure 4.9. The result of the analysis is shown in Fig 4.15.

The significance of clustered anomaly detection is immediately obvious. With $N = 1$, when clustering is not performed, there is an overwhelming number of false positives. This is indicated by the high R_{fd}. As soon as clustering is applied by setting $N = 3$, a dramatic drop in the R_{fd} is observed – showing that even minimal anomaly clustering is sufficient to yield a massive improvement in detector performance. By lowering the R_{fd}, clustering also allows a lower I_{th} to be used to achieve a higher R_{tp}.

The impact of anomaly clustering on R_{tp} for a given I_{th} is relatively low. As cluster size is increased, a gradual decrease is observed in the R_{tp}. This is expected since larger cluster sizes lead to missed detections around the edges of the interference patterns. The plot also shows that higher thresholds lead to lower R_{tp} for a given cluster size. This is also expected since a higher information content threshold leads to a higher number of missed detections.

From the analysis performed on the test signals, it is clear that it is challenging to determine a set of parameters that are inherently optimal for the anomaly detection algorithms proposed. This is due to the fact that the optimal parameter set depends on a number of problems

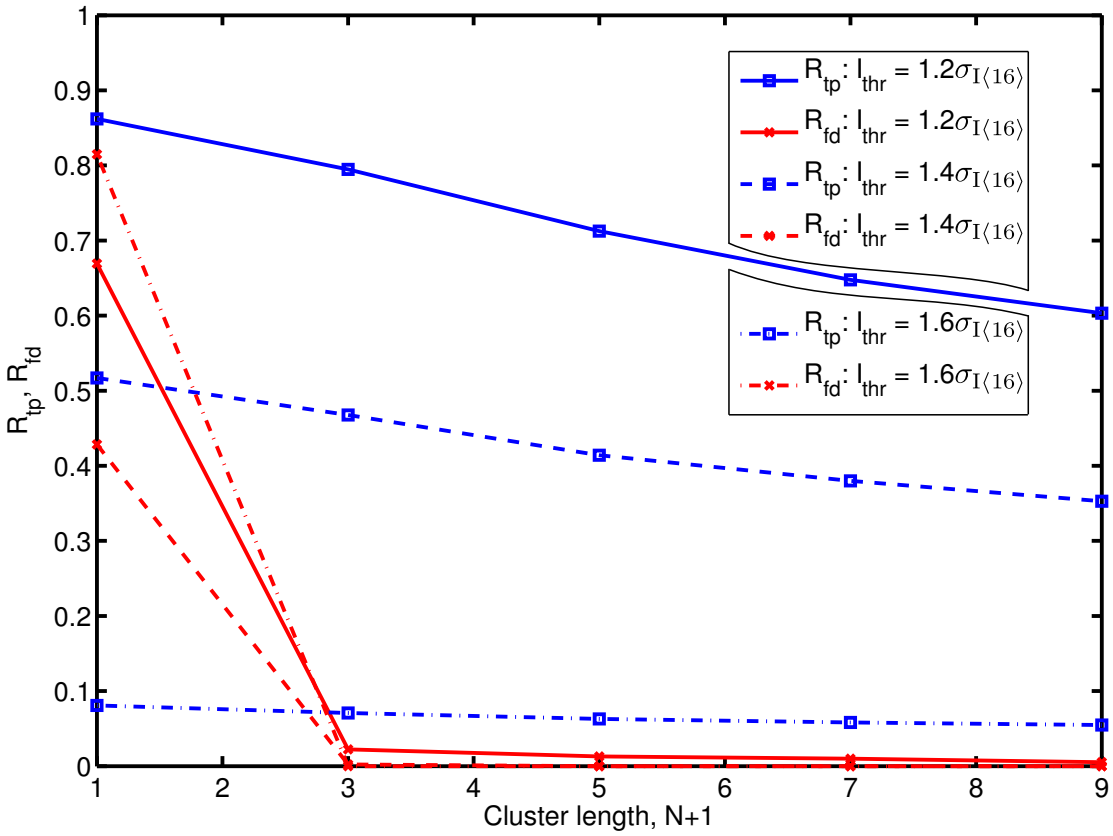


Figure 4.15: *Impact of cluster length on detector R_{tp} and R_{fd}. The number of histogram bins used is 16 and the thresholds utilised are $1.2\sigma_{I\langle 16\rangle}$, $1.4\sigma_{I\langle 16\rangle}$ and $1.6\sigma_{I\langle 16\rangle}$. Even the smallest cluster length ($N = 2$) is shown to provide a significant improvement in the R_{fd}.*

specific factors such as duration of the anomaly, dynamic range of the signal, etc. It may be possible to develop adaptive variants of the algorithms that automatically find the best parameter combinations subject to some performance criterion but that is beyond the scope of this thesis.

The ICA based algorithm is particularly sensitive to the parameters utilised. Generally, it is seen that parameter values that increase the R_{tp} (good) often also lead to an increase in the R_{fd} (bad) and vice versa. Trade-offs must therefore be made to meet the required detector performance characteristic (low R_{fd}, moderate R_{tp} or high R_{tp}, moderate R_{fd}). A moderate number of bins ($\beta = 20$), small cluster size ($N = 2$), and a threshold of $1.2\sigma_{I\langle 20\rangle}$ bits ($m = 1.2$) provides a good balance between R_{tp} (1.0) and R_{fd} (0.01) for this particular test vector (signal D).

The KLD based algorithm on the other hand is much more robust with respect to the param-

eter combinations utilised. The results clearly show that performance better than 30 dB of KLDR can be easily obtained with reasonable choice of parameter values such as a window length of 256 μs, $\beta = 32$ and decimation factor of 2.

4.4 Chapter summary

Two complementary anomaly detection algorithms utilising information theoretic measures are presented. Both algorithms are simple to implement and require little *a priori* information regarding the signal under test. Demodulation of the signal is also not required since the algorithms are capable of processing the baseband signal envelope itself in real-time. The information content analysis based method is capable of detecting singular anomalous events while the Kullback-Leibler divergence based method is also able to detect otherwise nominal events that are anomalous purely due to context (e.g. misaligned signal frames). In order to provide this context aware detection of anomalies, the KLD based algorithm requires the input signal to be periodic.

Analysis of a number of test signals captured over the air show that the KLD based scheme is successful at detecting all anomalies known to be present. The only information required by the algorithm in advance is the signal period. Extensive tests using a software implementation of the algorithm demonstrates that it is robust with respect to parameter choices since satisfactory performance can be maintained with reasonable parameter values even when the input is severely under-sampled. With PMF estimation window sizes of 256 μs, 32 histogram bins and factor of 10 under-sampling, KLDR of 25 dB or better can be achieved depending on the anomaly present.

With a simple change in the position of the PMF estimation windows, the KLD based algorithm can also be used for the detection of temporal boundaries in a data stream. No *a priori* information regarding the signal under test is required to perform the procedure. For periodic signals, the timing pattern obtained from clean training data can be further processed using autocorrelation to estimate the underlying period required by the anomaly detection algorithm. Tests on the available data sets have produced results that are indeed promising, with period estimates that are more than 99% accurate.

Unlike the KLD based method, the algorithm utilising ICA for anomaly detection does not require the input signal to be periodic. The only piece of information that is needed in ad-

vance is the set of reference event probabilities. A training data set known to be clean can be used to obtain the reference probabilities prior to analysis. The complexity and memory requirements of the algorithm are also very low.

It is clear from tests carried out using a software implementation of the algorithm that performance of the system is strongly affected by the choice of parameters such as histogram resolution, threshold and cluster size. Impact of these parameters on the detector performance by means of the true positive rate and false discovery rate has been analysed and guidelines for appropriate values have been provided. It is shown that a true positive rate of 100% and false discovery rate of 1% – guaranteeing zero missed detections with very few false positives – is possible for the signal tested with a suitable set of parameter choices.

Chapter 5
Hardware for anomaly detection using Kullback-Leibler divergence

5.1 Introduction

ONE of the salient features of technological progress is the ever increasing amount of data generated by systems and devices. This torrent of information is not only due to improvements in resolution, capacity and complexity of existing systems; the trend towards greater environmental awareness (e.g. cognitive radio [10, 11]) requires the use of extensive sensor networks that contribute to the large volume of data that need to be processed. Due to the overwhelming amount of information present, it is no longer efficient or possible to manually analyse the data in case of faults in a relatively complex system. Smart anomaly detection algorithms that can automatically locate unusual events that may exist alongside nominal samples in a dataset are therefore needed. Anomaly detection is a well-studied procedure that is widely utilised in a diverse range of application fields including biomedical science [21], finance [20] and network security [19]. An extensive survey of current anomaly detection techniques and application domains is presented in [22].

Real-time online processing of high-speed data is a feature that is highly desirable in any anomaly detection algorithm since it affords the ability to respond immediately to detected events. Clearly, the algorithm utilised must be simple, fast and capable of being implemented efficiently on a reasonable (commodity) hardware platform to achieve that goal. Unfortunately, the vast majority of the approaches described in the aforementioned survey are unsuitable for real-time implementation due to computational complexities that are quadratic in the number of input data points. To overcome this challenge, a pair of simple algorithms based on information content analysis (ICA) [90] and Kullback-Leibler divergence (KLD) [73] have been developed and presented in Chapter 4. The goal of this chapter is to discuss a field-programmable gate array (FPGA) oriented implementation of the KLD based method in detail [26].

KLD is a convenient and robust method of measuring the difference between two data sets in a statistical sense. Its definition and use in detecting anomalies in periodic signals has been discussed previously in Section 4.2 of Chapter 4. Regardless, the definition of KLD and the proposed anomaly detection algorithm are reintroduced here in brief for completeness. Given two probability mass functions (PMFs) p_n and q_n at time n, KLD can be computed using [73]

$$D\left(p_n \,||\, q_n\right) = \sum_{x \in X} p_n(x) \log_2 \frac{p_n(x)}{q_n(x)} \tag{5.1}$$

where X is the domain of the PMFs. The cardinality of X reflects the number of histogram bins, β, used to estimate the PMFs. $D\left(p_n \,||\, q_n\right)$, also referred to as D_n for brevity, is always a positive real number measure in bits. It is zero if and only if both PMFs are identical.

The proposed anomaly detection algorithm (described using a flowchart in Figure 4.2) exploits the cyclic temporal structure of periodic signals by using two data capturing windows separated by the signal period T_p. The windows provide the data sets P_n and Q_n, and hence the PMFs p_n and q_n. Since segments of the signal separated by the period are expected to have similar statistical distributions under nominal operating conditions, the KLD between the windows is expected to be close to zero. However, if it is found to be larger than some predefined threshold D_{th}, an anomaly may be present. An illustration of the algorithm is provided in Figure 4.3.

Thorough tests of performance (presented in Section 4.3) of the proposed anomaly detection algorithm against a number of recorded wireless test signals show that the method is indeed capable of detecting anomalous events. The results shown have been obtained by post-processing the data using a MATLAB based software implementation of the anomaly detection algorithm. In order to experiment with real wireless signals in real-time, a hardware based implementation of the algorithm is needed.

Analysis of the anomaly detection algorithm performed in Section 4.2.2 and summarised in Table 4.1 reveals that the algorithm requires $2|P_n| + 6\beta$ operations and sufficient storage for $|P_n| + \psi + 2\beta$ elements at each time step. ψ is the total number of discrete samples within one signal period, T_p. The analysis indicates that both the computational complexity and memory requirements of the procedure scale linearly with the input data size $|P_n|$ and the number of histogram bins utilised (β). While memory isn't necessarily a problem for any reasonable value of $|P_n|$ (e.g. 10^5) the computational complexity certainly is. A few

megabytes of memory is only a small fraction of the total amount available on a modern hand-held device such as the Nokia N900 which can provide up to 1000 megabytes of memory to applications [100]. In contrast, even a high performance application processor such as the ARM Cortex-A9 cannot provide the throughput necessary to process a high-speed signal in real-time. For examples, a signal with 10 MHz sample rate and $|P_n|$ of 10^5 requires at least 2×10^{12} operations per second for real-time analysis – this is far beyond the capacity of a ARM Cortex-A9 device which can only provide around 2×10^9 instructions per second running at a speed of 1 GHz [101]. Therefore, if at all feasible, a direct translation of the anomaly detection algorithm into a hardware implementation would be inefficient, inflexible and computationally very expensive.

The inefficiency arises from the fact that at each successive time instance, the PMFs and the KLD are completely recalculated; even though for this particular application scenario it is only a single sample that changes in each of the data sets. The inflexibility comes from the fact that the computational complexity depends on β, implying that a direct mapping of the algorithm would be limited by the initial choice of the histogram bin resolution. Finally, at each time step, the KLD equation (5.1) requires the evaluation of a number of logarithms and divisions – both of which can be very costly to implement with high precision in hardware [102–104]. Clearly, a number of optimisations must be made before the anomaly detection algorithm can be implemented on a reasonable hardware target such as the Xilinx Virtex-4 ML402 SX XtremeDSP Evaluation Platform.

The modifications necessary for an efficient implementation of the KLD based anomaly detection algorithm in hardware are described in the following section (Section 5.2). It includes a discussion of application specific optimisations for the KLD equation, efficient methods of histogram/PMF computation and hardware reuse for implementation cost minimisation. Section 5.3 follows with a description of the architecture developed for FPGA targeting implementations of the algorithm. Experimental results obtained from the actualised platform are shown in Section 5.4 while Section 5.5 concludes with a summary of the chapter.

5.2 Techniques for efficient implementations

There are a number of aspects of the anomaly detection algorithm that can be optimised for a fast and efficient implementation. The first is the computation routine used to evaluate

KLD itself. Additionally, it is also necessary to apply optimised methods for computing the PMFs and base-2 logarithms since these functions are an integral part of the KLD equation. While implementing the algorithm it is also necessary to ensure that the hardware resource requirements (memory, arithmetic units, etc.) are reasonable. A discussion of the techniques employed for efficient KLD computation, PMF estimation, base-2 logarithm evaluation and hardware resource sharing for cost minimisation follows.

5.2.1 Optimised KLD computation

The goal is to implement the KLD based anomaly detection algorithm for periodic signals in FPGA hardware. As it is a well-defined problem, application specific changes and optimisations can be made to the KLD computation routine to reduce complexity and significantly boost efficiency. The first of these modifications however is completely general and simply exploits the identity $\log\left(\frac{a}{b}\right) = \log(a) - \log(b)$

$$\begin{aligned} D_n &= \sum_{x \in X} p_n(x) \log_2 \frac{p_n(x)}{q_n(x)} \\ &= \sum_{x \in X} pn(x) \left\{ \log_2 \left[p_n(x) \right] - \log_2 \left[q_n(x) \right] \right\} \end{aligned} \tag{5.2}$$

to exchanges the division operation for a logarithm and subtraction. It might appear that this substitution is useless since it has been stated previously that both division and logarithm are generally costly functions to implement in hardware. However, due to properties inherent to this particular problem (computing KLD), the necessary base-2 logarithm values can be obtained in a highly efficient fashion. The approach utilised is described in Section 5.2.3.

The second set of changes are indeed application-specific and modify the way KLD is computed.The purpose of the algorithm is to analyse periodic signals by means of two sliding windows. This implies that the data samples encompassed by the two windows change very little between two successive time instances. In fact, all except one of the samples remain unchanged in each of the two windows. Since most of the data remain unchanged, the histogram and hence the PMF estimates also change little between two consecutive time instances. Change of a single value in a data set leads to changes in two bins of the associated histogram: the frequency count in the first is reduced by one to clear away the previous value while the count in the latter is increased by one to accommodate the current value of the changed sample. Therefore at any given time only a maximum of four histogram bins need

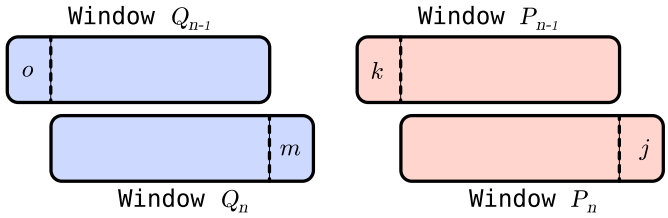


Figure 5.1: *At any given time, only four histogram bins need to be updated (two per window). Bins incremented (fresh samples) are denoted by j and m while bins decremented (old samples at the end of the window) are denoted by k and o.*

to be updated to yield the new set of PMFs. Figure 5.1 provides an illustration. Bin indices j and m correspond to the freshly acquired samples while k and o correspond to the samples removed from the data sets. Only a maximum of 4 additions and 4 divisions are now required to obtain the updated PMFs – leading to a significant reduction in the complexity of the algorithm.

By following the same set of arguments it can be seen that the KLD values also change very little between subsequent time steps for this particular application. As a result, it is possible to reuse D_{n-1} to calculate D_n using a difference equation of the form $D_n = D_{n-1} + \Delta_n$ where Δ_n is an adjustment factor that accounts for the changed PMFs. The adjustment factor is calculated by adding together terms in (5.2) containing $p_n(\{j,k,m,o\})$ and $q_n(\{j,k,m,o\})$, and subtracting terms containing $p_{n-1}(\{j,k,m,o\})$ and $q_{n-1}(\{j,k,m,o\})$. The complete difference equation version of the KLD equation (5.2) is then

$$\begin{aligned}
D_n &= \sum_{x\in X} p_n(x)\left\{\log_2\left[p_n(x)\right] - \log_2\left[q_n(x)\right]\right\} \\
&= D_{n-1} + \Delta_n \\
&= D_{n-1} - p_{n-1}(j)\left\{\log_2\left[p_{n-1}(j)\right] - \log_2\left[q_{n-1}(j)\right]\right\} \\
&\quad + p_n(j)\left\{\log_2\left[p_n(j)\right] - \log_2\left[q_n(j)\right]\right\} \\
&\quad - p_{n-1}(k)\left\{\log_2\left[p_{n-1}(k)\right] - \log_2\left[q_{n-1}(k)\right]\right\} \\
&\quad + p_n(k)\left\{\log_2\left[p_n(k)\right] - \log_2\left[q_n(k)\right]\right\} \\
&\quad - p_{n-1}(m)\left\{\log_2\left[p_{n-1}(m)\right] - \log_2\left[q_{n-1}(m)\right]\right\} \\
&\quad + p_n(m)\left\{\log_2\left[p_n(m)\right] - \log_2\left[q_n(m)\right]\right\} \\
&\quad - p_{n-1}(o)\left\{\log_2\left[p_{n-1}(o)\right] - \log_2\left[q_{n-1}(o)\right]\right\} \\
&\quad + p_n(o)\left\{\log_2\left[p_n(o)\right] - \log_2\left[q_n(o)\right]\right\}. \qquad (5.3)
\end{aligned}$$

Operations	Σ	$\times$	$\div$	log	Total
	20	8	16	16	60
Memory	$\lvert P_n \rvert + \psi + 2\beta$				

Table 5.1: *Runtime operational complexity and memory requirements of the differential KLD based anomaly detection algorithm.*

This is the general form of the differential KLD (DKLD) equation where the four bin indices j, k, m, and o are assumed to be unique. If that is not the case then each of the duplicate terms in the equation must be set to zero before D_n is calculated.

It is clear from (5.3) that the computational complexity of the algorithm need no longer be dependant on the number of histogram bins utilised in estimating the PMFs. Assuming that D_{n-1} is available, only sixteen additions/subtractions, sixteen logarithms, and eight multiplications are needed to calculate D_n – regardless of the value of β. This opens the path for a fixed complexity, flexible and efficient implementation that can be easily updated to accommodate arbitrary histogram resolutions.

The computational complexity and storage requirement of the anomaly detection algorithm using DKLD is shown in Table 5.1. Comparisons against the unmodified, direct translation version of the algorithm (Table 4.1) reveals that while memory utilisation remains unchanged, there is a vast difference in the number of operations required. Regardless of the window size and histogram bin count, sixty operations are all that is needed to compute the KLD. In addition to the sixteen additions/subtractions, sixteen logarithms and eight multiplications required for computing DKLD (5.3), four more additions/subtractions are required to update the affected histogram bins and sixteen divisions are required to obtain the necessary PMFs at times $n-1$ and n.

5.2.2 Histograms and PMFs

In its simplest form, a histogram computation routine consists of a number of alternative execution paths – each with a comparator to check against the bin range and an accumulator to store the frequency count. An illustration of the architecture is shown in Figure 5.2. The number of parallel branches required is equivalent to the number of histogram bins utilised. Each comparator checks the input sample against its predefined bin ranges and produces a

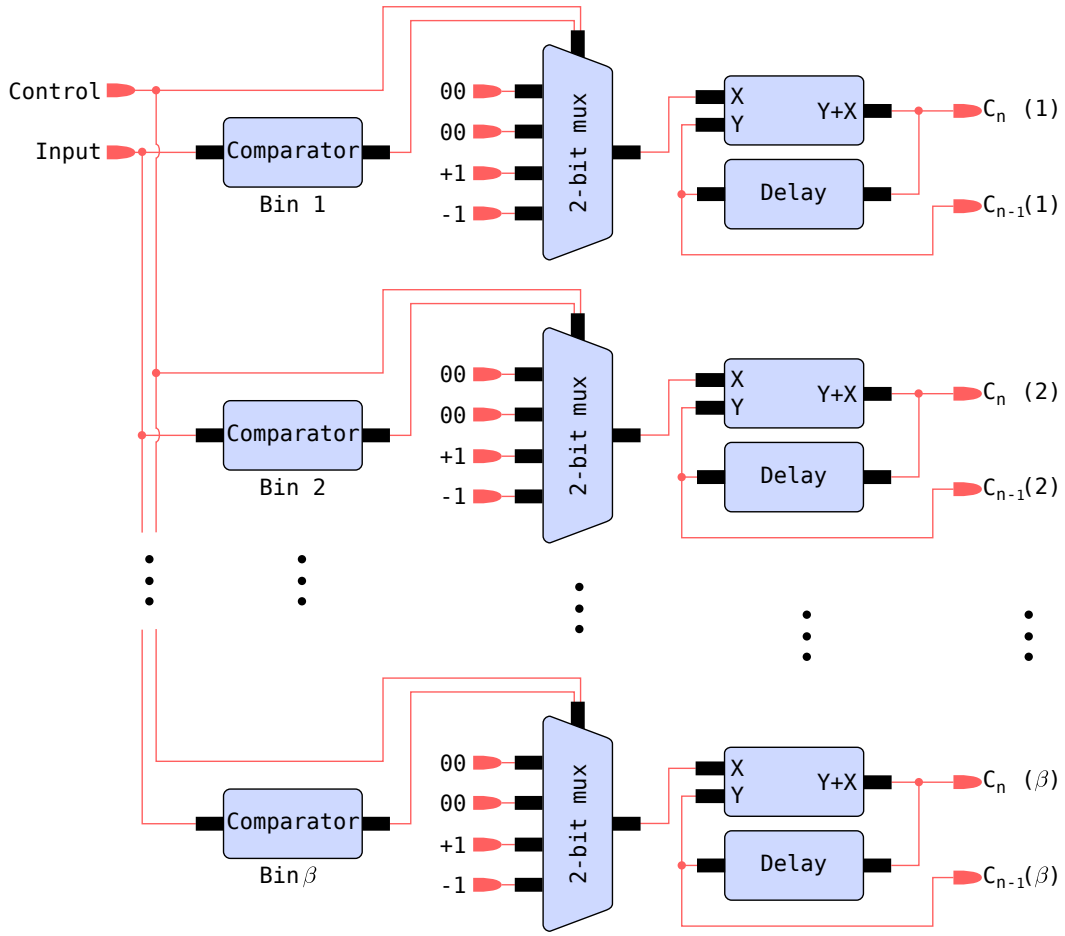


Figure 5.2: *Illustrative architecture of a simple histogram module. Each branch consists of a comparator, multiplexer and accumulator. The number of branches required is equivalent to the number of histogram bins (β) utilised. Each comparator checks the input against the bin range specified and produces a single bit output of '1' for a match and '0' otherwise. The comparator output is used along with a control bit to select the output (+1, 0 or -1) of a 2-bit multiplexer. This provides a mechanism for controlling whether the frequency count is incremented, decremented or left unchanged.*

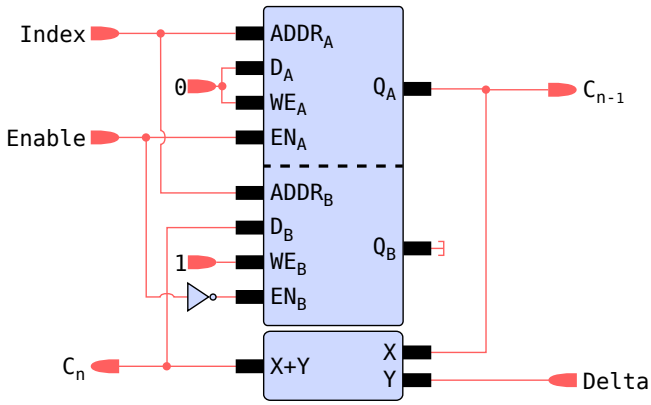


Figure 5.3: *Basic architecture of a dual-port RAM based histogram module. Bin indices are used as addresses and an externally supplied integer* `Delta` *is used to modify the frequency count associated with a given histogram bin. The module simultaneously outputs the frequency count associated with supplied bin index at times $n-1$ and n. The* `Enable` *signal ensures that only one port of the RAM is active at a time – avoiding read-write collisions.*

single bit output of '1' if a match is found or '0' otherwise. The comparator output is then used together with a single bit control signal to select between four output values of a 2-bit multiplexer that produces '0', '+1' or '-1' depending on the selector bit sequence. This is used to control whether the frequency count in the accumulator is increased, decreased or left unchanged. By using a delay element for each branch, the frequency counts for all of bins at times $n-1$ and n can be obtained simultaneously. While such an implementation is indeed simple, it is clearly inefficient and inflexible. The main problem is that the design does not readily support arbitrary bin resolutions. Therefore an alternative histogram architecture must be developed.

A solution is found through the use of random access memory (RAM). A histogram can be easily implemented by associating each bin with a unique address of the memory. Bin pre-loading (Section 4.2.1) can be achieved by simply initialising the memory with the required pre-load value. An arbitrary number of bins can also be supported by simply changing the size of the addressable space. This is the facility that is needed for a truly flexible implementation of the histogram function. An example of such a design is shown in Figure 5.3. The proposed architecture is highly efficient and economical – requiring just a block of memory and an accumulator to implement. It is based around a dual-port RAM (DPRAM) module for convenience. It is simply a RAM module that has two access ports to the same physical memory space – permitting a design that is compact and efficient.

The bin index is computed from the input sample and used to address both ports ($\texttt{ADDR}_\texttt{A}$ and $\texttt{ADDR}_\texttt{B}$) of the DPRAM. Port A is used exclusively for read operations and provides the stored frequency count ($\texttt{C}_{n-1}$) associated with the supplied bin index/address. The count obtained at the output ($\texttt{Q}_\texttt{A}$) is that at time $n-1$. In order to obtain the updated count ($\texttt{C}_\texttt{n}$) at time n, the value specified by `Delta` is added to $\texttt{C}_{n-1}$. `Delta` can have one of three possible values: '+1' for an increase in the frequency count, '-1' for a decrease and finally '0' for no change. Once the updated frequency count $\texttt{C}_\texttt{n}$ has been calculated, it is stored back into memory by means of the second port of the DPRAM. This interface is used exclusively for writing to the memory. $\texttt{ADDR}_\texttt{B}$ is still driven by the supplied bin index and the data to be stored, $\texttt{C}_\texttt{n}$, is provided via the data input bus $\texttt{D}_\texttt{B}$. Since port A is used only for reading while port B is used for writing, their write enable pins ($\texttt{WE}_\texttt{A}$ and $\texttt{WE}_\texttt{B}$) are held at logical `LOW` (Boolean *false*) and `HIGH` (Boolean *true*) respectively. The data bus of port A ($\texttt{D}_\texttt{A}$) is also held `LOW` since it is not used for writing to the memory. In order to prevent the read/write collisions that would otherwise arise due to simultaneous read/write access to a single memory address, the enable pins ($\texttt{EN}_\texttt{A}$ and $\texttt{EN}_\texttt{B}$) of the two ports are joined together using a `NOT` gate and driven using a single control signal that activates the ports in sequence. For each input `Index` a read operation is followed by a write operation, implying that the architecture must run at twice the input sample rate.

While the DPRAM based implementation is certainly more flexible compared to the simple conditional branching method previously described, it does have two minor disadvantages. The first is the fact that it requires a clock frequency that is twice as fast for a given input sample rate. The second is that at any given time, only the frequency counts associated with a single bin (`Index`) is available instead of the entire set. Any additional bin frequency count must be retrieved by addressing the DPRAM sequentially. However this is not as big a problem as it seems – at any given time, DKLD only requires maximum of 4 bin frequency counts out of β. Since two of the indices (e.g. j and k for P_n) must always be processed, the actual overhead compared to the simple branched method is only two additional lookups. The ability to compute histograms of arbitrary resolution with a semi-fixed hardware cost* far outweighs the apparent disadvantages of the proposed DPRAM based histogram computation method.

In order to compute the PMF from the histogram, the frequency count of each bin must

*As stated previously, only the size of the addressable memory depends on the bin count required.

be divided by $|P_n|$. As stated previously, general-purpose, full-precision division operations can be very costly to implement in hardware. However, with a simple constraint on the denominator, that cost can be reduced to virtually zero. If the data window size, $|P_n|$, is always chosen to be a power of two (PoT), i.e.

$$|P_n| = 2^E, \quad E = 0, 1, 2, \ldots$$

a division simply becomes a left shift in the position of the binary point in the fixed-point representation of the numerator. As this operation does not require any hardware, it is essentially *free*. Therefore, through the imposition of a simple constraint on an algorithm parameter, the number of divisions required by the anomaly detection algorithm can be reduced to zero.

5.2.3 Base-2 logarithms

Apart from the division operation, the only other function that poses a challenge for the anomaly detection algorithm implementation is the base-2 logarithm function. Although it can be implemented via a number of methods such as polynomial approximations and the well-known CORDIC [105] algorithm, such approaches are unnecessarily complex for this particular application. Instead of performing arithmetic computations, the necessary results can be obtained directly from a lookup table. Once again, intimate knowledge of the problem is used to develop an efficient application-specific solution.

The purpose of the module is to compute the base-2 logarithm of PMFs. As a result, the input is guaranteed to have a value between 0 and 1. This tight bound on the input implies that a lookup table of $\log_2(\mathbb{P})$, where $\mathbb{P}$ is in [0,1] can be used. Clearly it is neither possible nor practical to store the continues range of base-2 logarithm values for inputs between 0 and 1; therefore a maximum table size must be defined. Generally, this size will then dictate the precision available for an arbitrary input. However if the input is sampled and represented using a fixed-point format, a table can provide results that are precise within the accuracy bounds of the chosen output precision – as long as the table is large enough to contain results for every value representable using the chosen fixed-point word size for the input.

When bin pre-loading is used, no bin in the PMF can have a value that is exactly unity or zero, regardless of the distribution. As a result, the most significant bit (MSB) is always zero and the fractional bits are never all zero in a fixed-point representation of the input. Consequently,

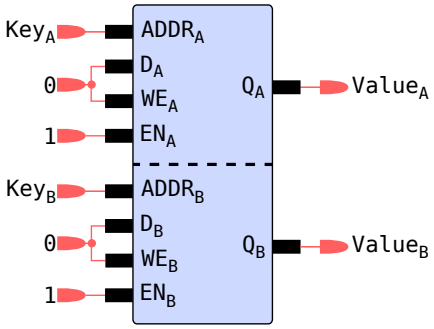


Figure 5.4: *Basic architecture of a dual-port RAM based lookup table. It is possible to perform two independent read operations simultaneously. The write enable (*$\texttt{WE}_\texttt{A}$ *and* $\texttt{WE}_\texttt{B}$*) and port enable (*$\texttt{EN}_\texttt{A}$ *and* $\texttt{EN}_\texttt{B}$*) pins are held at logic levels* `LOW` *and* `HIGH` *respectively in order to emulate the function of traditional ROM devices.*

for an M-bit word with $M-1$ fractional bits, a lookup table with $2^{M-1}-1$ entries is sufficient. The table that needs to be pre-computed is

$$\log_2\left(\frac{i}{2^{M-1}}\right); \quad i = 1,2,\ldots,2^{M-1}-1.$$

Such a table can be directly addressed using the $M-1$ fractional bits of the input in order to retrieve the required base-2 logarithm value. Since the table is pre-computed, zero arithmetic operations are needed to evaluate the base-2 logarithm function at runtime. Computational complexity is essentially traded for a small increase in the total required memory.

Lookup tables are usually based on read only memory (ROM) devices that are loaded with the relevant data at initialisation. These are typically single-port memory devices that allow only one lookup operation at a time. Since DKLD requires sixteen base-2 logarithm lookup operations at any given time, as many *identical* lookup tables would be needed to implement the anomaly detection algorithm. Therefore it is perhaps a better approach to base the lookup tables on DPRAM devices instead. By allowing two simultaneous and independently addressed read operations from the same physical memory cluster, a single DPRAM can provide the same capability as two traditional ROM devices while requiring only half the hardware resources.

An illustration of the DPRAM based solution to lookup tables is shown in Figure 5.4. It shows that the device can emulate the function of two separate but identical ROM devices by allowing simultaneous read operations on each of the two ports. The device is enabled perpetually by holding both enable pins ($\texttt{EN}_\texttt{A}$ and $\texttt{EN}_\texttt{B}$) at logic level `HIGH`. Unlike the DPRAM

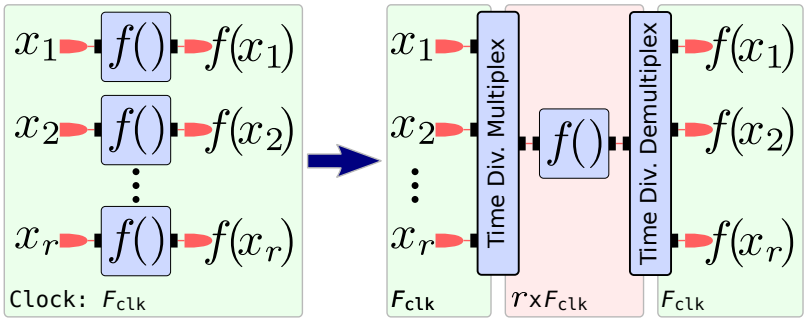


Figure 5.5: *Hardware utilisation efficiency can be improved by using resource sharing. When multiple parallel but identical execution path exist in a design, it is possible to collapse them into fewer serial paths – resulting in a reduction in the resources required. However, higher clock speed and control logic in the form of time division multiplexing and demultiplexing blocks are required to accommodate the modified design.*

based histogram module, address collision is not a problem since both ports are used for read operations only. In order to enforce this restriction the device is rendered read-only by holding both write enable pins ($\texttt{WE}_\text{A}$ and $\texttt{WE}_\text{B}$) at logic level `LOW`.

5.2.4 Resource sharing

There is one final technique that can be exploited to boost the hardware resource utilisation efficiency of the anomaly detection algorithm implementation. Known as resource sharing (RS), it is a method of reducing resource usage by reusing hardware blocks for performing identical sets of operations. It is typically applicable when multiple parallel but identical execution paths exist in the design. By serialising the parallel execution paths or operations, RS permits the use of fewer resource units to perform the same task as that originally intended. The reduction in resources required comes at a price however – it is necessary to increase the system clock speed by a factor equivalent to the reuse ratio in order to ensure that the serial processing of the data set takes no longer to execute than the original. Since some additional overhead in the form of control logic to handle parallel $\leftrightharpoons$ serial conversions is also required, RS should only be utilised when a significant reduction in resource utilisation is expected.

The concept of RS is illustrated in Figure 5.5. If a design running at a speed of F_{clk} MHz contains r identical and parallel execution paths to compute some function $f()$, then it is possible to collapse the design to a single execution path by serialising the input and running $f()$ at a speed of rF_{clk} MHz. The parallel/serial and serial/parallel conversion (time division

Operations	Σ	$\times$	$\div$	log	Total
	20	8	0	0	28
Memory	$\lvert P_n \rvert + \psi + 2\beta + \mathbb{L}$				

Table 5.2: *Runtime operational complexity and memory requirements of the fixed-point DKLD based anomaly detection algorithm.*

multiplexing/demultiplexing) blocks provide the necessary control logic and interface the different clock domains by appropriately adjusting the input/output (I/O) sample rates.

It is clear from (5.3) that the implementation can benefit significantly from RS. At any given time it is necessary to compute a total of sixteen additions, eight multiplications and sixteen base-2 logarithms. While RS can be applied to reduce the number of accumulators and multipliers required, it is undesirable due to the added complexity and relative abundance of such parts on modern processor devices. It is far more attractive to minimise the memory required by reducing the number of base-2 logarithm lookup tables utilised in the implementation. With a RS factor of eight ($r = 8$), a single DPRAM based lookup table is sufficient for simultaneously evaluating all sixteen base-2 logarithms.

The computational complexity and storage requirement of a fixed-point DKLD (FPDKLD) based implementation of the anomaly detection algorithm with PoT constraint on $|P_n|$, factor eight RS and a DPRAM based base-2 logarithm lookup table is shown in Table 5.2. $\mathbb{L}$ is the size of the lookup table used and depends on the precision required. Comparisons against Table 4.1 reveal that the use of application specific optimisations and simple efficiency improvement techniques can lead to a tremendous reduction in the computational complexity of the algorithm. The price for this is an increase in the operating frequency and memory utilisation. It is particularly important to note that the computation complexity of the anomaly detection algorithm has been transformed from an expression that scales linearly with $|P_n|$ and β to one that is a small constant. The twenty-eight operations now required to evaluate KLD consist of only additions and multiplications – functions that can be evaluated quickly and efficiently on-chip using dedicated hardware arithmetic logic units.

5.3 Architecture for an FPGA target

While the proposed anomaly detection algorithm can be implemented using either digital signal processing (DSP) or FPGA hardware, the latter is preferred due to the I/O intensive nature of the application. DSPs are better suited to problems that tend to be more computation intensive with relatively low raw data throughput. The massively parallel nature of FPGA architectures is also ideal for implementing the anomaly detection algorithm since it allows the DKLD equation and PMFs to be computed in a completely parallel fashion – greatly reducing the execution time of the algorithm compared to a sequentially processed implementation.

The architecture described targets the Xilinx Virtex-4 family of FPGA devices and is implemented using the SystemGenerator 10.1 tools in conjunction with MATLAB/Simulink R2007a from Mathworks. While the effort required to modify this design for other FPGA families and development tool-chains can vary considerably depending on the choice of hardware and software, the process itself is straightforward. Since no Xilinx-specific constructs are used in the core design, the functionality provided by the modules can be duplicated using a third party tool such as Synplify DSP (effort required: low) or low level hardware description language (effort required: high).

Top level design

The top level structure of the design is shown in Figure 5.6. The **In** and **Out** gateway ports define the boundary of the FPGA code. Any blocks outside that boundary are used for testing in software only. The input data is sampled at the same frequency as the operating frequency of the design. However, the KLD estimation module requires an input sample rate that is one-eighth of the system clock frequency. To accommodate this requirement, the input is first downsampled by a factor of four and then again by a factor of two. This two-stage downsampling of the input is performed to ensure accuracy of the samples used by the algorithm. Given a set of eight identical samples $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$, the first downsampling block is configured to capture the last sample from each subset of four samples – producing $\{x_4, x_8\}$ at the output. The second downsampling block is configured to capture the first sample from each subset of two samples at its input – producing $\{x_4\}$. Together, the two blocks provide the desired factor eight downsampling of the input and help maintain accuracy by capturing the sample from the middle of the input data set.

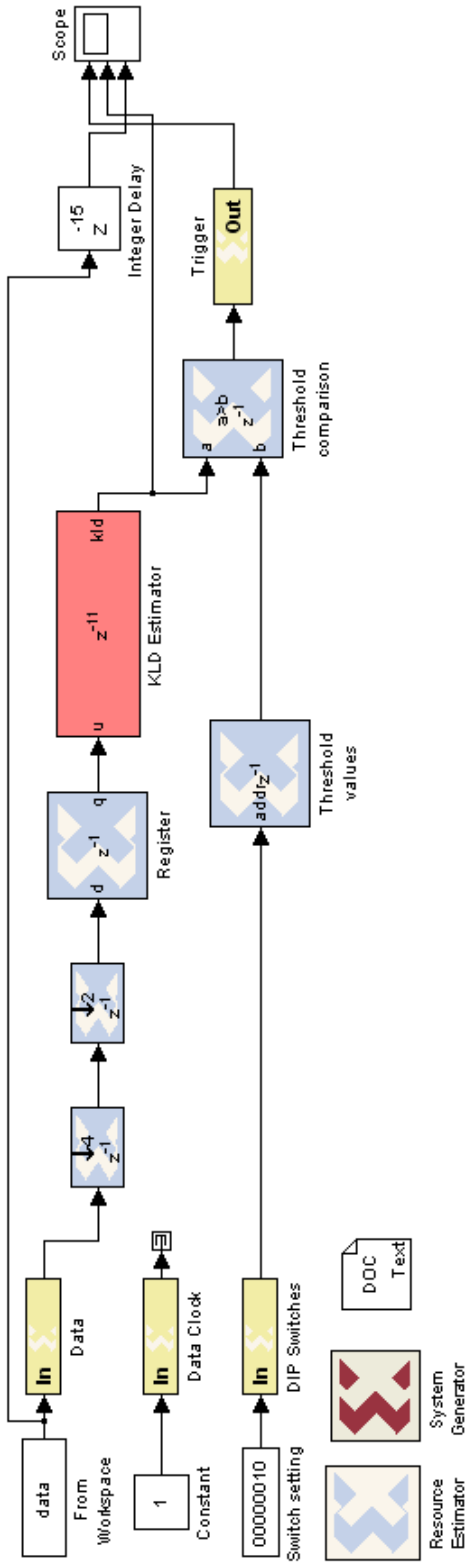


Figure 5.6: *Top level structure of the FPDKLD based implementation of the anomaly detection algorithm. The FPGA code boundary is defined by the* **In** *and* **Out** *gateway ports. The KLD estimator module runs at eight times the speed of the input sample rate. A set of eight switches on the target board functions as a runtime adjustable 8-bit KLD threshold.*

The downsampled input is processed by the KLD estimator module which represents the core of the algorithm. The estimated KLD is compared against a runtime configurable threshold to produce a trigger signal. The configurable threshold is obtained by combining a set of eight switches on the FPGA host board and a single on-chip ROM module. The set of switches is used as a single 8-bit address to retrieve one of the two hundred and fifty-six (2^8) predefined KLD thresholds stored in the ROM. Additional parameters such as signal period, data window size, input word size and histogram bin count can be assigned to the KLD estimator module at compile-time.

KLD estimator module

The internal structure of the KLD estimator module is shown in Figure 5.7. It implements the FPDKLD variant of the anomaly detection algorithm described previously in Section 5.2.4. After removing the sign of the input sample, it is passed through the bin sorting logic. The purpose of the this module is to evaluates the bin index associated with the provided input sample. Figure 5.8 illustrates the design.

The module computes the bin index, idx, from the input sample value, u, using the relationship

$$\text{idx} = \left\lfloor \frac{u}{\Delta B} \right\rfloor - \left\lfloor \frac{u_{\min}}{\Delta B} \right\rfloor \tag{5.4}$$

where $u_{\min}$ is the lower bound of the input values and ΔB is the bin width. For inputs in 2s complement format with zero fractional bits and a fixed word size, w, the lower bound can be defined as $u_{\min} = -2^{w-1}$ and the bin width is $\Delta B = \dfrac{2^{w-1} - u_{\min}}{\beta}$. For a simpler and more efficient implementation, the number of histogram bins used is restricted to be a PoT. With this constraint, it is clear that the value of ΔB must also be a PoT – allowing (5.4) to be evaluated without requiring any explicit division operation. As a result, the bin index is obtained effectively and efficiently from the input by simply bit-shifting it by an appropriate number of steps.

The bin index computed from the input is then passed to the PMF/histogram estimation module. It encapsulates the buffers representing P_n and Q_n and produces the set of probability values needed to compute the DKLD. For simplicity, the module produces time multiplexed signals on each of the six outputs. $p_{n-1}(\{j,k,m,o\})$ is obtained on the output labelled `P[n-1]`, $p_n(\{j,k,m,o\})$ is obtained on the output labelled `P[n]` and so on. In addition to

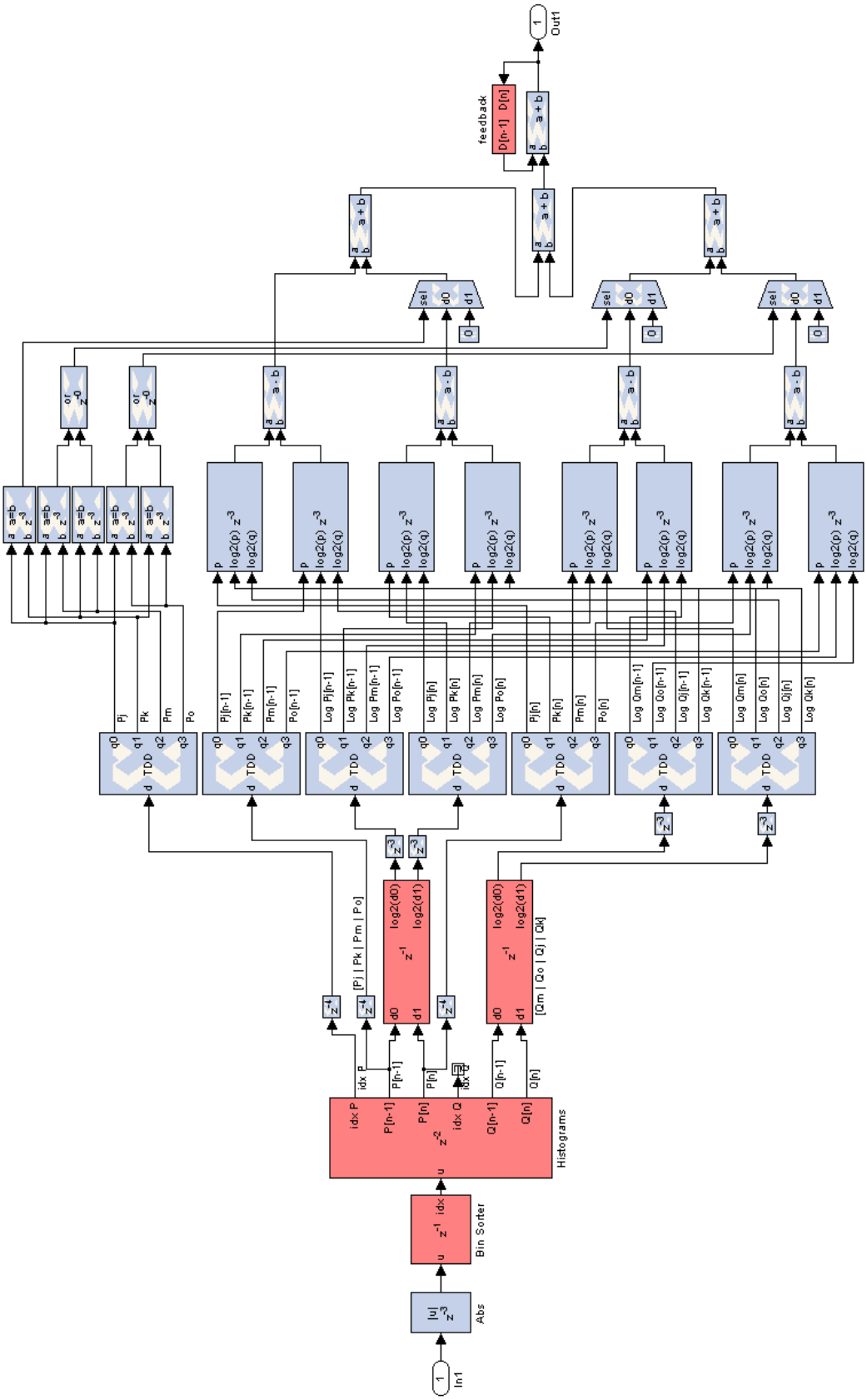


Figure 5.7: *KLD estimator using the FPDKLD method. The PMF/histogram estimator generates multiplexed results that must be demultiplexed before use. Two DPRAM based $\log_2$ lookup tables and logic for computing the eight terms of the DKLD equation are visible.*

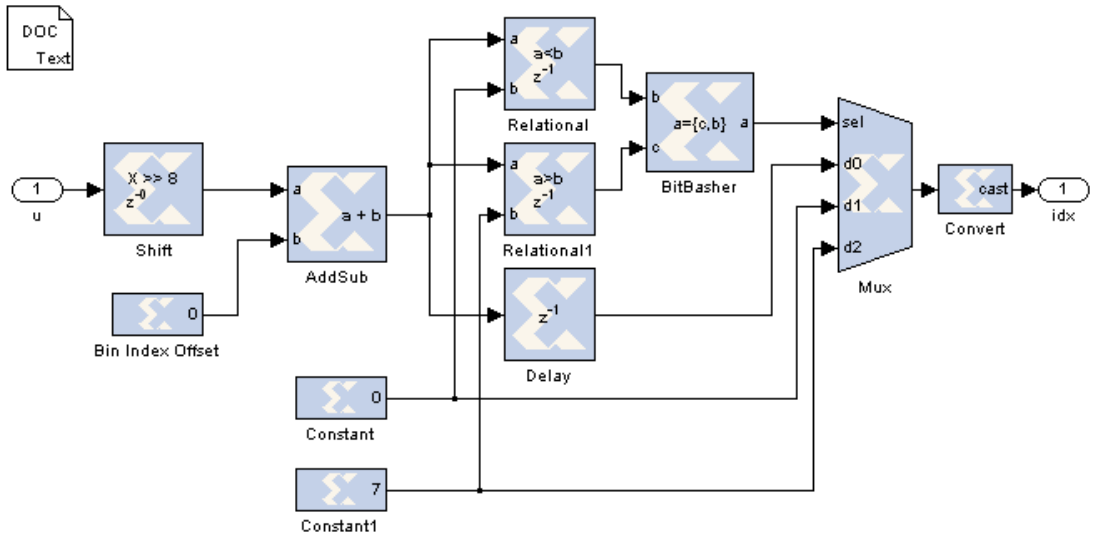


Figure 5.8: *Logic to compute the bin index associated with each input sample using* (5.4)*. The bin width is required to be a PoT. The design, as illustrated, is configured for eight histogram bins. The bin index offset is zero since* $u_{\min} = 0$*.*

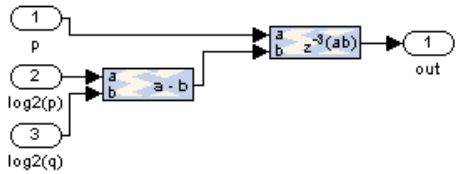


Figure 5.9: *Logic for computing each of the eight terms in DKLD.*

the probability values, the four bin indices $\{j,k,m,o\}$ are also obtained on the outputs labelled `idx P` and `idx Q`.

A set of two DPRAM based lookup tables are used to evaluate the sixteen base-2 logarithms[†] before the multiplexed data frames are demultiplexed to yield the individual components necessary to compute the DKLD using (5.3). The logic employed for calculating each of the eight terms of the equation is shown in Figure 5.9. Simple control logic consisting of a number of comparators and single-bit multiplexers prevent duplicate terms from being used in the final step of the DKLD calculation.

Due to effects that can be attributed to starting with non-full buffers, it is necessary to ignore the KLD values obtained until the system reaches a steady state. The precise number of values that need to be ignored is slightly larger than the combined depth of the buffers used in the

[†]Four simultaneous lookups on time multiplexed data sets containing four probability values each.

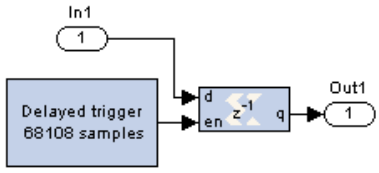


Figure 5.10: *Feedback logic utilised to supply D_{n-1} to the DKLD equation. It includes a configurable activation delay.*

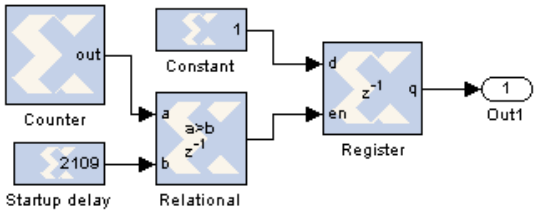


Figure 5.11: *Logic for a trigger signal with configurable delay. The initial value stored in the register is zero. The constant value compared against the counter is the required delay minus three. This accounts for the additional delays introduced by the comparator and register.*

design . This "initialisation period" allows the system to populate the PMFs buffers with signal data for meaningful DKLD computations. While the system is initialising the buffers, it is necessary to prevent the output from entering the feedback path utilised by DKLD. This is achieved through the use of a feedback block that incorporates a configurable activation delay. An implementation of the block is shown in Figure 5.10.

The logic used to obtain a delayed activation/trigger signal is shown in Figure 5.11. The desired functionality is obtained by using a register with an exposed enable port. The initial value stored in the register is zero and the enable port is driven by a comparison function that is initially false. The comparison performed is between a constant reflecting the desired delay minus three and a counter capable of counting to a number greater than the desired delay. The subtraction accounts for additional delays introduced by the comparator and the register. After an appropriate number of clock cycles, the counter reaches a suitable value and the comparator drives the enable pin of the register high – causing it to sample the constant value of one at the input and subsequently drive it to the output. The new output is maintained even when the enable signal to the register is eventually removed.

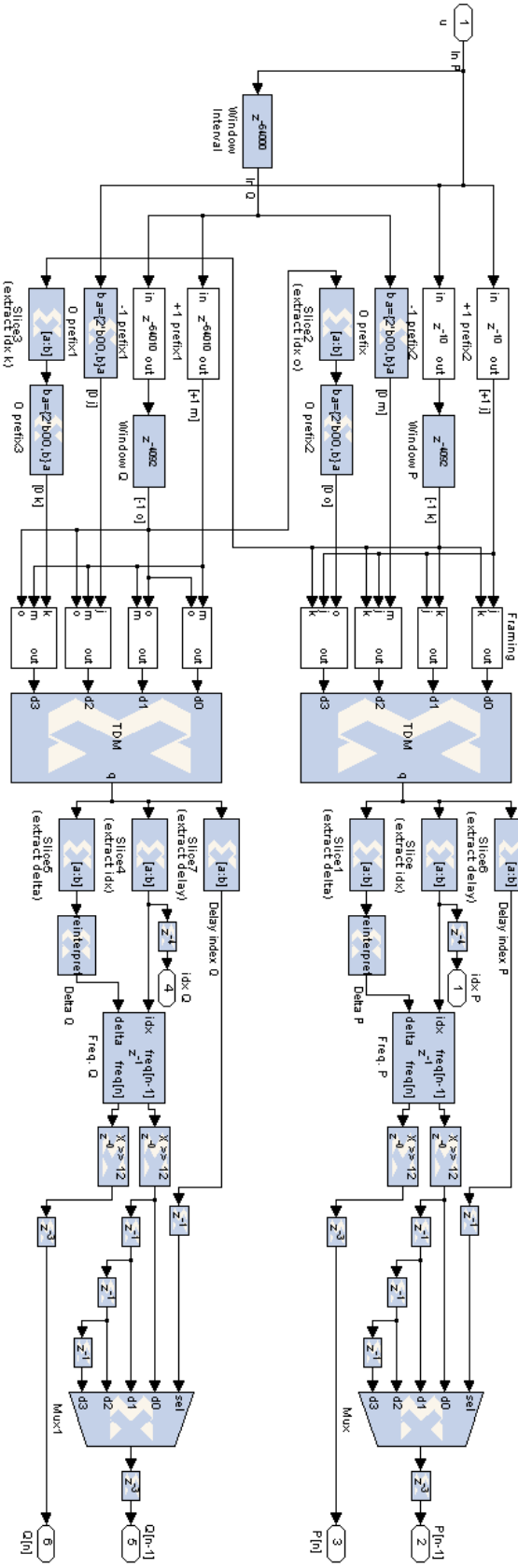


Figure 5.12: *Logic to compute the probability values required by DKLD. It also includes a buffer for storing the bin indices of the inter-window samples and two smaller buffers for implementing each of the windows. Each of the four bin indices are framed with a 4-bit prefix to control the final output logic and indicate the requested change in the associated histogram frequency. The four frames from each window are multiplexed for serial processing by the associated histogram modules.Since the window size is a PoT, bit shifting the multiplexed output from the histogram module produces the necessary probability values.*

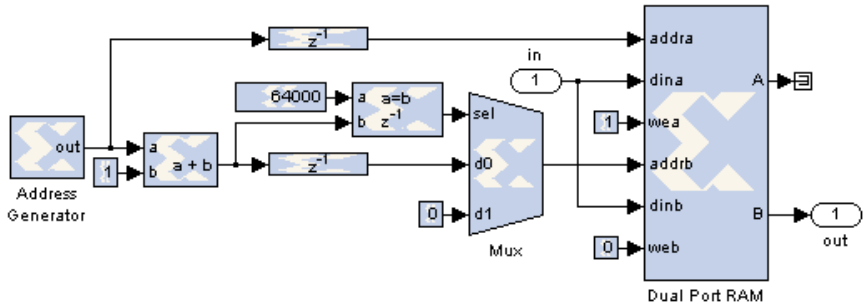


Figure 5.13: *Efficient circular buffer design based around a DPRAM device to emulate the behaviour of simple delay chains. A wrap-around counter continuously generates sequential addressed within a range corresponding to the depth of the buffer (64000 in this illustration). The read and write addresses are always a single bit apart, making the buffer circular and read/write collisions impossible. Port A of the DPRAM is used for writing while port B is used exclusively for reading.*

PMF/histogram estimator module

Details of the module used to estimate the histograms and hence the PMF is shown in Figure 5.12. It incorporates the sample buffers necessary to hold the intermediate values between the two data windows and the windows themselves. The buffers store bin indices rather than the real sample values to minimise the storage requirements. This exploits the fact that bin indices generally require much fewer bits to represent than the original input samples. Since the proposed algorithm utilises sliding windows, it is necessary to use some kind of a circular buffer that automatically overwrites older samples with new ones. While it is possible to use shift registers as simple delay chains to implement the buffers, such an approach is only suitable for cases where a relatively small number of values need to be stored. In order to implement the large buffers needed by the two windows and the inter-window samples, an alternative solution must be used.

Once again a DPRAM device is used to develop an efficient solution. The design is shown in Figure 5.13. A fixed word-size counter (returns to zero on overflow or configured limit) is used to continuously generate a sequence of addresses within a range of values corresponding to the depth required of the buffer. The address generated at any given time is used to write the input sample into the DPRAM device. The read address is obtained by adding one to the write address. Since the two addresses are always a single bit apart, the buffer is circular and read/write collisions are impossible. In the illustration, the buffer depth is configured as 64000 and port A is used for writing while port B is used for reading. The DPRAM based

circular buffer emulates the behaviour of simple delay chains. It is however more efficient for storing a large number of samples since on-chip memory blocks are utilised rather than slice flip-flops which are limited in number and necessary for implementing other logical functions of the algorithm.

The input to the PMF estimation module is the bin index j. In addition to being used for histogram and PMF estimation, it is stored in the inter-window sample buffer and the buffer corresponding to the window P_n. Bin index m is obtained from the output of the inter-window sample buffer and subsequently stored in the buffer corresponding to window Q_n. Indices k and o are obtained from the outputs of the window buffers P_n and Q_n respectively. A set of two 2-bit prefixes are added to each of the four bin indices to indicate the required change in the histogram frequency and control the module's output logic. The structure of a frame consisting of a set of prefix bits and the index is shown in Figure 5.14.

As stated previously in Section 5.2.2, "Delta" encodes the amount (+1,-1, or 0 in 2s complement format) by which the frequency count associated with the bin index should be changed. The "Output control" bit are used to operate the multiplexer at the final stage of the module's output for probability values at time $n-1$. It ensures that the correct values are obtained in the event that the same bin index appears more than once in the quartet.

Due to the effects of starting with non-full buffers, a number of samples corresponding to each relevant buffer depth must be ignored before appropriate "Delta" values can be appended to the bin indices. As a result the prefixing must be done using a block that supports a configurable activation delay. The design of such a module is shown in Figure 5.15. The prefix used (+1 or -1) and the delay can be set using the block's configuration interface. In reality, the delay required is slightly larger than the buffer depth to account for the latency from preceding blocks in the design. The delayed trigger block used is similar in design to that illustrated in Figure 5.11.

Since a given histogram module can only be addressed by a single bin index at a given time, it is necessary to employ factor of four RS on the histogram modules to obtain the eight probabilities needed by DKLD from each of the PMFs. This results in a quadrupling of the

Figure 5.14: *Bin index frame with control bits.*

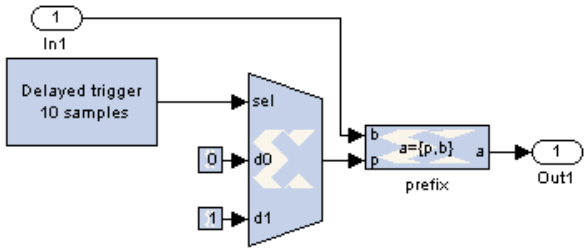


Figure 5.15: *Logic for attaching a configurable prefix (+1/-1) to the input. The module also includes a configurable activation delay.*

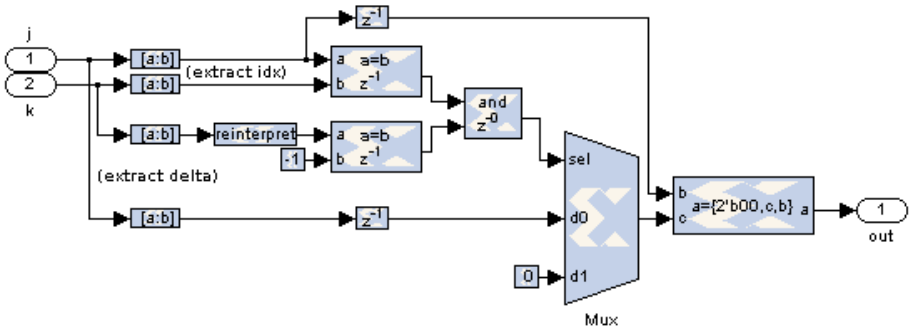


Figure 5.16: *Logic to handle the conditions $j = k$ and $j \neq k$. When the indices are equal, "Delta" is set to zero. The output delay is zero regardless of the condition.*

sample rate. Since the DPRAM in the histogram module requires a further doubling of the sample rate, there is a factor of eight difference in speed between the original input to the KLD estimator module and the input to the DPRAM device in the histogram module. RS leads to a serially multiplexed version of the four bin indices and this can be confusing for the histogram module. It has been stated previously that care must be taken to ensure that the frequency counts obtained for time $n-1$ are accurate in the event that the four bin indices j, k, m and o are not unique. The problem arises due to the fact that at any given time four operations are performed on a given histogram. Since the histogram itself does not have any concept of time this leads to a confusion regarding the frequency count at time $n-1$ when then bin index to be accessed refers to a location that has been previously modified within the same time frame. As a result, it is necessary to utilise some additional logic that monitors the quartet of bin indices and supplies the probability value obtained from the first instance of occurrence of duplicated indices. This is achieved by controlling the output delay using a multiplexer controlled by the "Output control" bits of the frame (Figure 5.14).

The conditions monitored (with respect to window P_n, the process is identical for window

Condition	Output delay		Delta	
	Value	Prefix bits	Value	Prefix bits
$j = k$	0	00	0	00
$j \neq k$	0	00	+1	01
$k = j$	1	01	0	00
$k \neq j$	0	00	−1	11
$m = j$ AND $m \neq k$	2	10	0	00
$m = k$ AND $m \neq j$	1	01	0	00
$m = j$ AND $m = k$	2	10	0	00
$m \neq j$ AND $m \neq k$	0	00	0	00
$o = j$ AND $o \neq k$	3	11	0	00
$o = k$ AND $o \neq j$	2	10	0	00
$o = j$ AND $o = k$	3	11	0	00
$o \neq j$ AND $o \neq k$	0	00	0	00

Table 5.3: *Dealing with duplicates in a bin index quartet. The conditions described are with respect to window P_n. The table is similar for Q_n.*

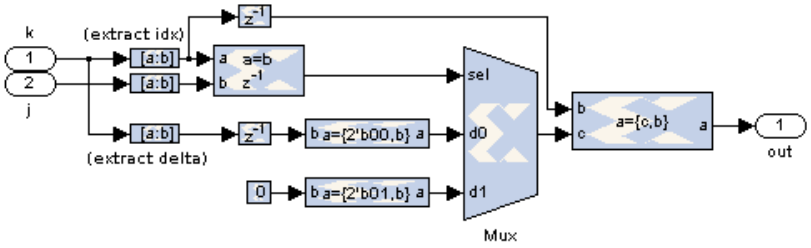


Figure 5.17: *Logic to handle the conditions $k = j$ and $k \neq j$. When the indices are equal, "Delta" is set to zero. The output delay depends on the condition.*

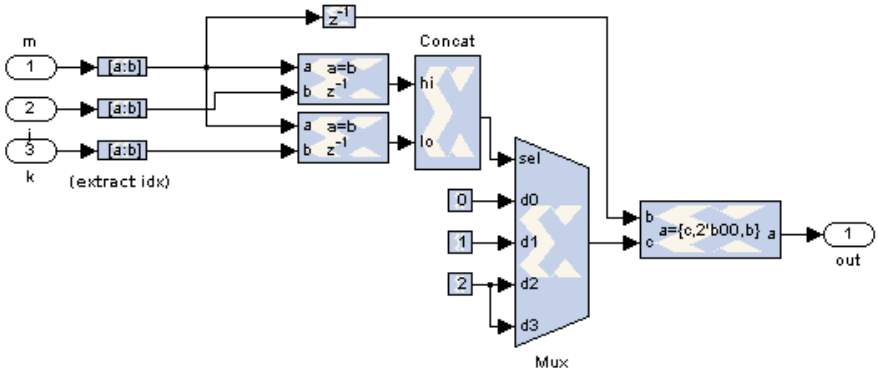


Figure 5.18: *Logic to handle the conditions ($m = j$ AND $m \neq k$), ($m = k$ AND $m \neq j$), ($m = j$ AND $m = k$) and ($m \neq j$ AND $m \neq k$). The output delay assigned depends on the condition.*

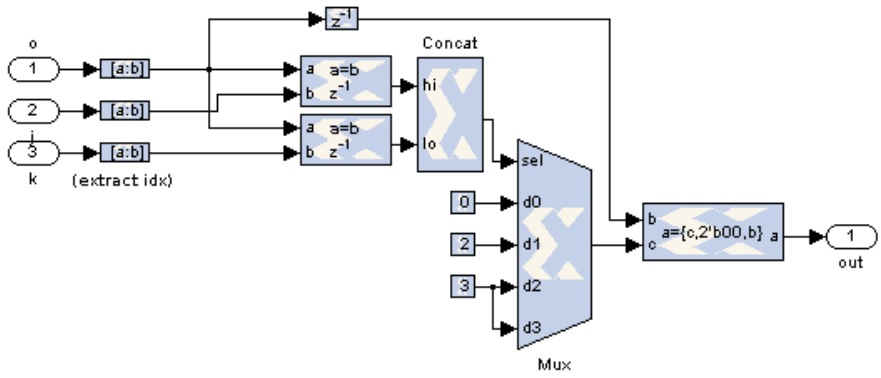


Figure 5.19: *Logic to handle the conditions ($o = j$ AND $o \neq k$), ($o = k$ AND $o \neq j$), ($o = j$ AND $o = k$) and ($o \neq j$ AND $o \neq k$). The output delay assigned depends on the condition.*

Q_n) are summarised in Table 5.3. The logic blocks used to handle these special cases (with respect to window P_n) are shown in Figure 5.16, Figure 5.17, Figure 5.18 and Figure 5.19. In addition to inserting the 2-bit prefix into the frame indicating the required output delay, the blocks performing comparisons between indices j and k (Figure 5.16 and Figure 5.17) can also reset the "Delta" bits to zero when $j = k$ since no changes need to be made to the frequency counts under such circumstances.

The DPRAM based histogram module implemented is shown in Figure 5.20. The design is identical to that described previously in Section 5.2.2 and illustrated in Figure 5.3 with the exception of the clock enable (CE) probe and sample rate changing blocks. Bin pre-loading is achieved by setting the initial value of each bin to the desired pre-load value. The upsampling block doubles the input sample rate in order to support the two clock cycles per sample needed by the histogram module. The stored frequency count is read on the first cycle while the modified count is stored back into memory on the second cycle. The downsampling blocks placed just prior to the module's outputs return the sample rate to its original value.

In contrast to the other blocks used in the design so far, the CE probe is not something that can be directly translated into hardware. It is a virtual function block that simply extracts the CE signal associated with a given signal. In a multi-rate design such as this, some modules require a different clock rate than others. To implement this in hardware the whole system is driven using the fastest clock signal required by the design and slower clock signals for modules that require it are then derived from the main clock by means of CE signals. Since the input sample rate for a given module is related to its CE rate, information regarding the latter can be extracted from the input signal using a construct such as the CE probe.

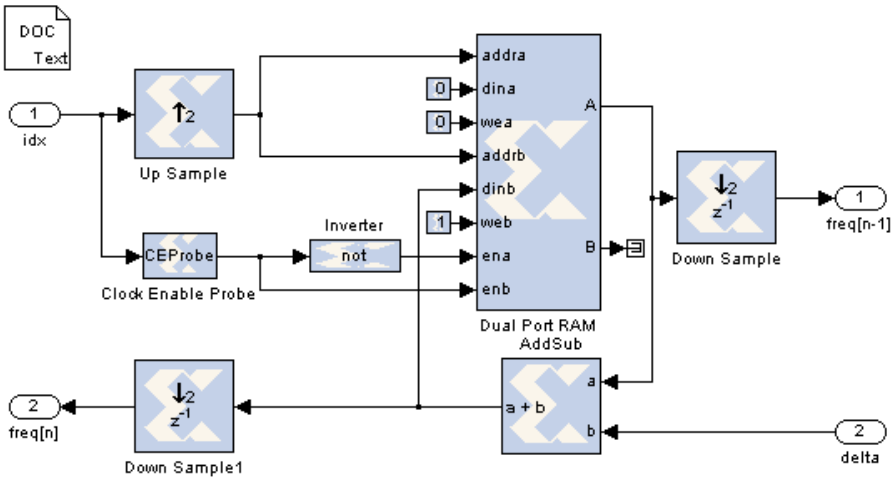


Figure 5.20: *Implementation of the DPRAM based histogram described previously in Section 5.2.2. The upsampling block at the input is necessary to double the effective clock rate. A CE probe attached to the original input provides the* 50% *duty cycle pulse needed by the port enable pins of the DPRAM. Downsampling blocks at the outputs return the sample rate to its original value.*

Since the DPRAM device used in the histogram module is the fastest component in the design, the CE signal associated with it (and its input signals) has a duty cycle of 100%. However, the input to the device is derived from a upsampling block that doubles the sample rate. This implies that the CE signal associated with the original inputs to the histogram module has a duty cycle of 50%. Therefore, extracting that CE signal using a CE probe provides the alternating enable signal that is needed to operate the port enable pins of the DPRAM device.

Targeting hardware

Once the implementation is completed, it must be converted into a bit-stream file that can be used to program the target FPGA. The first step of that process is to translate the design into hardware description language (HDL). This task is performed via the System Generator token placed in the top-level design. The configuration window of the token (Figure 5.21) allows a number of important options such as the target hardware part, desired clock period and design export format to be specified. After the necessary HDL files are generated, the design can be synthesised, translated, mapped, placed and finally converted into a bit-stream using a suitable tool-chain such as that provided by the Xilinx ISE environment.

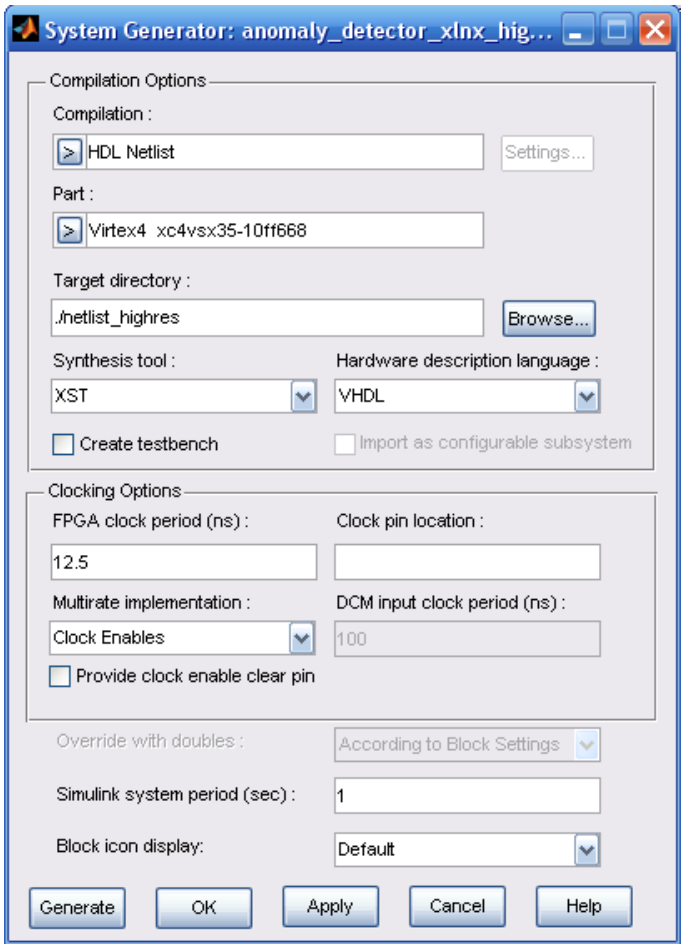


Figure 5.21: *System Generator token configuration options for generating HDL files from the implemented design.*

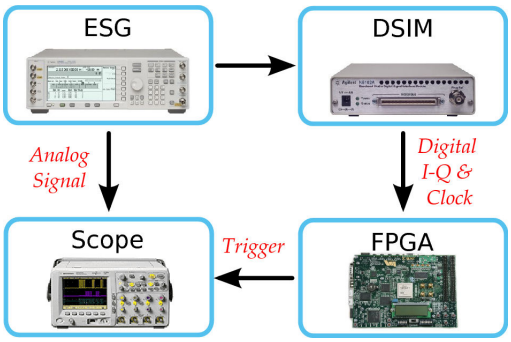


Figure 5.22: *Block diagram of the hardware test-bed. The E4438C ESG signal generator produces the signal under test in both analogue and digital format. The digital signal is passed to the FPGA platform via the N5102A digital signal interface module while the analogue signal is fed into the oscilloscope. The trigger signal from the FPGA core is also connected to the oscilloscope via a digital probe for comparison.*

5.4 Experimental results

A pair of wireless broadband (WiBro) signals known to contain a number of anomalous data segments are used to test the hardware based implementation of the anomaly detection algorithm. The hardware test chain used for verifying and testing the algorithm is described in Section 5.4.1 while results obtained from the platform are presented in Section 5.4.2.

5.4.1 Hardware test chain

In order to perform live tests of the anomaly detection algorithm a Xilinx ML402 platform containing a Virtex-4 SX35 FPGA is used as the implementation target. The hardware test chain utilised for verification purposes is shown in Figure 5.22. The Agilent E4438C ESG signal generator simultaneously provides analogue and digital versions of the in-phase and quadrature components of the signal under test. The digital data stream is connected to the FPGA platform via the Agilent N5102A digital signal interface module (DSIM) while the analogue signal is connected to a oscilloscope for display. The DSIM conditions the signal data (word size, bit alignment, clock relationship settings) and provides a synchronous clock signal that is used to drive the FPGA. The trigger output from the FPGA platform is also connected to the oscilloscope via a digital probe so that it can be directly compared against the signal under test. The trigger source for the oscilloscope is external and is also obtained from the signal generator. The physical set-up is shown in Figure 5.23.

It has been stated in Section 5.3 that the fastest components in the design are the two DPRAM

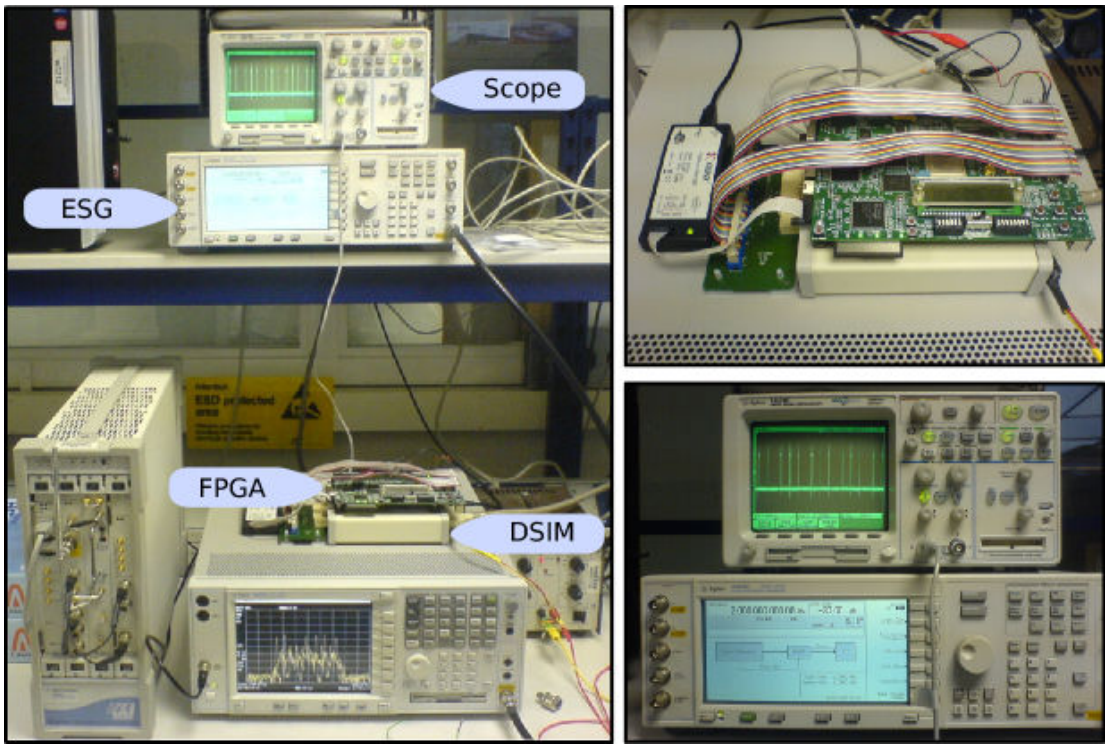


Figure 5.23: *Physical set-up of the hardware test chain.*

devices used in the histogram modules. The difference in speed between each of these components and base modules such as the bin sorting logic is eight fold due to RS and design of the histogram. This implies that for each data sample presented to the anomaly detection algorithm, eight clock cycles are required. Unfortunately the DSIM is only able to provide a maximum of four clock cycles per sample (CCPS) [106]. Since the implemented design requires eight CCPS, it is not possible to directly connect the DSIM clock output to the top level clock input of the design. The solution is to produce a wrapper for the design that accepts the DSIM clock and multiplies it by two before connecting it to the core implementation.

An illustration of the logic required to implement the clock multiplier circuit is shown in Figure 5.24. A digital clock manager (DCM) module is at the heart of the design. It is capable of digitally synthesising a range of clock frequencies with various phase characteristics from the input reference clock. There are eight of these modules available for use on the Virtex 4 SX35 device targeted by the design [107]. One of the basic functions of a DCM device is to double the frequency of the input reference clock signal. A global clock input buffer (IBUFG) connects the reference clock signal from the DSIM to the `CLK` input pin of the

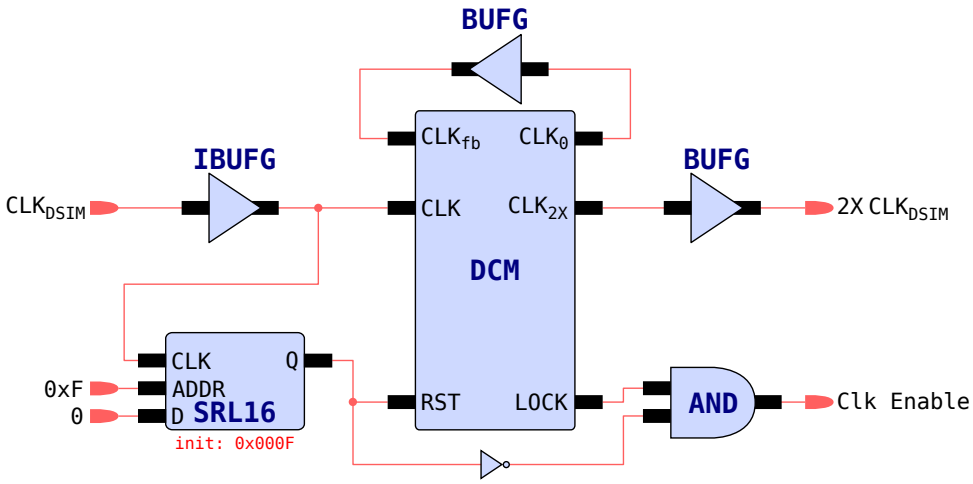


Figure 5.24: *Clock multiplier circuit using a DCM.*

DCM. The signal from the $\texttt{CLK}_0$ output (identical to CLK with zero phase difference) is connected to the $\texttt{CLK}_{fb}$ via a clock buffer (BUFG). This feedback of the clock signal is necessary to maintain relative phase relationships between the input and the outputs and eliminate clock skew [108]. The reset pin (RST) of the DCM is connected to the output of a 16-bit shift register lookup table (SRL16) initialised with the value 0x000F and clocked using the buffered reference clock signal. The address pin of the SRL16 is fixed at 0xF (binary value: 1111) – meaning that the output always produces the MSB of the stored 16-bit word. The data input pin is tied to zero. The net result of this arrangement is that twelve cycles after the input reference clock signal $\texttt{CLK}_{DSIM}$ is applied, the SRL16 produces a reset signal for the DCM that lasts for four cycles. This is done to ensure that the DCM only locks onto a stable input clock signal. Once the DCM finishes configuring itself and the clock output is stable, it drives the LOCK output high. This signal is combined with the inverse of the reset signal from the SRL16 to produce the clock enable signal for the core design. The doubled clock signal, $\texttt{2X CLK}_{DSIM}$, is obtained via another BUFG from the $\texttt{CLK}_{2X}$ output of the DCM. The phase difference between $\texttt{2X CLK}_{DSIM}$ and $\texttt{CLK}_{DSIM}$ is zero. Appendix B provides a HDL implementation of the clock multiplying logic and the accompanying constraint file.

With the clock multiplier in place, the design is ready for integration into the hardware test chain. The system is configured to operate at a frequency of 80 MHz – allowing for an input sample frequency of 10 MHz. As a result the sample clock of the signal generator is set to 10 MHz. The DSIM configuration interface is used to set the output word size to 12-bits in 2s complement format (MSB aligned) and the clock is set to 40 MHz with four CCPS. The

logic type is set to 2.5 V complementary metal oxide semiconductor (CMOS) to match the default I/O configuration of the FPGA host board. It is important to ensure that the FPGA is powered and programmed before turning on the DSIM module. This is done to avoid interference from the DSIM clock during the FPGA configuration phase.

5.4.2 Results

A pair of WiBro signals known to contain a number of different anomalous data segments are used to test the hardware platform. The design is configured with $|P_n| = 4096$ (320 μs), $\beta = 8$ and $\psi = 64000$ in order to match the signal period of 5 ms. Although both in-phase and quadrature channels of the signal are supplied by the DSIM, only the former is analysed by the hardware. This minimises the complexity of the implementation with the assumption that anomalies in the signal affect both channels in a similar fashion.

5.4.2.1 Test signal I

Figure 5.25 shows the result of analysing the first WiBro signal using the FPDKLD implementation of the anomaly detection algorithm. The image displayed is a trace of the signal under test and the associated trigger events, obtained directly from the oscilloscope. The WiBro test vector is identical to that shown previously in Figure 4.7 and analysed in Section 4.3.1.2. It can be seen that the signal period observed is 6.4 ms rather than the 5 ms that is expected for a WiBro signal. This is due to a difference between the original sampling rate of the recording (12.8 MHz) and the sampling rate used to play it back (10 MHz) using the signal generator.

It is also clear from the trace that the uplink (UL) frame located at the 20.7 ms mark is longer than the others and hence is anomalous in this context. With $D_{\text{th}} = 0.0313$, the FPGA implementation of the algorithm clearly succeeds in detecting this signal anomaly. The first trigger event obtained (A) coincides exactly with the anomalous segment of the unusual UL frame. A second trigger event (B) is observed when the UL frame structure subsequently returns to normal and the anomalous segment is no longer present. This behaviour is due to the twin-windowing nature of the algorithm. The width of each trigger pulse is approximately equal to the temporal duration of the anomaly it detects.

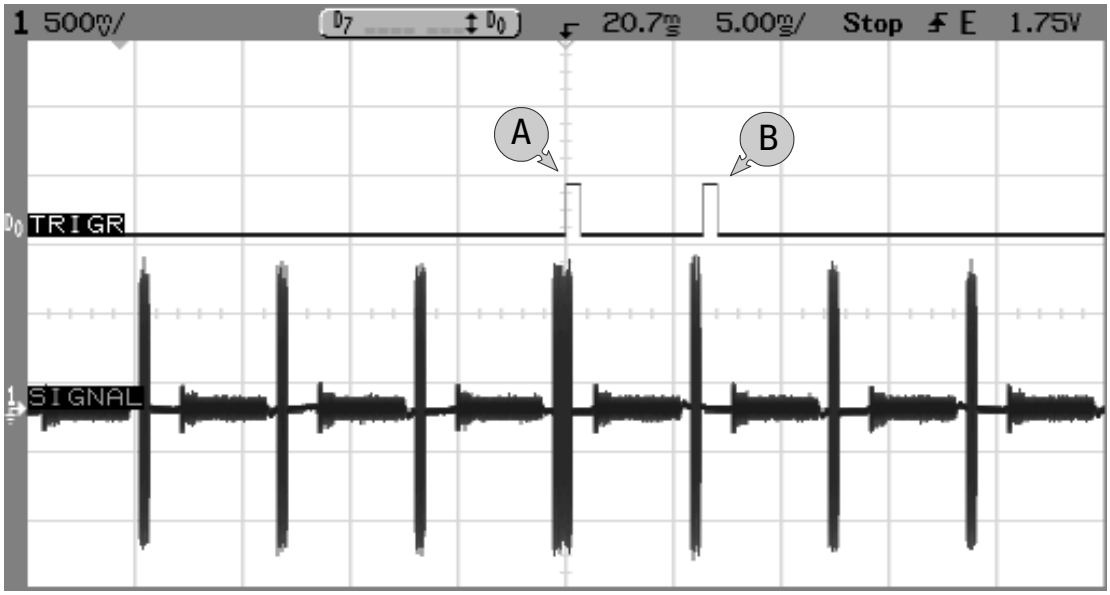


Figure 5.25: *Oscilloscope trace of a WiBro signal with a single anomalous frame and the associated trigger events obtained from the hardware platform. The KLD threshold is* 0.0313. *Trigger event A marks the start of the unusual segment of the anomalous frame. A second trigger event, B, is also obtained in the subsequent frame due to the disappearance of the anomalous feature.*

5.4.2.2 Test signal II

The second WiBro test signal is shown in Figure 5.26. It is identical to the signal shown previously in Figure 4.8 and analysed in Section 4.3.1.3 with the exception of an additional change in the timing structure. Once again the observed signal period is 6.4 ms rather than 5 ms due to a mismatch between the recording and playback sample frequencies. Analysing the signal with $D_{\text{th}} = 0.0625$ is seen to produce five trigger events – corresponding to the three anomalous conditions known to be present in the signal.

Trigger events A and B are due to a momentary disruption in the natural frame period of the signal that occurs close to the 1 ms mark. The first trigger event specifies the position where the UL frame should have been but is not. Since the entire frame appears to be missing, the duration of the trigger signal is approximately equal to the duration of the preceding UL frame. The second trigger event then detects the "missing" UL frame at a position where it should not be under nominal operating conditions.

The next set of trigger events – C and D – are caused by an ephemeral command sequence contained at the start of the UL frame seen at the 17.2 ms mark. The command causes the power control mechanism of the protocol to be initiated, resulting in the anomalous

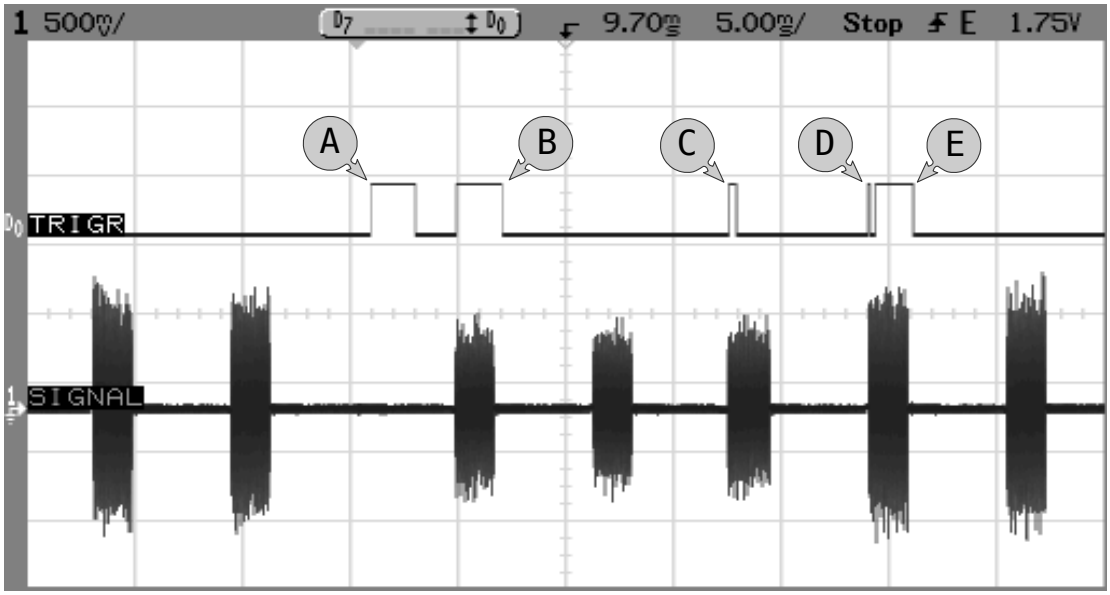


Figure 5.26: *Oscilloscope trace of a WiBro signal with multiple anomalous events and the associated trigger events obtained from the hardware platform. The KLD threshold is* 0.0625*. Trigger events A and B are caused by a momentary disruption in the periodic nature of the signal. Trigger event C flags the presence of a very brief command sequence that leads to a signal power level change. Trigger event D marks the position in the subsequent frame where the power change inducing command was previously present and finally, trigger event E corresponds to a sudden change in the overall signal power level.*

behaviour which is detected by trigger event E. Although it is indistinguishable from the rest of the frame in the signal trace, the algorithm succeeds in locating the anomalous command sequence – as clearly demonstrated by trigger event C. Absence of the power control inducing command from the subsequent UL frame at 23.7 ms is marked by trigger event D.

Once initiated, the power control mechanism causes a sudden increase in the transmission power that is clearly visible as a change in the voltage level of the UL frame at 23.7 ms. This behaviour can be considered to be anomalous and is immediately flagged by trigger event E. Since the power remains unchanged for the subsequent UL frame at 30.7 ms, no further trigger events are obtained.

5.5 Chapter summary

A highly efficient and effective implementation of the KLD based anomaly detection algorithm for periodic signals is presented in this chapter. The implementation described is for FPGA oriented signal processing platforms. Running at a speed of 80 MHz on a Xilinx Virtex-4 SX35 device, the algorithm is capable of processing data with an input sample rate of 10 MHz in real-time. Tests performed successfully on wireless signals captured over the air interface show that the implementation is indeed capable to detecting anomalies in test vectors that may be encountered under real-world operating conditions.

A difference equation form of the KLD equation (DKLD) is also presented. It is applicable to problems where the data sets to be compared using divergence analysis change little between successive time instances. The advantage of DKLD is its fixed computational complexity – meaning that the KLD can be computed efficiently regardless of the input data size or histogram resolution utilised. When targeting a specific class of hardware devices such as FPGAs, architecture specific optimisations can be used to further improve the efficiency of the implementation. This is the FPDKLD version of the anomaly detection algorithm where it is possible to obtain the instantaneous KLD value with just twenty-eight addition and multiplication operations.

Major architecture specific optimisations and methods employed in the FPDKLD implementation are: DPRAM based histograms/PMFs, base-2 logarithm lookup tables and circular buffers to emulate the function of delay chains. The concept of resource sharing is also employed to minimise hardware resource utilisation at the cost of a higher overall system clock speed.

Chapter 6
Conclusions

6.1 Summary of contributions

FOUR novel algorithms for speed estimation and signal anomaly detection are the primary contributions of the work presented in this thesis. Although methods for both speed estimation and anomaly detection exist in the literature, none offer the mix of characteristics that are necessary for applications hosted on a mobile hand-held platform. Some of the key weaknesses identified in the existing techniques are high computational complexity, low degree of flexibility and the need for substantial *a priori* information.

Majority of the speed estimation algorithms described in the literature utilise either the autocovariance/autocorrelation function, or the level crossing rate of the received fading signal envelope. A particular model of the underlying physical channel is therefore always assumed while developing the speed estimator. This is a weakness since the assumed model may not always match the behaviour of the real wireless channel. The algorithms presented in Chapter 3 are simple to implement and do not make any assumptions regarding the underlying wireless channel – with the exception that it exhibits frequency selective fading. The two-antenna approach to speed estimation is an adaptive algorithm that can provide accurate estimates over a wide range of speeds. At typical pedestrian velocities, the average error in the estimates is less than 2.67% when a terrestrial digital video broadcasting (DVB-T) transmitter is used as the wideband signal source. When accurate estimates of the channel dependent parameter are available, the spatial correlation function (SCF) of the channel can be used to obtain speed estimates that contain an error of less than 0.15% on average. When the extended, SCF based method is used for dead reckoning (DR) along with accurate heading information, the displacement error is less than 24.5 cm after 60 seconds for a typical pedestrian speed profile.

A review of the statistical and information theoretic methods of anomaly detection described in the literature reveals that they are generally computationally expensive, require substantial *a priori* knowledge or are simply unsuitable for use with complex, high-speed signals such as those from a broadband wireless system. In contrast, the algorithms described in Chapter 4

are simple, require zero *a priori* knowledge, make no assumptions regarding the underlying generative processes and are capable of processing high-bandwidth signals in real-time. The first of the two algorithms employ Kullback-Leibler divergence (KLD) to compare analogous segments of periodic signals by means of a pair of windows that are separated by the inherent signal period. It is not necessary to have *a priori* knowledge of the period since the algorithm is capable of automatically extracting this information from the signal. Tests performed on a set of real broadband wireless signals show that the proposed algorithm is extremely effective. Depending on the type of anomaly present, the difference between the anomalous event and the background noise can be as large as 60 dB. Satisfactory performance is maintained even when the test signal is severely under-sampled (20 dB difference even after under-sampling by a factor of 100). The performance of the algorithm is also robust with respect to the histogram resolution utilised for probability mass function (PMF) estimation.

The second anomaly detection algorithm is based on a measure of the information content of individual signal events and therefore can be also applied to signals that lack periodicity. The only pre-requisite of the algorithm is a training data set to initialise the reference event probabilities. Clustering is employed to reduce false positives – detection of an anomalous event is signalled only when a predefined number of contiguous events exceed the specified information threshold. Even with a small cluster length of three, performance is improved significantly. Tests show that a true positive rate of 100% can be achieved with a false discovery rate 1% – depending on the choice of the threshold and the histogram resolution.

An efficient architecture for implementing the KLD based method of anomaly detection in field-programmable gate array (FPGA) hardware is also given in Chapter 5. With the optimisations described, the computational complexity of the algorithm reduces to just 28 additions and multiplications. Live tests on real, 10 MHz wireless broadband signals performed through a Xilinx ML402 FPGA platform show that the algorithm is indeed effective in practice and capable of real-time operation.

The speed estimations and signal anomaly detection algorithms presented in this work are both simple to implement and effective – making them ideal candidates for the role of enabling technologies for next-generation mobile handset oriented services.

6.2 Limitations and scope for further research

Simulations of the speed estimation algorithms presented in Chapter 3 assume that the channel itself is static and therefore the channel parameters such as Doppler spread, angle of arrival (AoA), path delay and phase shift are time invariant. While this is a valid assumption for cases where scatterers in the channel move slowly relative to the transmitter and receiver, it may not hold for certain practical wireless channels. The second practical challenge facing the two-antenna speed estimation algorithm is the requirement that the antenna pair is always aligned parallel to the direction of motion. While this condition is easy to satisfy for systems mounted on a rigid platform such as a vehicle, it is much more difficult to enforce for portable devices – alignment errors will certainly translate to errors in the speed estimates. Additional work is therefore needed to determine the impact of imperfect antenna array alignment and movement in the channel on the performance of the speed estimation algorithms. Finally, it should be noted that due to its design, the algorithm cannot be used when a usable broadband signal source is absent or when the receiver is equipped with just a single antenna element. Use of antenna arrays containing more than two antenna elements for improving the speed estimates may also be investigated. In particular, an array consisting of two pairs of antenna elements perpendicular to each other may be used to obtain an estimate of the velocity vector (i.e. both heading and magnitude).

The primary limitation of the KLD based method of anomaly detection is that it is restricted to signals with periodic structure. The algorithm is generally robust with respect to parameter choices such as window size and histogram resolution but performance can suffer when extremely poor choices are made. It is also difficult for the algorithm to detect anomalies that do not lead to a significant change from the nominal probability distribution. Detection of anomalous events can be difficult under low signal to noise ratio (SNR) conditions; under such circumstances, the clustering approach introduced for the information content analysis (ICA) algorithm may be exploited: since anomalous events generally lead to a monotonically increasing divergence, detection can be signalled by the presence of monotonic sequences rather than an absolute divergence threshold.

Unlike the KLD based algorithm for anomaly detection, the information content analysis based method requires a clean training data set in order to initialise the reference event probabilities. Such a data set may not always be available in practice. The ICA algorithm is also significantly less robust with respect to the parameter choices compared to the algorithm

based on KLD. The information threshold in particular is seen to significantly affect detection performance. The ICA algorithm, in its current form, consists of a PMF with infinite memory that is updated using every signal event processed. As a result, anomalous events that have appeared in the past affect the outcome of tests performed on similar events in the future. Limitation of the PMF memory depth by means of a fixed-length window and its subsequent impact on the performance of algorithm are issues that provide avenues for further research in this area.

For an efficient implementation in FPGA hardware, the proposed architecture of the KLD based anomaly detection algorithm requires the window size to be a power of two. Although problems due to this restriction were not encountered during tests, it may limit the usability of the architecture with signals that require finer adjustments of the window size. The proposed architecture also requires a system clock that is eight times higher than the input sample rate. As a result, it may be challenging to use this architecture for processing inputs with very high sampling rates (e.g. above 100 million samples per second).

Appendix A
Relative error derivation and impact of acceleration

A.1 Derivation of the relative error in K

Given a correlation coefficient $\widehat{\rho}$ and the associated distance d, an estimate K' can be obtained using (3.24):

$$K' = -\frac{\ln(\widehat{\rho})}{23}\left(\frac{\lambda_c}{d}\right)^2 = C\frac{\ln(\widehat{\rho})}{d^2}, \tag{A.1}$$

where C $(= -\lambda_c^2/23)$ is a constant. Assuming that the real value of the coefficient is ρ, the true K is then:

$$K = -\frac{\ln(\rho)}{23}\left(\frac{\lambda_c}{d}\right)^2 = C\frac{\ln\left(\widehat{\rho} - e_\rho\right)}{d^2}, \tag{A.2}$$

where e_ρ is the error in the correlation coefficient. Therefore, the error in K' is given by:

$$K' - K = C\frac{\ln(\widehat{\rho})}{d^2} - C\frac{\ln\left(\widehat{\rho} - e_\rho\right)}{d^2} = C\frac{\ln\left(\frac{\widehat{\rho}}{\widehat{\rho} - e_\rho}\right)}{d^2}. \tag{A.3}$$

A simple manipulation of (3.22) results in

$$d^2 = \frac{\lambda_c^2 \ln(\rho)}{-23K} = C\frac{\ln\left(\widehat{\rho} - e_\rho\right)}{K} \tag{A.4}$$

which, when substituted into (A.3), yields an expression for the relative error in K:

$$e_K = \left|\frac{K' - K}{K}\right| = \left|\frac{\ln\left(\frac{\widehat{\rho}}{\widehat{\rho} - e_\rho}\right)}{\ln\left(\widehat{\rho} - e_\rho\right)}\right| = \left|\frac{\ln(\widehat{\rho})}{\ln\left(\widehat{\rho} - e_\rho\right)} - 1\right|. \tag{A.5}$$

A.2 Acceleration and the channel response

For a receiver with a single antenna, the spatiotemporal transfer function, $H(t,f,x)$, with time invariant multipath components can be written as

$$H(t,f,x)=\sum_{l=0}^{L-1}\alpha_l \exp\left\{2\pi f_{D_l} t + j\phi_l - j2\pi\tau_l f\right\} a\left(\theta_l\right). \tag{A.6}$$

Substituting the one-dimensional version of (3.2) into (A.6) and rewriting the Doppler spread as a function of the speed v and angle ω_l results in

$$H(t,f,x)=\sum_{l=0}^{L-1}\alpha_l \exp\left[j2\pi\frac{v}{\lambda_c}\cos\left(\omega_l\right) t + j\phi_l - j2\pi\tau_l f - j\mathscr{B}\cos\left(\theta_l\right) x\right]. \tag{A.7}$$

Since $v = v_0 + at$ and $x = x_0 + v_0 t + \frac{1}{2}at^2$, where v_0 is the initial speed, x_0 is the initial position and a is the acceleration,

$$H(t,f,x)=\sum_{l=0}^{L-1}\alpha_l \exp\left[j2\pi\frac{v_0+at}{\lambda_c}\cos\left(\omega_l\right) t + j\phi_l - j2\pi\tau_l f - \mathscr{B}\cos\left(\theta_l\right)\left(x_0 + v_0 t + \frac{1}{2}at^2\right)\right]. \tag{A.8}$$

It is therefore clear from (A.8) that when the acceleration is non-zero, the channel response is a function of quadratic rather than linear time t.

Appendix B
Implementation files

B.1 Clock wrapper

```
1 ----------------------------------------------------------------------------------
-- Company:         The University of Edinburgh
3 -- Engineer:        Mostafa Afgani
--
5 -- Create Date:    15:38:29 07/13/2009
-- Design Name:
7 -- Module Name:    kld_fullrate_top - structural
-- Project Name:    Smart triggers
9 -- Target Devices: Virtex 4 SX35 / ML402
-- Tool versions:   ISE 10.1
11 -- Description:   Top level wrapper for highres anomaly detector.
--                  Incorporates a DCM to obtain 2X clock
13 --
-- Dependencies:    ./netlist_highres/ files generated from SystemGenerator
15 --                  model
-- Revision:
17 -- Revision 0.03 - Input clock changed to 40MHz
-- Revision 0.02 - File Committed
19 -- Revision 0.01 - File Created
-- Additional Comments:
21 -- The DSIM clock must be set to 4X the ARB sample clock and the number of
-- clocks per sample must be set to 4. The DCM included in this design
23 -- takes care of multiplying the incoming clock to provide 8 clock cycles
-- per input data sample.
25 -- $Id:: kld_fullrate_top.vhd 1198 2010-03-25 21:25:33Z s0683019@EASE.ED.#$
----------------------------------------------------------------------------------
27 library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
29 use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
31
library UNISIM;
33 use UNISIM.VComponents.all;

35 entity kld_fullrate_top is
generic (
```

```
37    tClk : real := 25.0   -- Input clock period in ns (from DSIM)
);
39 Port (
    sysclk          : in  STD_LOGIC;                        -- Clk (40MHz)
41  dip_switches    : in  STD_LOGIC_VECTOR ( 7 downto 0); -- DIP switches
    data            : in  STD_LOGIC_VECTOR (11 downto 0); -- 12bit data
43  data_clock      : in  STD_LOGIC;                        -- Data clock
    trigger         : out STD_LOGIC                         -- KLD trigger
45 );
end kld_fullrate_top;
47

architecture structural of kld_fullrate_top is
49

    component anomaly_detector_xlnx_highres_cw
51  port (
       ce             : in  std_logic;
53     clk            : in  std_logic;   -- 80MHz
       dip_switches   : in  std_logic_vector( 7 downto 0);
55     data           : in  std_logic_vector(11 downto 0);
       data_clock     : in  std_logic;
57     trigger        : out std_logic
    );
59  end component;

61  constant LOGIC_HIGH : std_logic := '1';
    constant LOGIC_LOW  : std_logic := '0';
63

    signal sysclock, clk0, clkfb, clk2X, clk,
65         ce, dcm_rst, dcm_lck : std_logic;

67 begin

69  -- IBUFG for incoming clock (40MHz from DSIM)
    sysclk_IBUFG : IBUFG
71  generic map (
       IOSTANDARD => "DEFAULT"
73  )
    port map (
75     I => sysclk,
       O => sysclock
77  );

79  DCM_init_delay : SRL16
    generic map (
81     INIT => X"000F"
    )
83  port map (
       Q   => dcm_rst,
```

```
85      A0  => LOGIC_HIGH,
        A1  => LOGIC_HIGH,
87      A2  => LOGIC_HIGH,
        A3  => LOGIC_HIGH,
89      CLK => sysclock,
        D   => LOGIC_LOW
91  );

93  -- DCM to multiply incoming clock signal by 2
    DCM : DCM_BASE
95  generic map (
        -- Specify period of input clock in ns
97      CLKIN_PERIOD  => tClk,
        -- Specify clock feedback of NONE or 1X
99      CLK_FEEDBACK  => "1X",
        -- Specify performance mode
101     DCM_PERFORMANCE_MODE => "MAX_SPEED",
        -- Frequency mode of DLL: LOW/HIGH
103     DLL_FREQUENCY_MODE => "LOW",
        -- SOURCE_SYNCHRONOUS, SYSTEM_SYNCHRONOUS
105     DESKEW_ADJUST => "SOURCE_SYNCHRONOUS",
        -- Duty cycle correction, TRUE or FALSE
107     DUTY_CYCLE_CORRECTION => TRUE
    )
109 port map (
        CLK0   => clk0,     -- 0 degree DCM CLK ouptput
111     CLKFB  => clkfb,    -- DCM clock feedback
        CLK2X  => clk2X,    -- 2X CLK output
113     CLKIN  => sysclock,-- Clock input (from IBUFG, BUFG or DCM)
        LOCKED => dcm_lck, -- DCM is locked
115     RST    => dcm_rst  -- DCM asynchronous reset input
    );
117
    -- BUFG for DCM 1X feedback
119 DCM_FB_BUFG : BUFG
    port map (
121     I => clk0,
        O => clkfb
123 );

125 -- BUFG for the 2X clock signal output
    DCM_2X_BUFG : BUFG
127 port map (
        I => clk2X,
129     O => clk
    );
131
    -- Delay circuit init until DCM has locked
```

```
133     ce <= dcm_lck and (not dcm_rst);

135     anomaly_detector : anomaly_detector_xlnx_highres_cw
        port map (
137         ce           => ce,
            clk          => clk,
139         dip_switches => dip_switches,
            data         => data,
141         data_clock   => data_clock,
            trigger      => trigger
143     );

145 end structural;
```

B.2 Constraints

```
1 NET "sysclk" TNM_NET = sysclk;
  TIMESPEC TS_sysclk = PERIOD "sysclk" 25 ns HIGH 50%;
3
  # LOC constraints
5 NET "sysclk" LOC = "C13";
  NET "data<11>" LOC = "AC22";
7 NET "data<10>" LOC = "V22";
  NET "data<9>" LOC = "V21";
9 NET "data<8>" LOC = "W22";
  NET "data<7>" LOC = "AD25";
11 NET "data<6>" LOC = "AB22";
  NET "data<5>" LOC = "W21";
13 NET "data<4>" LOC = "W20";
  NET "data<3>" LOC = "AB21";
15 NET "data<2>" LOC = "AD22";
  NET "data<1>" LOC = "AE24";
17 NET "data<0>" LOC = "AF24";
  NET "data_clock" LOC = "AD26";
19 NET "dip_switches<7>" LOC = "R20";
  NET "dip_switches<6>" LOC = "R19";
21 NET "dip_switches<5>" LOC = "T26";
  NET "dip_switches<4>" LOC = "U26";
23 NET "dip_switches<3>" LOC = "U23";
  NET "dip_switches<2>" LOC = "V23";
25 NET "dip_switches<1>" LOC = "U25";
  NET "dip_switches<0>" LOC = "U24";
27 NET "trigger" LOC = "AA24";
```

References

[1] H. Celebi and H. Arslan, "Utilization of Location Information in Cognitive Wireless Networks," *IEEE Wireless Communications*, vol. 14, no. 4, pp. 6–13, Aug. 2007.

[2] N. E. Klepeis, W. C. Nelson, W. R. Ott, J. P. Robinson, A. M. Tsang, P. Switzer, J. V. Behar, S. C. Hern, and W. H. Engelmann, "The National Human Activity Pattern Survey (NHAPS): a Resource for Assessing Exposure to Environmental Pollutants," *Journal of Exposure Analysis and Environmental Epidemiology*, vol. 11, no. 3, pp. 231–252, May/Jun. 2001.

[3] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, 2nd ed. John Wiley & Sons, Inc., Feb. 2007, ISBN: 978-0-470-04190-1.

[4] C.-H. Lim, Y. Wan, B.-P. Ng, and C.-M. S. See, "A Real-Time Indoor WiFi Localization System Utilizing Smart Antennas," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 618–622, May 2007.

[5] A. Waadt, G. H. Bruck, and P. Jung, "An Overview of Positioning Systems and Technologies," in *Proc. of the 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*. Bratislava, Slovak Republic: IEEE, Nov. 24–27 2009, pp. 1–5.

[6] M. R. Gholami, E. Ström, and M. Rydström, "Indoor Sensor Node Positioning Using UWB Range Measurements," in *Proc. of the 17th European Signal Processing Conference (EUSIPCO)*. Glasgow, Scotland: EURASIP, Aug. 24–28 2009, pp. 1943–1947.

[7] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., ser. IEE Radar, Sonar and Navigation Series. Institution of Engineering and Technology, Oct. 2004, ch. Gyroscope Technology 2, pp. 118–126, ISBN 0863413587.

[8] M. Afgani, S. Sinanović, K. Khashaba, and H. Haas, "Radio Frequency Signature Correlation Based Speed Estimation for Indoor Positioning," *Journal of Communications (Invited Paper)*, vol. 4, no. 2, pp. 96–107, Mar. 2009. [Online]. Available: http://www.academypublisher.com/jcm/vol04/no02/jcm0402096107.html

[9] M. Afgani and H. Haas, "Speed Estimation Using Relative Radio Frequency Signature Matching," in *Proc. of the 66th IEEE Vehicular Technology Conference (VTC)*. Baltimore, Maryland, USA: IEEE, Oct. 1–3 2007, pp. 1970–1974.

[10] J. Mitola III, "Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio," Ph.D. dissertation, Royal Institute of Technology (KTH), May 2000.

[11] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[12] S. Haykin, D. J. Thomson, and J. H. Reed, "Spectrum Sensing for Cognitive Radio," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 849–877, Apr. 2009.

[13] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, Sep. 2006.

[14] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation Issues in Spectrum Sensing for Cognitive Radios," in *Proc. of the Asilomar Conference on Signals, Systems and Computers*, vol. 1. Pacific Grove, CA, USA: IEEE, Nov. 7–10, 2004, pp. 772–776.

[15] D. Cabric, A. Tkachenko, and R. W. Brodersen, "Spectrum sensing measurements of pilot, energy, and collaborative detection," in *Proc. IEEE Military Communications Conference (MILCOM)*, Washington, DC, USA, Oct. 23–25, 2006, pp. 1–7.

[16] F. F. Digham, M.-S. Alouini, and M. K. Simon, "On the Energy Detection of Unknown Signals Over Fading Channels," *IEEE Transactions on Communications*, vol. 55, no. 1, pp. 21–24, Jan. 2007.

[17] A. Fehske, J. Gaeddert, and J. Reed, "A New Approach to Signal Classification Using Spectral Correlation and Neural Networks," in *Proc. of the First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*. Baltimore, Maryland, USA: IEEE, Nov. 8–11 2005, pp. 144–150.

[18] M. Öner and F. Jondral, "Air Interface Recognition for a Software Radio System Exploiting Cyclostationarity," in *Proc. of the 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, vol. 3, Barcelona, Spain, Sep. 5–8, 2004, pp. 1947–1951.

[19] C. Krügel, T. Toth, and E. Kirda, "Service Specific Anomaly Detection for Network Intrusion Detection," in *Proc. of the 2002 ACM Symposium on Applied Computing*. Madrid, Spain: ACM, Mar. 10–14, 2002, pp. 201–208.

[20] R. J. Bolton and D. J. Hand, "Statistical Fraud Detection: A Review," *Statistical Science*, vol. 17, no. 3, pp. 235–249, Aug. 2002.

[21] J. Lin, E. Keogh, A. Fu, and H. V. Herle, "Approximations to Magic: Finding Unusual Medical Time Series," in *Proc. 18th IEEE Symposium on Computer-Based Medical Systems*, IEEE. Dublin, Ireland: IEEE, Jun. 23–24, 2005, pp. 329–334.

[22] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[23] M. Afgani, S. Sinanović, and H. Haas, "The Information Theoretic Approach to Signal Anomaly Detection for Cognitive Radio," *International Journal of Digital Multimedia Broadcasting (Invited Paper)*, vol. 2010, p. 18, Jun. 2010. [Online]. Available: http://www.hindawi.com/journals/ijdmb/2010/740594.html

[24] M. Afgani, S. Sinanović, and H. Haas, "Information Theoretic Approach to Signal Feature Detection for Cognitive Radio," in *Proc. of the 2008 IEEE Global Telecommunications Conference (GLOBECOM)*. New Orleans, Louisiana, USA: IEEE, Nov. 30–Dec. 4 2008, p. 5.

[25] M. Afgani, S. Sinanović, and H. Haas, "Anomaly Detection Using the Kullback-Leibler Divergence Metric," in *Proc. of the First International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*. Aalborg, Denmark: IEEE, Oct. 25–28 2008, p. 5.

[26] M. Afgani, S. Sinanović, and H. Haas, "Hardware Implementation of a Kullback-Leibler Divergence Based Signal Anomaly Detector," in *Proc. of the Second International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL) (Invited Paper)*. Bratislava, Slovak Republic: IEEE, Nov. 24 – 27 2009, p. 6.

[27] M. Afgani and H. Haas, "Information Content Analysis and Clustering for Signal Anomaly Detection," in *Proc. of the 70th IEEE Vehicular Technology Conference (VTC)*. Anchorage, Alaska, USA: IEEE, Sep. 20–23 2009, p. 5.

[28] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, ISBN: 0130422320, Dec. 2001.

[29] P. Höher, "A Statistical Discrete-Time Model for the WSSUS Multipath Channel," *IEEE Transactions on Vehicular Technology*, vol. 41, no. 4, pp. 461–468, Nov. 1992.

[30] R. B. Ertel, P. Cardieri, K. W. Sowerby, T. S. Rappaport, and J. H. Reed, "Overview of Spatial Channel Models for Antenna Array Communication Systems," *IEEE Personal Communications [see also IEEE Wireless Communications]*, vol. 5, no. 1, pp. 10–22, Feb. 1998.

[31] ETSI EN 300 744 v1.5.1 (2004-06), *Digital Video Broadcasting (DVB): Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television*, European Telecommunications Standards Institute (ETSI) Std., Jun. 2004.

[32] G. Durgin and T. Rappaport, "Theory of Multipath Shape Factors for Small-Scale Fading Wireless Channels," *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 5, pp. 682–693, 2000.

[33] A. Sampath and J. Holtzman, "Estimation of Maximum Doppler Frequency for Handoff Decisions," in *Proc. of the 43rd IEEE Vehicular Technology Conference (VTC)*. Secaucus, NJ, USA: IEEE, May 18–20 1993, pp. 859–862.

[34] W. Sheng and S. Blostein, "SNR-Independent Velocity Estimation for Mobile Cellular Communications Systems," in *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3. Orlando, Florida, USA: IEEE, May 13–17 2002, pp. 2469–2472.

[35] Y. R. Zheng and C. Xiao, "Mobile Speed Estimation for Broadband Wireless Communications," in *Proc. of the IEEE Wireless Communications and Networking Conference (WCNC)*. Kowloon, Hong Kong: IEEE, Mar. 11–15 2007, pp. 2420–2425.

[36] C. Xiao, K. Mann, and J. Olivier, "Mobile Speed Estimation for TDMA-Based Hierarchical Cellular Systems," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 4, pp. 981–991, Jul. 2001.

[37] C. Tepedelenlioğlu and G. Giannakis, "On Velocity Estimation and Correlation Properties of Narrow-Band Mobile Communication Channels," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 4, pp. 1039–1052, Jul. 2001.

[38] K. Anim-Appiah, "On Generalized Covariance-Based Velocity Estimation," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1546–1557, Sep. 1999.

[39] L. Zhao and J. W. Mark, "Mobile Speed Estimation Based on Average Fade Slope Duration," *IEEE Transactions on Communications*, vol. 52, no. 12, pp. 2066–2069, Dec. 2004.

[40] H. Zhang and A. Abdi, "Mobile Speed Estimation Using Diversity Combining in Fading Channels," in *Proc. of the IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 6. Dallas, TX, USA: IEEE, Nov. 29–Dec. 3 2004, pp. 3685–3689.

[41] A. Abdi, K. Wills, H. Barger, M.-S. Alouini, and M. Kaveh, "Comparison of the Level Crossing Rate and Average Fade Duration of Rayleigh, Rice and Nakagami Fading Models with Mobile Channel Data," in *Proc. of the 52nd IEEE Vehicular Technology Conference*, vol. 4. Boston, Massachusetts, USA: IEEE, Sep. 24–28 2000, pp. 1850–1857.

[42] L. Yang and M.-S. Alouini, "Average Level Crossing Rate and Average Outage Duration of Generalized Selection Combining," *IEEE Transactions Communications*, vol. 51, no. 12, pp. 1997–2000, Dec. 2003.

[43] P. Bello, "Characterization of Randomly Time-Variant Linear Channels," *IEEE Transactions on Communications Systems*, vol. 11, no. 4, pp. 360–393, Dec. 1963.

[44] I. A. Glover and P. M. Grant, *Digital Communications*, 2nd ed. Pearson Prentice Hall, 2004.

[45] L. Krasny, H. Arslan, D. Koilpillai, and S. Chennakeshu, "Doppler Spread Estimation in Mobile Radio Systems," *IEEE Communications Letters*, vol. 5, no. 5, pp. 197–199, May 2001.

[46] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels*, 2nd ed., ser. Wiley series in telecommunications and signal processing. John Wiley & Sons, Inc., 2005, ISBN: 978-0-471-64953-3.

[47] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*, 7th ed., A. Jeffrey and D. Zwillinger, Eds. Academic Press, Mar. 2007, ISBN-10: 0123736374.

[48] S. G. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. Academic Press, Sep. 1999, ISBN-10: 012466606X.

[49] R. Narasimhan and D. C. Cox, "Speed Estimation in Wireless Systems Using Wavelets," *IEEE Transactions on Communications*, vol. 47, no. 9, pp. 1357–1364, Sep. 1999.

[50] M. Gans, "A Power-Spectral Theory of Propagation in the Mobile-Radio Environment," *IEEE Transactions on Vehicular Technology*, vol. 21, no. 1, pp. 27–38, Feb. 1972.

[51] W. C. Jakes, Ed., *Microwave Mobile Communications*. IEEE Press, 1974.

[52] H. Zhang and A. Abdi, "Nonparametric Mobile Speed Estimation in Fading Channels: Performance Analysis and Experimental Results," *IEEE Transactions on Wireless Communications*, vol. 8, no. 4, pp. 1683–1692, Apr. 2009.

[53] W. Smith and D. Cox, "A Closed-Loop Doppler Measurement for Velocity Estimation in Mobile, Multipath Environments," in *Proc. of the IEEE Antennas and Propagation Society Conference (AP-S)*, vol. 3. Monterey, CA, USA: IEEE, Jun. 2004, pp. 2203–2206.

[54] A. Patcha and J.-M. Park, "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends," *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, Aug. 2007.

[55] M. Agyemang, K. Barker, and R. Alhajj, "A Comprehensive Survey of Numeric and Symbolic Outlier Mining Techniques," *Intelligent Data Analysis*, vol. 10, no. 6, pp. 521–538, Jan. 2006.

[56] V. Hodge and J. Austin, "A Survey of Outlier Detection Methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85–126, Oct. 2004.

[57] M. Markou and S. Singh, "Novelty Detection: A Review – Part 1: Statistical Approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, Dec. 2003.

[58] M. Markou and S. Singh, "Novelty Detection: A Review – Part 2: Neural Network Based Approaches," *Signal Processing*, vol. 83, no. 12, pp. 2499–2521, Dec. 2003.

[59] A. Lazarevic, L. Ertöz, V. Kumar, A. Ozgur, and J. Srivastava, "A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection," in *Proc. of the 2003 SIAM International Conference on Data Mining*, San Francisco, CA, USA, May 1–3 2003, pp. 25–36.

[60] Z. A. Bakar, R. Mohemad, A. Ahmad, and M. M. Deris, "A Comparative Study for Outlier Detection Techniques in Data Mining," in *Proc. of the 2006 IEEE Conference on Cybernetics and Intelligent Systems*. Bangkok, Thailand: IEEE, Jun. 7–9 2006, pp. 1–6.

[61] S. Albrecht, J. Busch, M. Kloppenburg, F. Metze, and P. Tavan, "Generalized Radial Basis Function Networks for Classification and Novelty Detection: Self-Organization of Optimal Bayesian Decision," *Neural Networks*, vol. 13, no. 10, pp. 1075–1093, Dec. 2000.

[62] M. F. Augusteijn and B. A. Folkert, "Neural Network Classification and Novelty Detection," *International Journal of Remote Sensing*, vol. 23, no. 14, pp. 2891–2902, Jul. 2002.

[63] W. Lee, S. Stolfo, and K. Mok, "A Data Mining Framework for Building Intrusion Detection Models," in *Proc. of the 1999 IEEE Symposium on Security and Privacy*. Oakland, CA , USA: IEEE, May 9–12 1999, pp. 120–132.

[64] J. Zhang and H. Wang, "Detecting Outlying Subspaces for High-Dimensional Data: The New Task, Algorithms, and Performance," *Knowledge and Information Systems*, vol. 10, no. 3, pp. 333–355, Oct. 2006.

[65] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying Density-Based Local Outliers," *ACM SIGMOD Record*, vol. 29, no. 2, pp. 93–104, Jun. 2000.

[66] R. Smith, A. Bivens, M. Embrechts, C. Palagiri, and B. Szymanski, "Clustering Approaches for Anomaly Based Intrusion Detection," in *Proc. of the Artificial Neural Networks in Engineering Conference (ANNIE)*, vol. 12. St. Louis, Missouri, USA: ASME, Nov. 5–8 2002, pp. 579–584.

[67] C. Manikopoulos and S. Papavassiliou, "Network Intrusion and Fault Detection: a Statistical Anomaly Approach," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 76–82, Oct. 2002.

[68] L. Tarassenko, P. Hayton, N. Cerneaz, and M. Brady, "Novelty Detection for the Identification of Masses in Mammograms," in *Proc. of the 4th International Conference on Artificial Neural Networks*, Paris, France, Oct. 9–13 1995, pp. 442–447.

[69] M. Desforges, P. Jacob, and J. Cooper, "Applications of Probability Density Estimation to the Detection of Abnormal Conditions in Engineering," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 212, no. 8, pp. 687–703, Jan. 1998.

[70] W. Lee and D. Xiang, "Information-Theoretic Measures for Anomaly Detection," in *Proc. of the IEEE Symposium on Security and Privacy*. Los Alamitos, CA, USA: IEEE, May 14–16 2001, pp. 130–143.

[71] C. C. Noble and D. J. Cook, "Graph-Based Anomaly Detection," in *Proc. of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Washington, DC, USA: ACM, Aug. 24–27 2003, pp. 631–636.

[72] Andreas Arning and Rakesh Agrawal and Prabhakar Raghavan, "A Linear Method for Deviation Detection in Large Databases," in *Proc. of the 2nd International Conference on Knowledge Discovery and Data Mining*, Portland, Oregon, USA, Aug. 2–4 1996, pp. 164–169.

[73] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed., ser. Wiley Series in Telecommunications, D. L. Schilling, Ed. John Wiley & Sons, Sep. 1991.

[74] H. L. V. Trees, *Detection, Estimation, and Modulation Theory: Part I*, 1st ed. John Wiley & Sons, Inc., 2001.

[75] S. Basu and M. Meckesheimer, "Automatic Outlier Detection for Time Series: An Application to Sensor Data," *Knowledge and Information Systems*, vol. 11, no. 2, pp. 137–154, Feb. 2007.

[76] C. Romesburg, *Cluster Analysis for Researchers*. Lulu Press, 2004.

[77] J. Ma and S. Perkins, "Online Novelty Detection on Temporal Sequences," in *Proc. of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, Aug. 24–27 2003, pp. 613–618.

[78] E. Keogh, J. Lin, S.-H. Lee, and H. Van Herle, "Finding the Most Unusual Time Series Subsequence: Algorithms and Applications," *Knowledge and Information Systems*, vol. 11, no. 1, pp. 1–27, Nov. 2006.

[79] D.-Y. Yeung and C. Chow, "Parzen-Window Network Intrusion Detectors," in *Proc. of the 16th International Conference on Pattern Recognition*, vol. 4. Quebec City, Canada: IEEE, Aug. 11–15 2002, pp. 385–388.

[80] E. Parzen, "On Estimation of a Probability Density Function and Mode," *Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962.

[81] E. Eskin, "Anomaly Detection over Noisy Data using Learned Probability Distributions," in *Proc. of the 17th International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., Jun. 29 – Jul. 2 2000, pp. 255–262.

[82] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. McGraw–Hill, 2002.

[83] J. Farrell and M. Barth, *The Global Positioning System and Inertial Navigation*, 1st ed., S. Chapman, Ed. McGraw-Hill, Dec. 1998, ISBN 007022045X / 9780070220454.

[84] C. Fritsche, A. Klein, and D. Wurtz, "Hybrid GPS/GSM Localization of Mobile Terminals Using the Extended Kalman Filter," in *Proc. of the 6th Workshop on Positioning, Navigation and Communication (WPNC)*. Hannover, Germany: IEEE, Mar. 19 2009, pp. 189–194.

[85] W. M. Yeung and J. K. Ng, "Wireless LAN Positioning Based on Received Signal Strength from Mobile Device and Access Points," in *Proc. of the 13th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. Daegu, Korea: IEEE, Aug. 21–23 2007, pp. 131–137.

[86] C. Kocks, E. Scheiber, D. Xu, A. Viessmann, S. Wang, G. H. Bruck, and P. Jung, "A Localization and Tracking Application for UWB," in *Proc. of the 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*. Bratislava, Slovak Republic: IEEE, Nov. 24–27 2009, pp. 1–5.

[87] J. Collin, O. Mezentsev, and G. Lachapelle, "Indoor Positioning System Using Accelerometry and High Accuracy Heading Sensors," in *Proc. of ION GPS/GNSS 2003 Conference*. Portland, OR: Institute of Navigation, Sep. 9–12, 2003.

[88] A. Lawrence, *Modern Inertial Technology: Navigation, Guidance, and Control*, 2nd ed., ser. Mechanical Engineering Series, F. F. Ling and W. H. Hart, Eds. Springer-Verlag, 1998, ISBN: 978-0-387-98507-7.

[89] A. Abdi and M. Kaveh, "Level Crossing Rate in Terms of the Characteristic Function: a New Approach for Calculating the Fading Rate in Diversity Systems," *IEEE Transactions on Communications*, vol. 50, no. 9, pp. 1397–1400, Sep. 2002.

[90] J. G. Proakis, *Digital Communications*, 4th ed., ser. McGraw-Hill Series in Electrical and Computer Engineering, S. W. Director, Ed. McGraw-Hill Higher Education, Dec. 2000.

[91] J. Robertson, E. W. Tallman, and C. H. Whiteman, "Forecasting Using Relative Entropy," Nov. 2002, FRB of Atlanta Working Paper No. 2002-22.

[92] H. Nakahara and S.-I. Amari, "Information-Geometric Measure for Neural Spikes," *Neural Computation*, vol. 14, no. 10, pp. 2269–2316, Oct. 2002.

[93] D. D. Falconer, F. Adachi, and B. Gudmundson, "Time Division Multiple Access Methods for Wireless Personal Communications," *IEEE Communications Magazine*, vol. 33, no. 1, pp. 50–57, Jan. 1995.

[94] M. Basseville, "Distance Measures for Signal Processing and Pattern Recognition," *Signal Processing*, vol. 18, no. 4, pp. 349–369, Dec. 1989.

[95] H. Shimazaki and S. Shinomoto, "A Method for Selecting the Bin Size of a Time Histogram," *Neural Computation*, vol. 19, no. 6, pp. 1503–1527, Jun. 2007.

[96] R. Krichevsky and V. Trofimov, "The Performance of Universal Encoding," *IEEE Transactions on Information Theory*, vol. 27, no. 2, pp. 199–207, Mar. 1981.

[97] D. H. Johnson, C. M. Gruner, K. Baggerly, and C. Seshagiri, "Information-Theoretic Analysis of Neural Coding," *Journal of Computational Neuroscience*, vol. 10, no. 1, pp. 47–69, Jan. 2001.

[98] H. Haas and S. McLaughlin, Eds., *Next Generation Mobile Access Technologies: Implementing TDD*. Cambridge University Press, ISBN: 13:9780521826228, Jan. 2008.

[99] Y. Benjamini and Y. Hochberg, "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *Journal of the Royal Statistical Society - Series B (Methodological)*, vol. 57, no. 1, pp. 289–300, 1995.

[100] Nokia, "Nokia N900 Specifications," Retrieved Dec. 13 2009. [Online]. Available: http://www.nokia.co.uk/find-products/all-phones/nokia-n900/specifications

[101] ARM, "The ARM Cortex-A9 Processors," Whitepaper, Sep. 2009, rev. 2.0 retrieved Dec. 13 2009. [Online]. Available: http://www.arm.com/pdfs/ARMCortexA-9Processors.pdf

[102] J. Detrey and F. de Dinechin, "Parameterized Floating-Point Logarithm and Exponential Functions for FPGAs," *Microprocessors and Microsystems*, vol. 31, no. 8, pp. 537–545, Feb. 2007, Special Issue on FPGA-based Reconfigurable Computing (3).

[103] G. Sutter and J.-P. Deschamps, "High Speed Fixed Point Dividers for FPGAs," in *Proc. International Conference on Field Programmable Logic and Applications (FPL)*. Prague, Czech Republic: IEEE, Aug. 31 – Sep. 2, 2009, pp. 448–452.

[104] N. Sorokin, "Implementation of High-Speed Fixed-Point Dividers on FPGA," *Journal of Computer Science & Technology*, vol. 6, no. 1, pp. 8–11, Apr. 2006.

[105] J. S. Walther, "A Unified Algorithm for Elementary Functions," in *Proc. Spring Joint Computer Conference*, vol. 38, American Federation of Information Processing Societies. Atlantic City, New Jersey, USA: AFIPS Press, May 18–20, 1971, pp. 379–385.

[106] Agilent Technologies, "N5102A Baseband Studio Digital Signal Interface Module," Technical overview, Dec. 9 2003, retrieved Dec. 30 2009. [Online]. Available: http://www.home.agilent.com/agilent/techSupport.jspx?pid=227081

[107] Xilinx, Inc., "Virtex-4 Family Overview," Product specification DS112, Sep. 28 2007, v3.0 retrieved Dec. 30 2009. [Online]. Available: http://www.xilinx.com/support/documentation/data_sheets/ds112.pdf

[108] Xilinx, Inc., "Virtex-4 FPGA User Guide," User guide UG070, Dec. 1 2008, v2.6 retrieved Dec. 30 2009. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug070.pdf

Publications and Patents

Page 152 M. Afgani, S. Sinanović, and H. Haas, "The Information Theoretic Approach to Signal Anomaly Detection for Cognitive Radio," *International Journal of Digital Multimedia Broadcasting (Invited Paper)*, May 2010.

Page 169 M. Afgani, S. Sinanović, K. Khashaba, and H. Haas, "Radio Frequency Signature Correlation Based Speed Estimation for Indoor Positioning," *Journal of Communications (Invited Paper)*, vol. 4, no. 2, pp. 96–107, Mar. 2009.

Page 181 M. Afgani, S. Sinanović, and H. Haas, "Hardware Implementation of a Kullback-Leibler Divergence Based Signal Anomaly Detector," in *Proc. of the Second International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL) (Invited Paper)*, Bratislava, Slovak Republic, Nov. 24 – 27, 2009.

Page 187 M. Afgani and H. Haas, "Information Content Analysis and Clustering for Signal Anomaly Detection," in *Proc. of the Vehicular Technology Conference (VTC)*. Anchorage, Alaska, USA, Sep. 20–23, 2009.

Page 192 M. Afgani, S. Sinanović, and H. Haas, "Information Theoretic Approach to Signal Feature Detection for Cognitive Radio," in *Proc. of the Global Telecommunications Conference (GLOBECOM)*, New Orleans, Louisiana, USA, Nov. 30–Dec. 4, 2008.

Page 199 M. Afgani, S. Sinanović, and H. Haas, "Anomaly Detection Using the Kullback-Leibler Divergence Metric," in *Proc. of the First International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*, Aalborg, Denmark, Oct. 25–28, 2008.

Page 204 M. Afgani and H. Haas, "Speed Estimation Using Relative Radio Frequency Signature Matching," in *Proc. of the 66th IEEE Vehicular Technology Conference (VTC)*, Baltimore, Maryland, USA, Oct. 1–3, 2007, pp. 1970–1974.

M. Afgani, H. Haas and S. Beauregard, "Method and Device for Determining the Speed of a Moving Entity" (under examination), European patent EP1977267 (A1), United States patent pub. no. US 2009/0189813 A1.

M. Afgani, H. Haas and S. Sinanović, "Analyzer for Signal Anomalies" (under examination), United States patent pub. no. US 2009/0248336 A1.

The Information Theoretic Approach to Signal Anomaly Detection for Cognitive Radio

Mostafa Afgani, Sinan Sinanović and Harald Haas
The University of Edinburgh,
AGB, King's Buildings, Mayfield Road,
Edinburgh EH9 3JL, UK
Email:{m.afgani, s.sinanovic,h.haas}@ed.ac.uk

Abstract

Efficient utilisation and sharing of limited spectrum resources in an autonomous fashion is one of the primary goals of cognitive radio. However, decentralised spectrum sharing can lead to interference scenarios that must be detected and characterised to help achieve the other goal of cognitive radio – reliable service for the end user. Interference events can be treated as unusual and therefore anomaly detection algorithms can be applied for their detection. Two complementary algorithms based on information theoretic measures of statistical distribution divergence and information content are proposed. The first method is applicable to signals with periodic structures and is based on the analysis of Kullback-Leibler divergence. The second utilises information content analysis to detect unusual events. Results from software and hardware implementations show that the proposed algorithms are effective, simple and capable of processing high speed signals in real-time. Additionally, neither of the algorithms require demodulation of the signal.

The Information Theoretic Approach to Signal Anomaly Detection for Cognitive Radio

I. Introduction

Cognitive radio (CR) is the term used to describe smart, reconfigurable wireless communications devices that are capable of automatically adjusting their operating characteristics in order to adapt to changes in the radio environment. The purpose of such a system is to enable efficient use of the available radio spectrum and provide reliable service to the end user [1]. The motivation for efficient spectrum utilisation arises from the fact that it is a very limited resource. Although the electromagnetic spectrum is (for all intents and purposes) infinite, only a small fraction of it is usable for personal wireless communications as we know it today. Furthermore, while the spectrum available remains fixed, the number of wideband wireless systems contending for access keeps growing – further compounding the spectrum scarcity problem.

Traditionally, the radio spectrum has been divided into a number of usable bands by regulatory bodies such as the Federal Communications Commission (FCC) in the USA and the Office of Communications (Ofcom) in the UK. Each of the bands are then assigned for exclusive access by a particular operator or service. A notable exception is of course the set of bands known as the industrial, scientific and medical (ISM) bands where emission from unlicensed consumer electronic devices is tolerated. While this restrictive approach to sharing the radio spectrum succeeds at providing a certain degree of interference protection, it is an inefficient use of the available resources since it is extremely unlikely that all of the bands are in use at the same time at a given place.

CR systems aim to simultaneously provide better quality of service and spectrum utilisation by dynamically moving the communication link from crowded or occupied bands to ones that do not appear to be in use by a primary licensed system at that instant. In order to carry out this task, secondary CR devices perform *spectrum sensing* – a procedure used to identify "holes" (free bands) in the spectrum and characterise the radio environment [2]. While there are a number of diverse approaches to problem [3], none of them are perfect. Energy detection based methods [4] are limited by signal to noise ratio (SNR) constraints while methods relying on cyclostationary features [5] are limited by the amount of *a priori* information available regarding the signal structure of the primary system. As a result of these shortcomings, spectrum sensing cannot completely avert the risk of interference that arises from dynamic spectrum sharing. Since interference generally leads to *anomalous* signal behaviour, an additional layer of simple signal processing algorithms that can help detect and characterise that behaviour is useful.

Anomaly detection refers to the process of locating unusual and unexpected events that may exist alongside nominal samples in a dataset. It is a process that is already utilised in large number of diverse application domains. Typical examples include the detection of: unauthorised access to computer systems [6], irregularities in vital signs such as electrocardiogram (ECG) traces [7], fraud in financial services [8], etc. An extensive survey of current anomaly detection techniques and application domains is provided in [9].

The aforementioned survey reveals that there are many different approaches to solving the anomaly detection problem – each with its own set of advantages and disadvantages. However, there is one drawback that is shared by most algorithms: computational complexity. The computational effort required makes it difficult to adapt these techniques for real-time and online processing of the input signal. This is unfortunate since any algorithm employed on an interactive communications system such as a CR platform must be capable of real-time operation to maintain a seamless user experience. To overcome this challenge, two complementary anomaly detection algorithms based on simple information theoretic measures have been developed and are presented in this paper. The first method utilises Kullback-Leibler divergence (KLD) [10] while the latter uses the information content of individual signal events [11]. The algorithms are easy to generalise and broadly define anomalies as events that lead to changes in the nominal probability distribution of the radio signal. As a result it is possible to employ the techniques for the detection of a wide range of disruptive events such as interference, timing errors, transmitter malfunction and so on.

KLD is a convenient and robust method of measuring the difference between two data sets in a statistical sense. Due to its versatility and general appeal, it finds use in fields as diverse as economics [12] and computational neuroscience [13]. As a statistical comparison tool, KLD can also be employed for the automatic and real-time detection of unusual (anomalous) data segments. The proposed KLD based technique utilises two data windows to perform a statistical comparison of neighbouring segments of signals with periodic structures (e.g. systems utilising time division multiple access [14]). Since segments separated by the signal period are expected to be analogous and hence have similar statistical characteristics, any deviation can be taken to imply the presence of an anomaly.

Unlike the KLD based method, the information content analysis (ICA) algorithm can also be applied to signals lacking any kind of periodic features. Information content is a quantity that is directly related to the probability of an event: the lower the probability, the higher the information content. Since anomalies are, by definition, rare (low probability), the associated information content is high. The proposed anomaly detection algorithm exploits this fact by analysing the signal

for high information content events.

Software implementations of the algorithms have been tested against a set of real wireless signals with promising results. Additionally, a Xilinx Virtex4 field-programmable gate array (FPGA) based hardware implementation of the KLD based method has shown that the algorithm is indeed capable of real-time analysis of high speed, high bandwidth signals.

A brief review of some of the anomaly detection algorithms described in literature is provided in Section II while the proposed algorithms are described in detail in Section III. Results from applying the techniques to the test signals and measures of performance are provided in Section IV. The hardware implementation is briefly discussed in Section V while Section VI concludes the paper with a summary of the contributions made and directions for future work.

II. Review of Existing Methods

Anomaly detection, also known as novelty detection or outlier detection, is a rich field of research with a very large body of work that exists in literature. The existence of multiple survey-type papers such as [9], [15], [16], [17], [18], [19] is a testament to the true extent of the subject of anomaly detection. It is therefore surprising to learn that it is still very much an active area of research lacking generic algorithms that can be applied universally to anomaly detection problems. Most of the methods described in literature are based on tightly constrained frameworks that apply to very specific classes of problems.

Existing techniques of anomaly detection can be separated into a handful of classes depending on the underlying approach. *Classification* based methods utilise supervised machine learning techniques to categorise nominal and anomalous behaviour while *clustering* and *nearest-neighbour* based techniques rely on measures of the relative distance between points of data. *Statistical* techniques detect anomalies by comparing the test data points against stochastic models of nominal behaviour. *Information theoretic* methods employ measures of information such as Kolmogorov complexity and entropy and work under the assumption that anomalies lead to a change in the information content. The algorithms proposed in this paper employ techniques that are both statistical and information theoretic in nature.

A statistical method of detecting anomalies in sensor data streams is proposed by Basu and Meckesheimer in [20]. Relying on the assumption that the data stream is continuous, the method exploits the fact that correlation between neighbouring data points is higher than between points separated by a relatively long length of time. The described algorithm detects anomalous events by comparing the value of each event against the median of a data set composed of neighbouring events. The performance of the method then depends on the size of the data set and the threshold. Since the algorithm expects an input where subsequent data points change little under nominal circumstances, it is unsuitable for use in typical communications systems where the signal strength can vary considerably even under normal operating conditions.

An algorithm for detecting anomalous network traffic by means of a combined statistical and information theoretic measure is described by Krügel *et al.* in [6]. For each packet, an anomaly score is computed by considering the packet type, length and payload distribution. If the combined score exceeds a certain threshold established through training, existence of an anomalous packet is signalled. Since the algorithm is designed for operation in the network layer, it cannot be utilised for link monitoring and anomaly detection in the physical layer.

Another set of statistical anomaly detection algorithms are presented by Desforges *et al.* and Yeung and Chow in [21] and [22] respectively. Both papers propose the use of the Parzen windows method of non-parametric smooth probability density estimation in order to establish a stochastic model of the data distribution. While Yeung and Chow simply test whether a data point belongs to a given model, Desforges *et al.* also construct a model of the test data set and compare that against the reference. Since the model of the underlying process is determined once at the onset of the experiments, the algorithms cannot cope with non-stationary systems. Utilisation of Parzen windows method for density estimation also makes the algorithms computationally expensive and unsuitable for real-time implementation.

A technique for detecting anomalous segments ("discords") in structured time series such as ECG traces is described by Lin *et al.* in [7]. Given a time series containing a discord, the algorithm essentially splits the series into a set of small segments and computes the mutual distance between the segments. If a segment is then found to have a minimum distance larger than a predefined threshold, it is labelled as anomalous. Although the algorithm shows promising results, it is unsuitable for real-time implementation due to the computational complexity cost associated with performing a search for anomalous segments.

Finally, the use of various information theoretic measures for anomaly detection is discussed by Lee and Xiang in [23]. However, the focus of the paper is on determining the suitability of data models through the use of measures such as entropy and relative entropy (*i.e.* KLD) rather than algorithms for detecting anomalies.

It is evident from this survey of existing techniques that there is a lack of algorithms that offer the features needed (non-parametric with a low computational complexity and the ability to handle non-stationary behaviour) to analyse radio frequency signal envelopes in real-time for anomalies.

III. Anomaly Detection Algorithms

The detection algorithms utilise KLD and information content analysis respectively to determine the presence of anomalies. Both quantities are ultimately calculated from estimates of the statistical probabilities of events in the signal.

Given two data sets P_n and Q_n, at time n, that contain samples from domain X, it is possible to obtain empirical estimates of the associated probability mass functions (PMFs) p_n and q_n from a non-parametric model such as a histogram. Once the PMF estimates are available, the KLD between them can be calculated using [10]

$$D\left(p_n \parallel q_n\right) = \sum_{x \in X} p_n\left(x\right) \log_2 \frac{p_n\left(x\right)}{q_n\left(x\right)}, \tag{1}$$

where $x \in X$. Since base-2 logarithm is used, the divergence is measured in *bits*. KLD between two PMFs is generally asymmetric: i.e. $D(p_n \parallel q_n) \neq D(q_n \parallel p_n)$ and the triangle inequality is not satisfied. When $p_n = q_n$, the KLD is zero; otherwise it is a positive real number ($\mathbb{R}_+$). For brevity and convenience, $D(p_n \parallel q_n)$ will also be referred to as D_n in this paper.

KLD belongs to a class of distance measures known as *f-divergence* (or *Ali-Silvey distances*). Some of the other distance measures that belong to the same class are: variational distance (symmetric), Hellinger distance (symmetric), and Chernoff distance (generally asymmetric) [24]. While they are all equally suitable for quantifying the statistical difference between two probability distributions, KLD and variational distance are the least complex and therefore easiest to implement. Variational distance is defined as

$$\begin{aligned} V(p_n \parallel q_n) &= \frac{1}{2}\sum_{x\in X}|p_n(x) - q_n(x)| \\ &= \frac{1}{2}\|p_n - q_n\|_1 \end{aligned} \tag{2}$$

where $\|p_n - q_n\|_1$ is commonly known as the $\mathcal{L}_1$ distance (L1D) between the PMFs p_n and q_n. Furthermore, KLD and L1D (and hence the variational distance) are related by the inequality [10]

$$D(p_n \parallel q_n) \geq \frac{1}{2\ln 2}\|p_n - q_n\|_1^2 . \tag{3}$$

Crucially, it states that $D(p_n \parallel q_n)$ is bounded by $\|p_n - q_n\|_1^2$ and not $\|p_n - q_n\|_1$. It is an important distinction as it implies that for certain PMF pairs the KLD may in fact be *smaller* than the L1D. For a pair of largely dissimilar PMFs*, as is generally the case when comparing an anomalous data set against a nominal reference, larger distance magnitudes are obtained from KLD rather than L1D. However, when both PMFs are similar (e.g. a nominal data set and the reference) it can lead to L1D values that are larger compared to KLD – increasing the likelihood that false positives are detected. As a result, it is expected that KLD is better suited for statistical anomaly detection compared to L1D. This is confirmed by the results seen in Section IV where the performance of a KLD based algorithm is compared against one based on L1D. The algorithm for anomaly detection using KLD is described in Section III-B.

Information content analysis is another technique based on an information theoretic quantity that can be utilised for the detection of anomalies. The amount of information, $I_n(x)$, conveyed by any discrete random event, x_n, at time n, is directly related to its probability of occurrence, $p_n(x)$ [11]:

$$I_n(x) = -\log_2\{p_n(x)\}; \quad n = 1, 2, \ldots \tag{4}$$

Since base-2 logarithm is used once again, information is also measured in *bits*. The equation implies that an event with a very high probability of occurrence carries very little information while a large amount of information is conveyed by the occurrence of rare events (i.e. $I_n(x) \to \infty$ as $p_n(x) \to 0$). Information is always real, positive ($\mathbb{R}_+$) and monotonically increasing with decreasing values of event probability. ICA is essentially a non-linear scaling function that favours the unusual.

*Differences that are large enough to produce a $\mathcal{L}_1$ distance of $2\ln 2$ or greater, to be precise.

A. Histogram and PMF Estimation

It is clear from (1) and (4) that both KLD analysis and ICA require estimation of empirical event probabilities. One approach to obtaining the necessary estimates is via event histograms. In addition to being simple to implement, histograms are non-parametric – implying that no assumptions need to be made regarding the underlying distribution of the sample data.

For samples that originate from domain X, the histogram is obtained by first partitioning X into bins B such that

$$X = \bigcup_{l=1}^{\beta} B_l$$

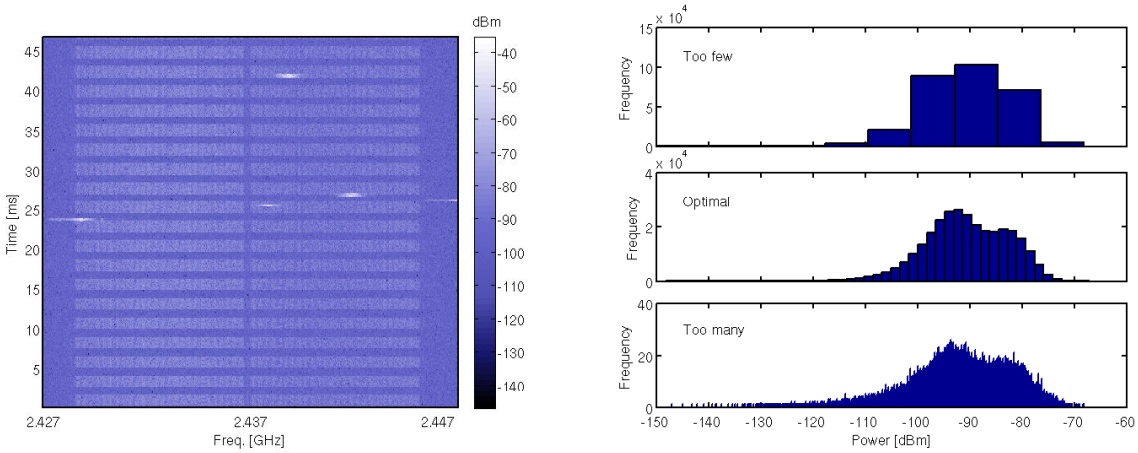
and then counting the number of samples that belong each bin. β is the total number of bins used to construct the histogram. Once the histogram is available, the empirical PMF of the sample set is easily obtained by simply dividing the histogram by the cardinality of the set.

Given a statistically significant sample size, it is clear that the only parameter that affects the quality of the PMF estimate obtained is the bin allocation B. If the partitions are then assumed to be equidistant for simplicity, the only variable that remains is the number of bins utilised: β. The effect of β on the histogram of a random process is shown in Fig. 1. The random process in question is the instantaneous power density at any time-frequency point of the signal shown in Fig. 1a. It is a wireless local area network (WLAN) signal experiencing bursts of interference from a Bluetooth (BT) device. Histograms of the power density obtained using three different values of β are shown in Fig. 1b.

When a small number of bins are utilised, i.e. β is small, the histogram is insensitive to small scale variations in the input. As a result of the poor resolution, the estimated model fails to adequately capture the subtleties in the behaviour of the underlying random process. On the other hand, when the value of β utilised is too large, the resolution is too high and the histogram is overly sensitive – resulting in an estimate that is noisy. The optimal value of β yields a good balance between resolution and sensitivity.

A method of computing the optimal bin size (and hence the optimal β) for constructing a histogram, subject to some minimum mean square error criterion, is provided in [25]. While the algorithm described therein is conceptually simple, it unfortunately requires the use of exhaustive search to iteratively minimise a certain cost function – making it too computationally expensive to be evaluated in real-time on a hand-held mobile device with limited energy and processing power.

The impact of β on the performance of each of the anomaly detection algorithms has been investigated and the results are presented in Section IV-B4.



(a) Signal with interference

(b) Effect of bin count

Fig. 1. (a) Spectrogram of a wireless local area network (WLAN) signal experiencing interference from a Bluetooth device. The regular pattern is a single WLAN frame repeatedly transmitted by a signal generator. The frequency hopping nature of the Bluetooth transmission is clearly visible in the plot. (b) Impact of the number of bins utilised, β, on a histogram of the instantaneous power density of the WLAN signal. When β is too small, the histogram is insensitive to small changes and does not effectively capture the subtleties of the process. At the other extreme, it is too sensitive and therefore susceptible to noise. The optimal β is with respect to some minimum error criterion [25] and provides a good balance between resolution and sensitivity.

Depending on the choice of B, there may be zeros in the estimated PMFs due to the presence of empty histogram bins. Under such circumstances, calculation of the KLD using (1) can be a problem as it leads to instances where $0 \log_2 \{0/q_n(x)\}$ or $p_n(x) \log_2 \{p_n(x)/0\}$ have to be evaluated. While it is certainly possible to handle these as special cases by setting them to 0 and ∞ respectively through continuity arguments, it may be better to simply avoid zeros in the PMFs. It is possible to avoid empty histogram bins and hence zeros in PMFs by adding a small number, λ, to every bin of the histogram. As pre-loading of histogram bins in this manner undoubtedly distorts the estimate of the true PMF, the pre-load value must be carefully chosen. According to the work done by Krichevsky and Trofimov [26], [27], $\lambda = 0.5$ is a good choice.

B. Algorithm Based on KLD

The capability of KLD to quantise the difference, in a statistical sense, between two data sets to single real value is ideal for use in anomaly detection since it provides a convenient detection metric. A general description of the algorithm is provided here while a discussion of the optimisations needed for an efficient hardware implementation is provided in the section (III-C) that follows.

The flowchart in Fig. 2 shows the proposed algorithm. At time n, the process starts with the acquisition of the two data sets to be compared using KLD. One of the data sets is a reference (Q_n) while the other is the one under test (P_n). If the samples in the data sets do not directly represent the parameter of interest, they must be processed. Once the data sets have been suitably transformed, the associated PMFs $p_n(x)$ and $q_n(x)$ are estimated and used to compute the KLD, D_n. If D_n is then observed to be larger than some predefined KLD threshold, D_{th}, the test data set may be anomalous.

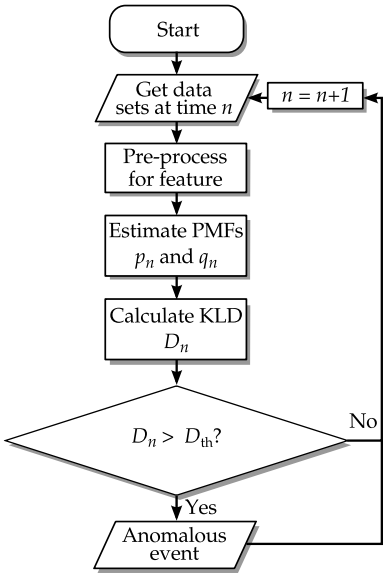


Fig. 2. The algorithmic flowchart. KLD is used to compare the statistical distribution of a test data set against that of a reference. If the divergence, D_n, is greater than some predefined threshold, D_{th}, the test set may be anomalous.

This general approach to detecting anomalies using KLD can be easily adapted for use with signals containing periodic structures. One example of such a signal is IEEE 802.16e wireless broadband (WiBro) which utilises time division duplexing (TDD) [28]. Periodic signals are expected to have statistics that are also periodic – implying that segments of the signal separated by the period, T_p, should have probability distributions that are very similar under normal circumstances.

TABLE I
COMPLEXITY ANALYSIS OF KLD

Operations	Σ	$\times$	$\div$	log	Total
	$2\|P_n\|+\beta$	β	3β	β	$2\|P_n\|+6\beta$
Memory	$\|P_n\|+\psi+2\beta$				

Therefore, by simply acquiring the data sets P_n and Q_n from two sliding signal windows of length T_w with centres separated by T_p, the proposed algorithm can be utilised for the detection of anomalies in periodic signals. KLD analysis can be performed on the signal envelope itself and as a result, demodulation is unnecessary and the only *a priori* information required by the algorithm is the signal period T_p.

While the steps required to compute the KLD are all simple and straightforward, the storage (data buffers) and the number of arithmetic operations required grow linearly with the size of input data sets. As these data sets can be very large when analysing high speed signals, it can easily lead to scenarios where it may not be possible to provide for the resources required by the algorithm. Analysis of the algorithm's complexity and memory requirements follows and is summarised in Table I.

The input data sets P_n and Q_n themselves require a buffer capable of holding at least $|P_n|+\psi$ elements, where ψ is the number of samples corresponding to the signal period and $|P_n|$ $(=|Q_n|)$ is the size of the data windows in samples. Only a single buffer is required for the input data since one of the data sets is essentially just a ψ-delayed version of the other in this case. Computing the frequency count over the bins (B), for the purpose of estimating the histograms, requires up to $|P_n|$ additions for each of the two windows. Once the histograms are available, the PMFs are obtained by dividing the frequency count in each of the β bins by $|P_n|$. Two buffers of size β each are then required to store the resulting PMFs. Computation of the KLD from the PMFs then require a further β divisions, logarithms, multiplications and additions respectively.

C. Hardware Implementation

The analysis performed in the previous section (III-B) reveals that a direct interpretation of the algorithm to hardware would be inefficient, inflexible and computationally expensive. The inefficiency arises from the fact that at each successive time instance, the PMFs and the KLD are completely recalculated; even though it is only a single sample that changes in each of the data sets. The inflexibility comes from the fact that the computational complexity depends on β, implying that a direct interpretation would be limited by the initial choice of the histogram bin resolution. Finally, logarithms and divisions can be very costly to implement in hardware. Fortunately, there are several well-known methods that can be adopted to overcome each of these challenges.

The complexity that arises from the division operation in (1) can be removed by exploiting the identity $\log\left(\frac{a}{b}\right)=\log(a)-\log(b)$:

$$\begin{aligned} D_n &= \sum_{x\in X} p_n(x)\log_2\frac{p_n(x)}{q_n(x)} \\ &= \sum_{x\in X} p_n(x)\left\{\log_2\left[p_n(x)\right]-\log_2\left[q_n(x)\right]\right\}. \end{aligned} \quad (5)$$

The division operation is exchanged for a subtraction and a second base-2 logarithm operation which can be implemented in a very efficient manner by means of a lookup table.

Further efficiency improvements can be achieved by making application specific changes to the way the algorithm is evaluated. Since the purpose of the algorithm is to analyse periodic signals by means of two sliding windows, it holds that at any given instance, only one sample in each of the data sets changes. This in turn implies that only a maximum of 4 histogram/PMF bins need to be updated at that instant – two for each data set/window. The two bins per window account for the freshly acquired sample (bin frequency count incremented by one) and the sample that is dropped at the end of the window (bin frequency count reduced by one). An illustration is provided in Fig. 3.

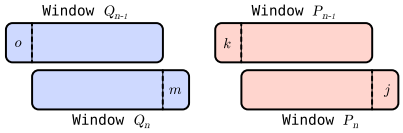


Fig. 3. At any given time, only a maximum of four histogram bins need to be updated (two per window). Bins incremented (fresh samples) are denoted by j and m while bins decremented (old samples at the end of the window) are denoted by k and o.

This also means that the KLD values change very little between subsequent time steps for this particular application – suggesting that it is possible to re-write the equation (5) in the form of a differential equation:

$$\begin{aligned} D_n &= \sum_{x\in X} p_n(x)\left\{\log_2\left[p_n(x)\right]-\log_2\left[q_n(x)\right]\right\} \\ &= D_{n-1} \\ &\quad -p_{n-1}(j)\left\{\log_2\left[p_{n-1}(j)\right]-\log_2\left[q_{n-1}(j)\right]\right\} \\ &\quad +p_n(j)\left\{\log_2\left[p_n(j)\right]-\log_2\left[q_n(j)\right]\right\} \\ &\quad -p_{n-1}(k)\left\{\log_2\left[p_{n-1}(k)\right]-\log_2\left[q_{n-1}(k)\right]\right\} \\ &\quad +p_n(k)\left\{\log_2\left[p_n(k)\right]-\log_2\left[q_n(k)\right]\right\} \\ &\quad -p_{n-1}(m)\left\{\log_2\left[p_{n-1}(m)\right]-\log_2\left[q_{n-1}(m)\right]\right\} \\ &\quad +p_n(m)\left\{\log_2\left[p_n(m)\right]-\log_2\left[q_n(m)\right]\right\} \\ &\quad -p_{n-1}(o)\left\{\log_2\left[p_{n-1}(o)\right]-\log_2\left[q_{n-1}(o)\right]\right\} \\ &\quad +p_n(o)\left\{\log_2\left[p_n(o)\right]-\log_2\left[q_n(o)\right]\right\} \end{aligned} \quad (6)$$

where the four bin indices j, k, m, and o are assumed to be unique. If not, any duplicate terms in the equation are set to zero.

The differential equation form of KLD (DKLD) shows that its computational complexity is no longer dependent on the number of histogram bins utilised in evaluating the PMFs. Assuming that D_{n-1} is available, only 16 additions/subtractions,

TABLE II
COMPLEXITY ANALYSIS OF DKLD

	Σ	×	÷	log	Total
Operations	20	8	16	16	60
Memory	$\lvert P_n\rvert + \psi + 2\beta$				

TABLE III
COMPLEXITY ANALYSIS OF FP-DKLD

	Σ	×	÷	log	Total
Operations	20	8	0	0	28
Memory	$\lvert P_n\rvert + \psi + 2\beta + L$				

16 logarithms, and 8 multiplications are needed to calculate D_n – regardless of the value of β. This opens the path for a fixed complexity, flexible and efficient implementation that can be easily updated to accommodate a wide range of histogram resolutions.

The computational complexity and storage requirement of DKLD is shown in Table II. Comparisons against the unmodified, direct interpretation version of KLD (Table I) reveals that while memory utilisation remains unchanged, there is a vast difference in the number of operations required. Regardless of the window size and histogram bin count, 60 operations are needed to compute the KLD. In addition to the 16 additions/subtractions, 16 logarithms and 8 multiplications required for the DKLD (6), 4 more additions/subtractions are required to update the affected histogram bins and 16 divisions are required to obtain the necessary PMFs at times $n-1$ and n from the histogram.

Switching to a fixed-point representation and using a lookup table for base-2 logarithms provides further reductions in complexity at the expense of a slight increase in the memory requirements. The size of the table, L, then dictates the precision available. Additionally, if $|P_n|$ is chosen such that it is always a power of two (PoT), i.e.

$$|P_n| = 2^{\sigma}, \quad \sigma = 0, 1, 2, \ldots$$

no division operations are required to obtain the PMFs since division by a PoT is simply a bit-shift operation that costs nothing in hardware.

The complexity and storage requirements of a fixed-point DKLD (FP-DKLD) based algorithm utilising a log lookup table and PoT constraint on $|P_n|$ is shown in Table III. It can be seen that with some simple changes and constraints, the complexity of the anomaly detection algorithm can be greatly reduced – allowing for efficient and high speed hardware implementations. Results obtained from a Xilinx Virtex4 FPGA implementation of the FP-DKLD algorithm are shown in Section V.

D. Information Content Analysis Algorithm

Unlike the KLD based anomaly detection algorithm just described, the ICA based method analyses individual input samples rather than aggregate sets of data. The information conveyed by the events is the detection metric utilised. The following is a general description of the algorithm.

First and foremost, it is necessary to establish the type of event that is under observation. This can be any property that is associated with the signal under test (e.g. instantaneous amplitude, phase, or power). If the event type chosen is measurable directly from the signal envelope, demodulation is unnecessary for anomaly detection. The ICA algorithm utilises supervised learning to establish a reference histogram (and hence probability) of events; therefore, some clean signal is required for training. Once the reference histogram is obtained, online analysis of the signal under test can commence. Events from the test signal are extracted and used to update the reference histogram. This yields updated event probabilities and hence the associated information content. If the information content $I_n(x)$ of any event x_n, at time n, is above some predefined threshold I_{th}, an anomaly may be present.

Once again, it is clear that the event histogram plays a central role in the anomaly detection algorithm. It has been stated previously in Section III-A and illustrated by Fig. 1 that the number of bins utilised, β, has a significant impact on the sensitivity of the histogram and hence the effectiveness of the detection algorithms. When β is too small, anomalous events may not be detected due to poor sensitivity – leading to missed detections. On the other hand, when β is too large, even nominal events will appear to have low probability – leading to a large number of false positives. It is therefore necessary to find a β that offers a good balance between sensitivity and probability of detecting false positives.

The event histograms shown in Fig. 1b reveal another potential challenge for the ICA algorithm. It can be seen that the histograms have long tails with numerous low probability (i.e. high information content) events even when the signal is behaving nominally. Although this is expected for any analogue signal transmitted over a lossy physical channel, it raises the possibility that numerous false positives are observed at a detector that employs a simple information content threshold. It is certainly possible to reduce the number of tail events by using a smaller number of bins, but that leads to reduction in sensitivity and hence an increase in the probability of missed detections.

Examination of the interference scenario in Fig. 1a reveals an important distinction between anomalous events and the underlying signal – anomalies tend to appear in clusters while nominal low-probability signal events are decidedly "singular". This difference is the key to reducing the number of false positives while still maintaining a low rate of missed detections. The proposed algorithm is easily augmented to benefit from this insight: instead of triggering on individual high information content events, the detector must search for contiguous groups of events that exceed the predefined information content threshold.

A flowchart of the algorithm with simplified clustering is shown in Fig. 4. The general approach is as before, with the exception of the last step. With the simple clustering extension, detection of an anomaly is signalled only when a contiguous sequence of N previous events and the current event exceeds a

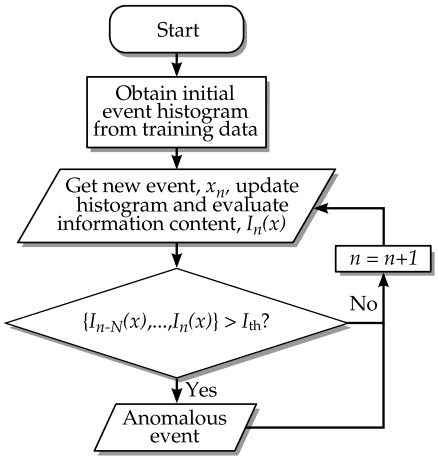


Fig. 4. Flowchart of information content analysis algorithm with clustered anomaly detection. First, clean data is used to initialise the reference event histogram and event probabilities. Then, events in the signal under test are used to update the reference histogram and event probabilities. The updated values are used to estimate the information content of the events. If the information content of $N+1$ contiguous events exceed a predefined threshold, $I_{\rm th}$, an anomaly may be present in the signal under test.

predefined information content threshold. Sequence detection is used rather than full two-dimensional clustering to minimise the complexity of the algorithm. This is permissible since a sequence can be considered as a one-dimensional cluster. The effect of the cluster size utilised on the detector performance is examined in Section IV-B5.

The discrimination threshold is an important aspect of any detector. While the optimum threshold is problem and cost function specific, it is generally chosen to minimise missed detections while still maintaining a low rate of false positives. For the proposed anomaly detection algorithm, it is not possible to define a single information content threshold, $I_{\rm th}$, that is suitable for use with any arbitrary signal. $I_{\rm th}$ is signal specific and may be set automatically using information obtained from the clean training data. After the reference event histogram and probabilities have been estimated, the reference information content associated each event type can be easily computed using (4). For β bins, the standard deviation, $\sigma_{\mathrm{I}\langle\beta\rangle}$, of the reference information content provides a measure of the spread and may be used to obtain the threshold:

$$I_{\rm th} = m\sigma_{\mathrm{I}\langle\beta\rangle}. \tag{7}$$

m is a multiplicative factor greater than 1. The effect of $I_{\rm th}$ on detector performance is investigated in Section IV-B3.

Due to the simplicity of the ICA algorithm, its runtime operational complexity and memory requirements are negligibly small. On completion of the initial training phase, a small buffer capable of holding just $\beta + 1$ elements is required to store the event histogram and the total events count. At runtime, analysis of an event requires 2 additions to increment the relevant bin count and the total events count. Division of the incremented bin count by the total is then needed to obtain the event probability. After the probability is computed, a single base-2 logarithm is needed to calculate the event's information content. An additional buffer capable of holding $N + 1$ elements is also needed to accommodate information content clustering. A summery of the complexity analysis is provided in Table IV. It reveals that in addition to being negligibly small, the fixed runtime operational complexity is independent of any algorithmic parameter (e.g. histogram resolution) – suggesting that fast and efficient implementations for power limited hand-held devices are possible.

TABLE IV
RUNTIME COMPLEXITY ANALYSIS OF ICA

	Σ	$\times$	$\div$	log	Total
Operations	2	0	1	1	4
Memory	$\beta + N + 2$				

IV. RESULTS

In order to evaluate the performance of the proposed anomaly detection schemes, signals with different classes of abnormalities are employed as test cases. All of the signals under test are actual radio frequency transmissions captured using spectrum analysis hardware and therefore represent scenarios likely to be encountered by real world wireless devices.

Analysis of the test signals are provided in the following section while detailed performance analysis of the algorithms based on parameters such as histogram bin resolution, data window size, sampling rate and cluster length are provided in Section IV-B.

A. Data Analysis

Of the four data sets available, the first three are used to demonstrate the PMF divergence analysis (KLD/L1D) based technique while the last is used to demonstrate the ICA based technique.

1) Test Signal A: The signal is shown as a time series in the upper half of Fig. 5. It consists of a single WLAN frame that repeats with a period of 2.45 ms and a burst of interference from a Bluetooth device that is visible at 11.5 ms. The signal is similar to that shown earlier in Fig. 1.

Both KLD and L1D are used to analyse the signal for the purpose of obtaining results that can be directly compared. Two windows with a duration of 256 μs each are employed to process the time series signal. The window centres are separated by 2.45 ms to match the WLAN frame repetition interval. The windows estimate the PMFs of the signal power. The number of histogram bins utilised is the optimal value (51 in this case) as obtained from the algorithm proposed by Shimazaki and Shinomoto [25]. In any case, it is shown in a subsequent section (IV-B4) that the number of histogram bins used does not have a significant impact on the outcome – therefore, an arbitrary but reasonable choice such as 32 can also be used instead.

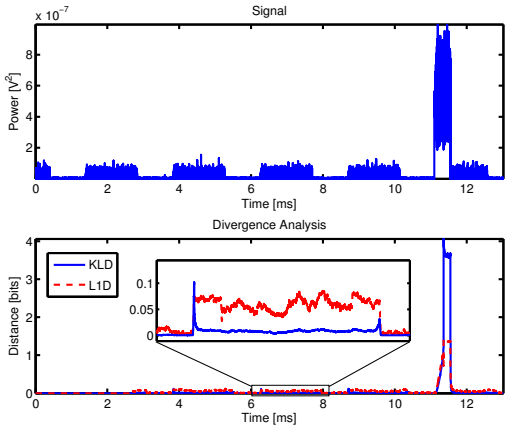


Fig. 5. *Top*: WLAN signal with interference at 11.5 ms from a Bluetooth device. *Bottom*: Both KLD and L1D analysis of the signal result in detection of the anomaly. KLD appears to be better than L1D since it leads to a much larger peak and lower noise.

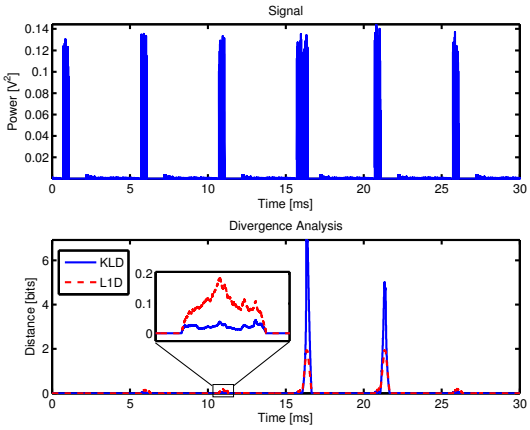


Fig. 6. *Top*: Wireless broadband (WiBro) signal. The uplink sub-frame at 16 ms has a longer duration than others and is therefore unusual in this context. *Bottom*: KLD & L1D analysis both reveal the anomalous segment of the WiBro signal. A second peak is obtained when the signal returns to normal – this is due to the twin-windowing nature of the anomaly detection algorithm. Once again, KLD analysis results in a larger peak and lower noise.

The result of the analysis is also shown in Fig. 5. Both KLD and L1D based methods are successful at detecting the presence of the anomaly (BT interferer). However, it is clear that KLD is the better choice as it produces a larger peak compared to L1D when the anomaly is detected. The baseline noise level with KLD is also much lower than that obtained with L1D – confirming the hypothesis presented in Section III.

2) Test Signal B: The second signal under test is a wireless broadband (WiBro) signal. It is shown in the upper half of Fig. 6. Due to the proximity of the recording equipment to the mobile terminal (MT), the uplink (UL) sub-frames show a higher power level then downlink (DL) sub-frames. From the plot, it can be seen that the UL sub-frame at 16 ms is longer than any of the other UL sub-frames. In context of this particular signal snapshot, this behaviour is unusual and hence can be considered to be anomalous. Once again, two windows with a duration of 256 μs each are employed to estimate the signal power PMFs. The window centres are separated by 5 ms – corresponding to the frame period of the signal. The optimal histogram bin allocation scheme in [25] is once again used to determine the number of bins utilised ($\beta = 51$).

Analysis of the signal is also shown in Fig. 6. A sharp peak in the divergence at 16 ms reveals the presence of the unusual UL sub-frame. A second peak is obtained when the signal returns to normal in the following UL sub-frame. Once again, the superiority of KLD over L1D as a divergence metric is demonstrated by the larger peaks and lower baseline noise levels.

3) Test Signal C: The third signal used to test the detection capabilities of the divergence based algorithm is shown in Fig. 7. It depicts communication between a mobile terminal and base station using the WiBro standard. Since the recording is made at the MT, there is significantly more power in the UL sub-frames. Although unnoticeable in the time series, the UL sub-frame at 20 ms contains an additional command

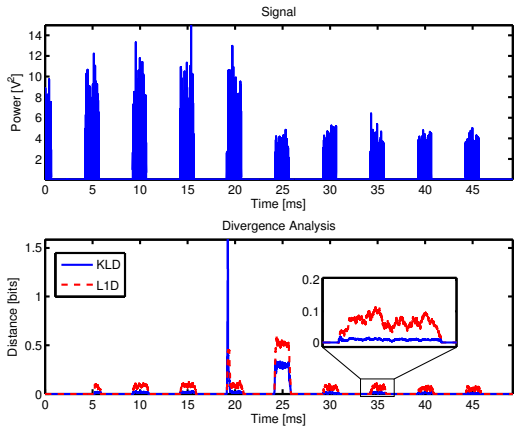
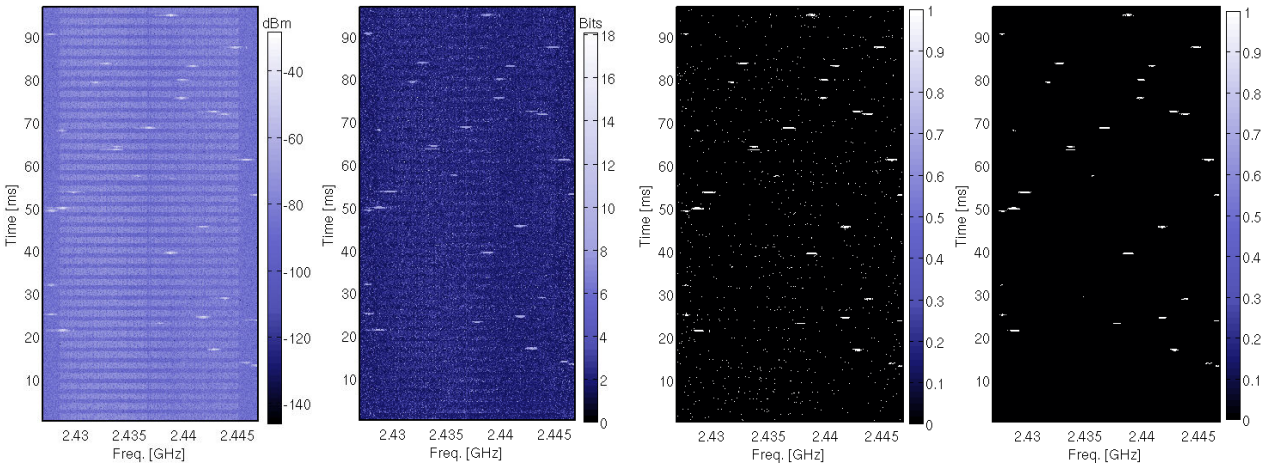


Fig. 7. *Top*: Message exchange between a mobile terminal and a base station using the WiBro communications protocol. An extra command sequence in the uplink sub-frame at 20 ms initiates the power control loop. *Bottom*: The extra command sequence (20 ms) and subsequent power change (25 ms) are both revealed by KLD/L1D analysis of the signal. A larger KLD peak results from the anomalous command but not from the power change. L1D also leads to larger noise levels compared to KLD.

sequence that triggers the subsequent change in the transmit power observed at 25 ms. As a result, there are effectively two unusual events in the signal: the extra command and the subsequent change in power level. The parameters utilised for analysis of the signal are identical to those used in Section IV-A2.

The divergence analysis plot in Fig. 7 shows that both anomalies can be successfully detected using KLD and L1D. Since the width of a KLD peak corresponds to the temporal duration of the anomaly responsible, the first peak at 20 ms is very sharp as it is due to the extra command sequence in the UL sub-frame. Since the subsequent change in power at 25



(a) Test signal (b) Information content (c) Anomalies ($N = 0$) (d) Anomalies ($N = 2$)

Fig. 8. Analysis of a WLAN signal with interference from a Bluetooth device. (a) Spectrogram of the original signal. Data bursts from the Bluetooth device are clearly visible as high power, lightly shaded patches. (b) Information content of events (power density at any given time-frequency point) in the signal. (c) Anomalies detected (light patches) using a threshold of $1.25\sigma_{\mathrm{I}\langle 16\rangle}$ (7.15 bits) and a cluster length of 1. The result is noisy and there are a lot of false positives (appearing as singular, lightly shaded spots). (d) Anomalies detected when the threshold is left unchanged at $1.25\sigma_{\mathrm{I}\langle 16\rangle}$ and the cluster length is increased to 3. The outcome is now much cleaner with virtually zero false positives.

ms affects the entire UL sub-frame, the second peak is much broader and spans the entire sub-frame.

The plot also reveals that for the second anomalous event (power change), KLD is smaller than the associated L1D. This is not unexpected since it has been hypothesised in Section III that for differences that lead to a L1D of $2\ln 2$ or smaller, L1D can be larger than KLD. It is also the reason why L1D generally leads to larger baseline noise levels compared to KLD.

4) Test Signal D: The final test signal is used to evaluate the ICA algorithm. It is similar in nature to the signal shown in Fig. 1a. Spectrogram of the test signal is shown in Fig. 8a. It is a much longer signal with numerous interference events to provide a statistically significant sample size. The plot depicts a real WLAN signal with Bluetooth interference captured over the air-interface. The WLAN signal consists of a single frame that is repeated periodically by a vector signal generator. The characteristic frequency hopping pattern of the Bluetooth device marks the locations of the interference (anomalous) events.

The signal spectrogram is estimated from the time series using non-overlapping Hamming windows that are 64 μs long. A 1024 point FFT (fast Fourier transform) is used to obtain a frequency resolution of approximately 20 kHz. The signal event under observation is the instantaneous power density at any given time-frequency coordinate. The first 10 ms of the signal is assumed to be free from interference and is therefore used for training purposes. 16 equally spaced histogram bins divide the range between the maximum and minimum power densities observed in the training data. The $\sigma_{\mathrm{I}\langle 16\rangle}$ for the training data is 5.72 bits.

Fig. 8b shows the information content of events in the test signal. As expected, the anomalous events have a higher information content and there are highlighted while the regular underlying structure is suppressed. The plot also shows that there is a lot of noise (tiny spots of high information content) from individual low-probability signal events that are otherwise nominal. The reason for this behaviour has been outlined in Section III-D.

Anomalies detected using a threshold of $1.25\sigma_{\mathrm{I}\langle 16\rangle}$ and a cluster length of 1 (i.e. only the current event) is shown in Fig. 8c. It is immediately obvious from the large number of small, lightly shaded spots that there are a lot of false positives. Again, singular low-probability signal events are responsible since they can potentially have higher information content than actual anomalous events. Keeping I_{th} the same and increasing the cluster length to 3 yields the result shown in Fig. 8d. It reveals that a simple change in the cluster length is sufficient for reducing the number of false positives to virtually zero.

B. Performance Analysis

The analysis of test signals presented in Section IV-A show that both algorithms perform well for the parameter combinations chosen. In order to investigate and quantify the impact of other parameter choices, it is necessary to define and utilise metrics that reflect performance.

For the divergence based technique, the ratio between the anomaly detection peak and the maximum of the baseline noise level is a good indicator of performance since it is a reflection of the range over which a threshold can be applied. It can be seen from the results presented in Fig. 5, 6 and 7 that KLD is an extremely effective discriminator for statistical changes in the observed data. Even with such *real test vectors captured over-the-air*, the KLD peaks produced by anomalous events are many orders of magnitude larger than baseline noise levels associated with nominal data. As a result, 100% probability of detection can be achieved over a wide range of

KLD threshold values (the anomalous peak is approximately 140 times as large as the background noise level in Fig. 6) while still guaranteeing a 0% probability of false positives – making such classical measures of detector performance inadequate for gauging the true extent of the algorithm's performance.

Another reasons against the suitability of classical performance measures such as receiver operating characteristic (ROC) curves is the scarcity of available test data. Probability of detection and false positives are inherently statistical measures of performance that require a large sample size to produce meaningful results. Since the focus of this work is exclusively on practical applications of the proposed algorithm, the number of test vectors available are limited and each test signal (A, B and C) contains only 1 or 2 anomalous events. So instead of attempting to extract questionable probability measures from the limited data set, KLDR is utilised to quantify the observed performance.

When KLD is used as the measure of divergence, the KLD ratio (KLDR) is defined as

$$\mathrm{KLDR} = \frac{\mathrm{KLD}_{\mathrm{anom}}}{\mathrm{KLD}_{\mathrm{bg}}} \tag{8}$$

where $\mathrm{KLD}_{\mathrm{anom}}$ is the maximum of the detection peak and $\mathrm{KLD}_{\mathrm{bg}}$ is the maximum of the background baseline noise level. KLDR is the metric that is used to quantify the algorithm's performance.

For the ICA based algorithm, the circumstances are different. The test set (signal D) contains a sufficient amount of nominal and anomalous events to allow the use of more traditional performance metrics. Performance is measured in terms of the detector true positive rate (R_{tp}) and false discovery rate (R_{fd}). R_{tp} is defined as the ratio of the number of correctly detected anomalous events (Σ_{tp}) to the total number of anomalous events present (Σ_{ta}):

$$R_{\mathrm{tp}} = \frac{\Sigma_{\mathrm{tp}}}{\Sigma_{\mathrm{ta}}} = 1 - \frac{\Sigma_{\mathrm{md}}}{\Sigma_{\mathrm{ta}}} \tag{9}$$

where Σ_{md} is the number of anomalous events that missed detection. R_{fd} is the ratio of false positives (Σ_{fp}) to the total number of anomalies detected (includes both Σ_{fp} and Σ_{tp}) [29]:

$$R_{\mathrm{fd}} = \frac{\Sigma_{\mathrm{fp}}}{\Sigma_{\mathrm{fp}} + \Sigma_{\mathrm{tp}}}. \tag{10}$$

R_{fd} is preferred over the more common false positive rate (R_{fp}) as it is more useful in this context. R_{fp} is defined as the ratio between Σ_{fp} and all non-anomalous events (Σ_{tn}) in the signal:

$$R_{\mathrm{fp}} = \frac{\Sigma_{\mathrm{fp}}}{\Sigma_{\mathrm{tn}}}. \tag{11}$$

It is also known as the false alarm rate. Since Σ_{tn} is a very large number, R_{fp} is close to zero for most parameter combinations and therefore does not adequately reflect the variations observed in detector performance.

To summarise, KLDR is used to evaluate the performance of the KLD based algorithm while R_{tp} and R_{fd} are used to evaluate the ICA based algorithm.

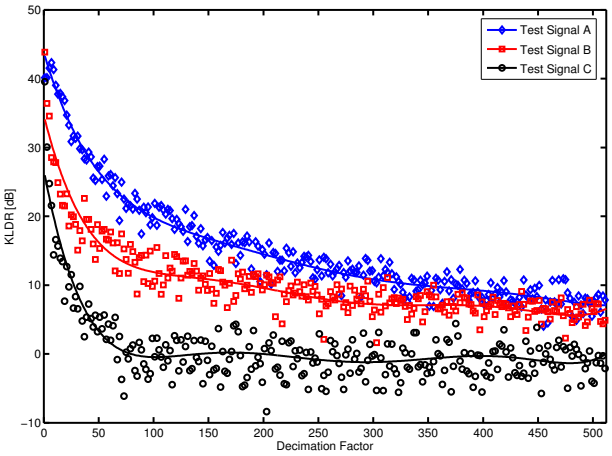


Fig. 9. Performance of the KLD based anomaly detection scheme under various data sampling rates. Decimation factor refers to the amount by which the input signal is under-sampled relative to the signal bandwidth. A factor of unity corresponds to a sampling frequency equal to the signal bandwidth. For two of the test cases (A and B), a KLDR well above 10 dB can be maintained for a window length of 256 μs, histogram bin count of 32, and a decimation factor of 100. Signal C is unable to accommodate such high decimation rates.

1) Sampling Rate: Continuous processes such as time-series must be sampled before the anomaly detection algorithms can be applied. The sampling frequency employed is crucial as it dictates the size of the input data sets, $|P_n|$, and therefore the memory utilisation of the KLD based algorithm – as indicated in Table II. For a given window length, a higher frequency implies that more data samples have to be stored and sorted to construct the histograms. If the frequency is too low, small scale signal features and anomalies may be lost. According to the Nyquist sampling criterion, a signal must be sampled with a frequency at least twice as large as its bandwidth to be reconstructible. For wideband signals this leads to a very high sampling frequency and hence a prohibitively large volume of data – heavily increasing the resource requirements of the proposed scheme. Since neither of the proposed algorithms require the time-series to be reconstructible, a far lower sampling frequency can be used instead. Fig. 9 shows how the performance of the KLD based anomaly detection scheme is affected by under-sampling of the input time series. The window length utilised is 256 μs and the histograms used to construct the PMFs are 32 bins wide. The amount by which the input time-series is under-sampled relative to the bandwidth is defined as the decimation factor. Therefore, a factor of unity implies that the signal is sampled at the same frequency as the signal bandwidth.

The results indicate that decimation factors as large as 500 can be successfully employed depending on the type and duration of the anomaly present. For test signals A and B, a KLDR of more than 10 dB can be maintained even with a decimation factor of 100. This is an important result as it indicates that satisfactory performance levels can be maintained with little input data and hence memory-limited implementations of the algorithm. At high decimation factors, performance is poor for test signal C. This is because the first

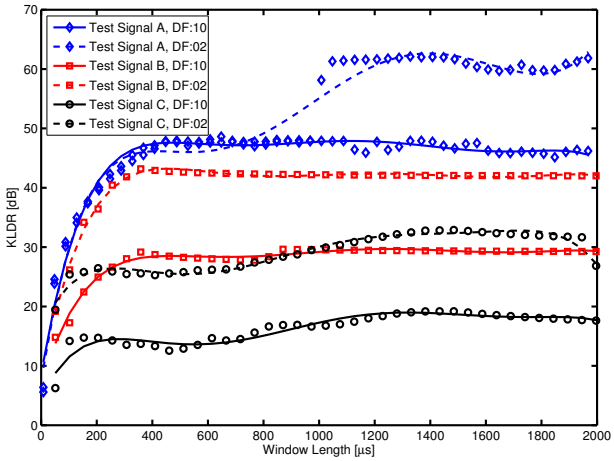


Fig. 10. Performance of the KLD based anomaly detection scheme under various window lengths and hence the input sample size $|P_n|$. Decimation factors (DF) of 2 and 10 are utilised and the number of histogram bins is 32. In all cases, an increase in the amount of data leads to a better performance. However, beyond a certain window length, the performance is no longer strongly affected.

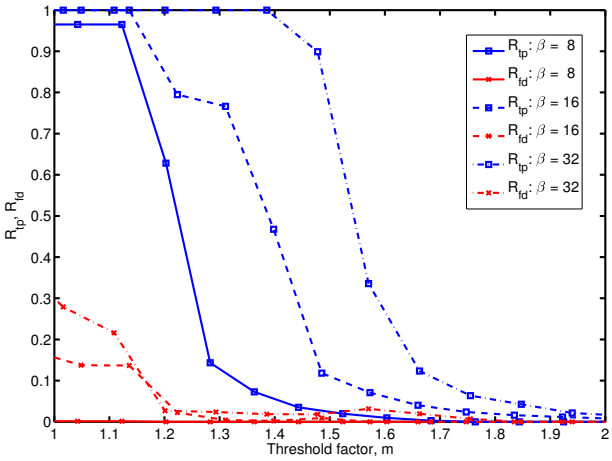


Fig. 11. Effect of information threshold (7) on detector R_{tp} and R_{fd} for the ICA algorithm. The cluster length utilised is 3. Histogram bin sizes of 8, 16, and 32 are used for comparison. R_{tp} for each β shows a sharp decrease when I_{th} (i.e. $m \cdot \sigma_{\mathrm{I}\langle\beta\rangle}$) is increased above a certain limit. Majority of the anomalous events have an information content less than this and miss detection. $m = 1.35$ and $\beta = 32$ yields the best performance.

anomaly (extra command sequence) is temporally brief and is likely to be lost when the signal is heavily under-sampled. As for the second anomaly in the signal, the change in power is simply not large enough to produce a significant increase in the divergence.

Results for the ICA based method are not shown since the input sampling rate has no bearing on the complexity or memory requirements of the algorithm – as shown in Table IV. The only requirement then on the sampling rate is that it must be fast enough to capture events that are suspected of being anomalous .

2) Window Length: The window size, and hence the input data set size $|P_n|$, is another parameter that is relevant for the KLD based algorithm but not the ICA based algorithm. The effect of the PMF estimation window size on the performance of the algorithm shown in Fig. 10. The number of histogram bins utilised is 32. Results are shown for under-sampling factors of 2 and 10. At lower decimation factors, more data is available and the KLDR improves uniformly across all window sizes for signals A and B. At smaller window sizes, performance for signal A is unaffected by the choice of the decimation factor due to the relatively long duration of the anomaly. This is because even at a decimation factor of 10, a sufficient number of anomalous samples are represented in the PMF.

As anticipated, the performance is poor at small window sizes where the amount of data available is insufficient to adequately model the underlying PMFs. Increasing the window length leads to an improvement of the performance. However, for signals B and C, the gains become marginal for windows larger than approximately 400 μs. The transition shown by signal A at a window length of 1 ms for a decimation factor of 2 is due to a sudden reduction in the KLD noise at the frame edges (as seen in the zoomed-in segment of Fig. 5) while the detection peak remains at approximately the same level. It is no coincidence that the duration of the inter-frame spacing for the signal is also 1 ms. It is a signal feature that is detected by the algorithm alongside the actual anomalous events. when the window size is increased beyond this feature size, it can no longer be resolved effectively by the detection algorithm and lead to a decrease in the noise level seen at the frame edges.

The initial KLDR improvements with increasing window size is due to improvements in the PMF estimates which in turn lead to a reduction in the baseline KLD levels. At larger window sizes, the anomalous samples represent smaller fractions of the data and hence contribute less to the shape of the estimated PMF – resulting in a decrease of the KLD due to the anomaly. As the background levels are also reduced by an increase in the data size, the overall KLD ratio (i.e. the KLDR) remains relatively constant.

3) Information Threshold: As stated previously in Section III, the discrimination threshold (7) is an important aspect of any detector. The impact of I_{th} on R_{tp} and R_{fd} of the ICA based algorithm is investigated using a cluster length of 3 and histogram bin sizes of 8, 16, and 32. The result of the analysis is shown in Fig. 11.

The plot shows that there is a hard I_{th} boundary for each β after which R_{tp} drops rapidly. This implies that the majority of the anomalous events share similar characteristics and convey information equivalent to that boundary. When I_{th} is increased further through the use of a larger threshold factor m, R_{tp} approaches zero due to an ever increasing number of missed detections.

At low information content thresholds, R_{fd} is also high – specially for high values of β. As explained earlier in Section III-A, a higher resolution makes the detector more susceptible to noise, leading to an increase in the number of false positives and hence the R_{fd}.

The impact of D_{th} on the performance of the KLD based

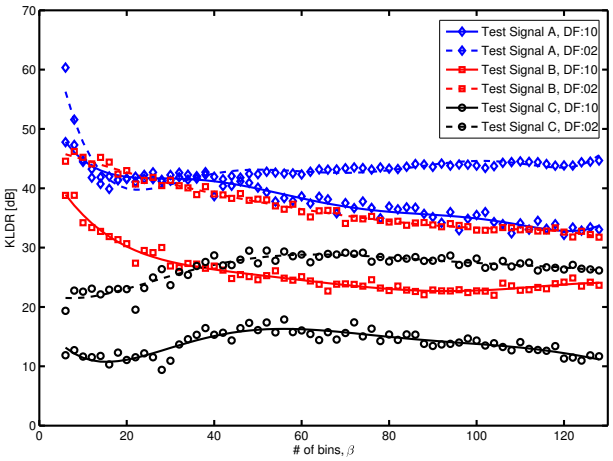


Fig. 12. Performance of the KLD based anomaly detection scheme under various histogram bin counts. The window size is set at 256 μs and the decimation factors used are 2 and 10. Generally, a larger number of bins lead to poorer performance due to increased noise in the estimates. However, the rate of change is small and therefore the drop in performance is insignificant over a wide range of bin resolutions.

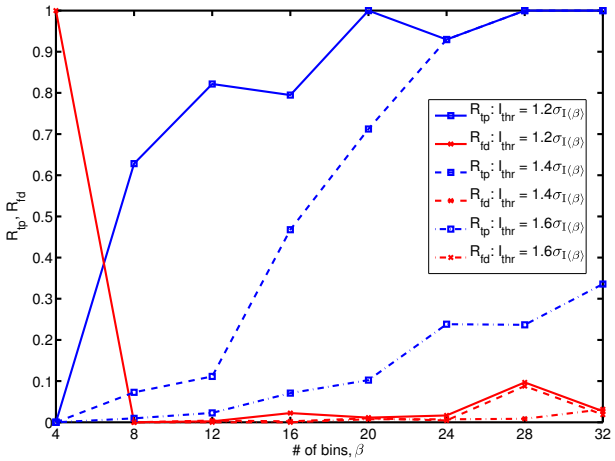


Fig. 13. Effect of histogram resolution on detector $R_{\rm tp}$ and $R_{\rm fd}$. The cluster length is 3 and thresholds are $1.2\sigma_{\mathrm{I}\langle\beta\rangle}$, $1.4\sigma_{\mathrm{I}\langle\beta\rangle}$ and $1.6\sigma_{\mathrm{I}\langle\beta\rangle}$. $R_{\rm tp}$ improves with resolution while $R_{\rm fd}$ deteriorates. $\beta = 20$ with a threshold of $1.2\sigma_{\mathrm{I}\langle 20\rangle}$ yields the best performance.

method has not been investigated and therefore cannot be shown. The reasons for this are:

- The number of anomalous events available is insufficient to investigate statistical trends.
- Detection is often guaranteed for a wide range of thresholds due to the large KLDR (greater than 30 dB) values that are observed.

It is the second reason that generally makes it straightforward to choose a suitable $D_{\rm th}$.

A mathematical treatment of the impact of $D_{\rm th}$ and $I_{\rm th}$ on detector performance is beyond the scope of this article. Such a framework requires well defined theoretical models of the data distribution which are difficult to obtain for real data vectors. Equations derived using the simplifying assumption that the distributions belong to a well-known class such as Gaussian would be of little use in context of the test signals used in this paper. Since the signals do not conform to any standard probability density function, it is out of necessity that the thresholds are determined empirically.

4) Histogram Resolution: Histogram bin resolution, represented by the parameter β, is of relevance to both of the proposed algorithms. Fig. 12 shows how performance of the KLD based algorithm is affected by the choice of the number of histogram bins used to classify the input data and estimate the PMFs. The window length is set at 256 μs and results are shown for decimation factors of 2 and 10. Once again, the smaller decimation factor provides uniformly improved performance over the entire range of β values. The only exception is signal A where the performance for smaller β values appear to be independent of the decimation factor used. The reasons for this is the relatively long duration of the anomaly – as explained previously in Section IV-B2.

The only trend common to all three signals is that the performance changes little with increasing bin numbers, with signal C showing an optimum in the vicinity of $\beta = 55$. This indicates that the behaviour observed is specific to the type of anomaly present in a signal. While the number of bins utilised does not appear to have a significant impact on the performance of the scheme for a fixed amount of data, the decrease observed is due to noisier PMF estimates that are obtained for larger values of β. Noisy PMFs lead to larger background KLD values and hence a reduced KLDR.

Fig. 13 shows how the $R_{\rm tp}$ and $R_{\rm fd}$ varies for the test signal (Fig. 8a) with the number of histogram bins utilised. The cluster length utilised is 3 and $I_{\rm th}$ of $1.2\sigma_{\mathrm{I}\langle\beta\rangle}$, $1.4\sigma_{\mathrm{I}\langle\beta\rangle}$ and $1.6\sigma_{\mathrm{I}\langle\beta\rangle}$ are used for comparison. The plot reveals that when $\beta = 4$, $R_{\rm tp}$ is zero and $R_{\rm fd}$ is unity for all thresholds tested. This is because the sensitivity is very low and no anomalies can be detected ($R_{\rm tp}$ = 0). Events exceeding the threshold are low-probability signal events and hence are all false positives ($R_{\rm fd}$ = 1). As β is doubled to 8, the resolution improves and there is a corresponding increase in the $R_{\rm tp}$. The $R_{\rm fd}$ also drops to a negligibly small value. As β is increased further, the $R_{\rm tp}$ increases due to better detector resolution. The $R_{\rm tp}$ improvements come at a cost however – the detector is more susceptible to noise at higher resolutions. This is evident from the gradual increase in the $R_{\rm fd}$.

Comparison between the three detection thresholds reveals that a higher $R_{\rm tp}$ is achieved with a lower threshold. Unfortunately, this also leads to a higher $R_{\rm fd}$. This behaviour is in accordance with the explanation provided in Section IV-B3.

5) Cluster Size: In order to investigate the impact of the information cluster length, N, on the ICA based detector, β is set at 16 and the analysis is performed for $I_{\rm th}$ of $1.2\sigma_{\mathrm{I}\langle 16\rangle}$, $1.4\sigma_{\mathrm{I}\langle 16\rangle}$ and $1.6\sigma_{\mathrm{I}\langle 16\rangle}$ on the test signal shown in Fig. 8. The result of the analysis is shown in Fig 14.

The significance of clustered anomaly detection is immediately obvious. With $N = 0$, when clustering is not performed, there is an overwhelming number of false positives. This is indicated by the high $R_{\rm fd}$. As soon as clustering is applied by setting $N = 2$, a dramatic drop in the $R_{\rm fd}$ is observed

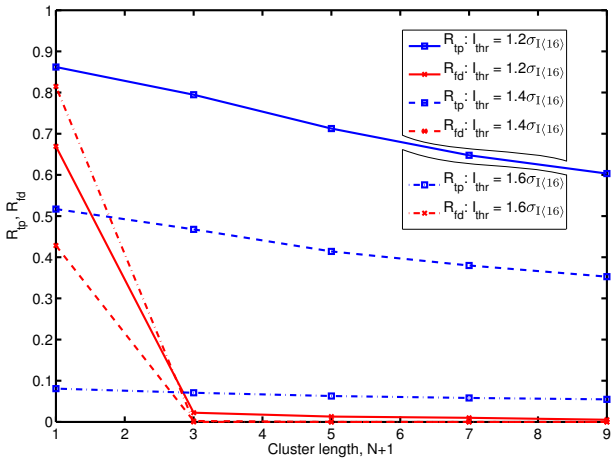


Fig. 14. Impact of cluster length on detector R_{tp} and R_{fd}. The number of histogram bins used is 16 and the thresholds utilised are $1.2\sigma_{\text{I}\langle 16\rangle}$, $1.4\sigma_{\text{I}\langle 16\rangle}$ and $1.6\sigma_{\text{I}\langle 16\rangle}$. Even the smallest cluster length ($N = 2$) is shown to provide a significant improvement in the R_{fd}.

– showing that even minimal anomaly clustering is sufficient to yield a massive improvement in detector performance. By lowering the R_{fd}, clustering also allows a lower I_{th} to be used to achieve a higher R_{tp}.

The impact of anomaly clustering on R_{tp} for a given I_{th} is relatively low. As cluster size is increased, a gradual decrease is observed in the R_{tp}. This is expected since larger cluster sizes lead to missed detections around the edges of the interference patterns. The plot also shows that higher thresholds lead to lower R_{tp} for a given cluster size. This is also expected since a higher information content threshold leads to a higher number of missed detections.

From the analysis performed on the test signals, it is clear that it is challenging to determine a set of parameters that are inherently optimal for the anomaly detection algorithms proposed. This is due to the fact that the optimal parameter set depends on a number of problems specific factors such as duration of the anomaly, dynamic range of the signal, etc. It may be possible to develop adaptive variants of the algorithms that automatically find the best parameter combinations subject to some performance criterion but that is beyond the scope of this paper.

The ICA based algorithm is particularly sensitive to the parameters utilised. Generally, it is seen that parameter values that increase the R_{tp} (good) often also lead to an increase in the R_{fd} (bad) and vice versa. Trade-offs must therefore be made to meet the required detector performance characteristic (low R_{fd}, moderate R_{tp} or high R_{tp}, moderate R_{fd}). A moderate number of bins ($\beta = 20$), small cluster size ($N = 2$), and a threshold of $1.2\sigma_{\text{I}\langle 20\rangle}$ bits ($m = 1.2$) provides a good balance between R_{tp} (1.0) and R_{fd} (0.01) for this particular test vector (signal D).

The KLD based algorithm on the other hand is much more robust with respect to the parameter combinations utilised. The results clearly show that performance better than 30 dB of KLDR can be easily obtained with reasonable choice of parameter values such as a window length of 256 μs, $\beta = 32$ and decimation factor of 2.

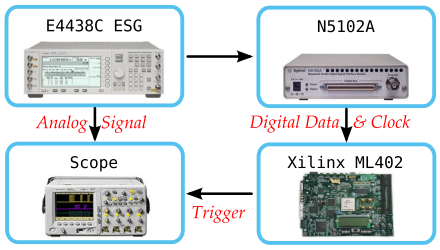


Fig. 15. Block diagram of the hardware test-bed. The E4438C ESG signal generator produces the signal under test in both analogue and digital format. The digital signal is passed to the FPGA platform via the N5102A digital signal interface module while the analogue signal is fed into the oscilloscope. The trigger signal from the FPGA core is also connected to the oscilloscope via a digital probe for comparison.

V. Hardware Platform

The FP-DKLD version of the detection algorithm presented in Section III-C has been implemented on a Xilinx Virtex-4 ML402 SX XtremeDSP Evaluation Platform to serve as a proof of concept and allow the testing of signals in real-time. To facilitate and accelerate code development, Xilinx SystemGenerator 10.1 is used in conjunction with MATLAB R2007a for the primary design flow. The implemented design runs at a clock speed of 80 MHz and is capable of processing input with a 10 MHz sample rate. The hardware chain used to test and validate the FP-DKLD implementation is shown in Fig. 15.

The Agilent E4438C ESG signal generator simultaneously provides analogue and digital versions of the signal under test. The digital data stream is connected to the FPGA platform via the Agilent N5102A Digital Signal Interface Module (DSIM) while the analogue signal is connected to a oscilloscope for display. The DSIM conditions the data (word size, bit alignment, clock relationship settings) and provides a synchronous clock signal that is used to drive the FPGA core. The trigger output from the FPGA platform is also connected to the oscilloscope via a digital probe so that it can be directly compared against the signal under test.

A pair of Wireless Broadband (WiBro) signals known to contain a number of different anomalous data segments are used to test the hardware platform. The design is configured with $|P_n| = 4096$ (320 μs) and $\beta = 8$. It is not necessary to down-sample the input data stream since the implemented design is capable of processing the input at its original rate.

The DSIM module provides the samples to the FPGA as 12-bit words in 2s complement format. The sample and DSIM clocks are set at 10 MHz and 40 MHz respectively – providing 4 clock cycles per input sample (CCPS). Although the design requires 8 CCPS, the DSIM is only capable of providing a maximum of 4 CCPS. To obtain the required 8 CCPS, the clock signal is doubled on the FPGA using an on-chip digital clock manager (DCM) module. Use of a DCM also has added benefit of providing clock buffering and de-skewing.

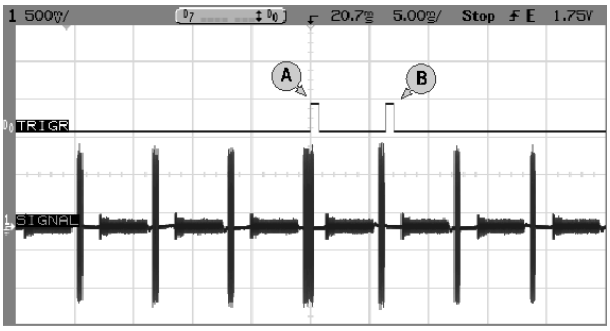


Fig. 16. Oscilloscope trace of a WiBro signal with a single anomalous frame and the associated trigger events. The KLD threshold is 0.0313. Trigger event A marks the start of the unusual segment of the anomalous frame. A second trigger event, B, is also obtained in the subsequent frame due to the disappearance of the anomalous feature.

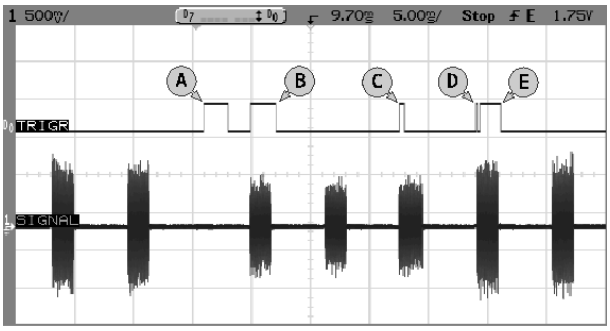


Fig. 17. Oscilloscope trace of a WiBro signal with multiple anomalous events and the associated trigger events. The KLD threshold is 0.0625. Triggers A and B are caused by a momentary change in the signal period. Trigger C flags the presence of a very brief command sequence that leads to a signal power level change. Trigger D marks the position in the subsequent frame where the power change inducing command was previously present and finally, trigger E corresponds to a sudden change in the overall signal power level.

A. Test Signal I

Fig. 16 shows the result of analysing the first WiBro signal using the FP-DKLD implementation of the algorithm. The signal analysed is identical to that shown in Fig. 6 and analysed in Section IV-A2. It is clear from the oscilloscope trace that one of the UL frames is longer than the others and hence is anomalous. With $D_{\text{th}} = 0.0313$, the FPGA implementation of the algorithm clearly succeeds in detecting the signal anomaly. The first trigger event obtained (A) coincides exactly with the anomalous segment of the unusual UL frame. A second trigger event (B) is observed when the UL frame structure subsequently returns to normal and the anomalous segment is no longer present.

B. Test Signal II

The second WiBro signal tested is shown in Fig. 17. It is identical to that shown in Fig. 7 and analysed in Section IV-A3 with the exception of an additional change in the timing structure. Analysing the signal with $D_{\text{th}} = 0.0625$ is seen to produce five trigger events – corresponding to the three anomalous conditions known to be present in the signal.

Trigger events A and B are due to a momentary disruption in the natural frame period of the signal. The first event marks the position where the UL frame should have been but is not while the second event marks the opposite: finding a UL frame where there should be none.

Events C and D are caused by a very brief command sequence at the beginning of the fifth UL frame that causes the power control loop to be initiated – which is then responsible for event E. Although invisible to the naked eye, the algorithm succeeds in locating the anomalous command sequence as clearly demonstrated by trigger event C. Since that command sequence is no longer present in the subsequent UL frame, its disappearance is marked by trigger event D.

Once initiated, the power control loop causes a sudden increase in the signal power level. This behaviour can be considered to be anomalous and is flagged by trigger event E. There are no other events associated with the change in power level as the signal power is seen to remain high beyond this point.

VI. Summary & Conclusions

Two complementary anomaly detection algorithms utilising information theoretic measures have been presented. Both algorithms are simple to implement and require little *a priori* information regarding the signal under test. Demodulation of the signal is also not required since the algorithms are capable of processing the baseband signal envelope itself in real-time. The information content analysis based method is capable of detecting singular anomalous events while the Kullback-Leibler divergence based method is also able to detect otherwise nominal events that are anomalous purely due to context (e.g. misaligned signal frames). In order to provide this context aware detection of anomalies, the KLD based algorithm requires the input signal to be periodic.

Analysis of a number of test signals captured over the air show that the KLD based scheme is successful at detecting all anomalies known to be present. Extensive tests using a software implementation of the algorithm demonstrates that it is robust with respect to parameter choices since satisfactory performance can be maintained with reasonable parameter values even when the input is severely under-sampled. With PMF estimation window sizes of 256 μs, 32 histogram bins and factor of 10 under-sampling, KLDR of 25 dB or better can be achieved depending on the anomaly present.

Although the primary purpose of the KLD based algorithm is to act as an anomaly detector, it can also be used to detect frame boundaries in a signal. The modification required is trivial: eliminate the spacing that normally separates the two PMF estimation windows. Since frame boundary detection is expected to reveal the underlying cyclic structure of a periodic signal, it may be used as a precursor to the actual anomaly detection algorithm to automatically learn the period of the signal – thus eliminating the need for any *a priori* information regarding a test vector.

A variation on the anomaly clustering technique presented in context of the ICA based algorithm may also be applied to the KLD based algorithm to further improve detection of

anomalous events. Anomalies generally lead to KLD peaks that increase monotonically until some maximum divergence is reached. It may be possible to exploit this observation to improve detection under low SNR conditions by restricting detection to signal segments that lead to monotonically increasing KLD values that are also above some predefined KLD threshold.

Both boundary detection and monotonic sequence detection are techniques that add significantly to the KLD based anomaly detection algorithm. Therefore, they will be the primary focus of work done in the future on this subject.

In addition to the MATLAB based software, the algorithm has been implemented on a Xilinx Virtex4 FPGA based hardware platform for evaluation under real world physical conditions. The design is highly efficient and capable of processing 10 MHz input signals without requiring any undersampling. Successful tests with a set of WiBro signals indicate the algorithm is indeed capable of processing high speed test vectors in real-time.

Unlike the KLD based method, the algorithm utilising ICA for anomaly detection does not require the input signal to be periodic. The only piece of information that is needed in advance is the set of reference event probabilities. A training data set known to be clean can be used to obtain the reference probabilities prior to analysis. The complexity and memory requirements of the algorithm are also very low.

It is clear from tests carried out using a software implementation of the algorithm that performance of the system is strongly affected by the choice of parameters such as histogram resolution, threshold and cluster size. Impact of these parameters on the detector performance by means of the true positive rate and false discovery rate has been analysed and guidelines for appropriate values have been provided. It is shown that a true positive rate of 100% and false discovery rate of 1% – guaranteeing zero missed detections with very few false positives – is possible for the signal tested with a suitable set of parameter choices.

The ICA based algorithm presented in this paper utilises a histogram with infinite memory – i.e. it maintains a record of all samples analysed. Clearly this implies that the information content of anomalous events such as a interference drop over time if they happen with sufficient frequency. If such behaviour is undesirable, it is necessary to implement a windowed histogram. Along with tests against other types of anomalous signals, it will be the focus of further research on the ICA algorithm.

ACKNOWLEDGEMENT

We would like to thank Roy Macnaughton and Peter Cain for the provision of test data. We greatly acknowledge the financial support of this project from Agilent Technologies and the University Relations Program.

REFERENCES

[1] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[2] S. Haykin, D. J. Thomson, and J. H. Reed, "Spectrum Sensing for Cognitive Radio," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 849–877, Apr. 2009.

[3] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation Issues in Spectrum Sensing for Cognitive Radios," in *Proc. of the Asilomar Conference on Signals, Systems and Computers*, vol. 1. Pacific Grove, CA, USA: IEEE, Nov. 7–10, 2004, pp. 772–776.

[4] D. Cabric, A. Tkachenko, and R. W. Brodersen, "Spectrum sensing measurements of pilot, energy, and collaborative detection," in *Proc. IEEE Military Communications Conference (MILCOM)*, Washington, DC, USA, Oct. 23–25, 2006, pp. 1–7.

[5] M. Öner and F. Jondral, "Air Interface Recognition for a Software Radio System Exploiting Cyclostationarity," in *Proc. of the 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, vol. 3, Barcelona, Spain, Sep. 5–8, 2004, pp. 1947–1951.

[6] C. Krügel, T. Toth, and E. Kirda, "Service Specific Anomaly Detection for Network Intrusion Detection," in *Proc. of the 2002 ACM Symposium on Applied Computing*. Madrid, Spain: ACM, Mar. 10–14, 2002, pp. 201–208.

[7] J. Lin, E. Keogh, A. Fu, and H. V. Herle, "Approximations to Magic: Finding Unusual Medical Time Series," in *Proc. 18th IEEE Symposium on Computer-Based Medical Systems*, IEEE. Dublin, Ireland: IEEE, Jun. 23–24, 2005, pp. 329–334.

[8] R. J. Bolton and D. J. Hand, "Statistical Fraud Detection: A Review," *Statistical Science*, vol. 17, no. 3, pp. 235–249, Aug. 2002.

[9] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[10] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed., ser. Wiley Series in Telecommunications, D. L. Schilling, Ed. John Wiley & Sons, Sep. 1991.

[11] J. G. Proakis, *Digital Communications*, 4th ed., ser. McGraw-Hill Series in Electrical and Computer Engineering, S. W. Director, Ed. McGraw-Hill Higher Education, Dec. 2000.

[12] J. Robertson, E. W. Tallman, and C. H. Whiteman, "Forecasting Using Relative Entropy," Nov. 2002, FRB of Atlanta Working Paper No. 2002-22. [Online]. Available: http://ssrn.com/abstract=355460

[13] H. Nakahara and S.-I. Amari, "Information-Geometric Measure for Neural Spikes," *Neural Computation*, vol. 14, no. 10, pp. 2269–2316, Oct. 2002.

[14] D. D. Falconer, F. Adachi, and B. Gudmundson, "Time Division Multiple Access Methods for Wireless Personal Communications," *IEEE Communications Magazine*, vol. 33, no. 1, pp. 50–57, Jan. 1995.

[15] A. Patcha and J.-M. Park, "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends," *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, Aug. 2007.

[16] M. Markou and S. Singh, "Novelty Detection: A Review – Part 1: Statistical Approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, Dec. 2003.

[17] ——, "Novelty Detection: A Review – Part 2: Neural Network Based Approaches," *Signal Processing*, vol. 83, no. 12, pp. 2499–2521, Dec. 2003.

[18] A. Lazarevic, L. Ertöz, V. Kumar, A. Ozgur, and J. Srivastava, "A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection," in *Proc. of the 2003 SIAM International Conference on Data Mining*, San Francisco, CA, USA, May 1–3 2003, pp. 25–36.

[19] Z. A. Bakar, R. Mohemad, A. Ahmad, and M. M. Deris, "A Comparative Study for Outlier Detection Techniques in Data Mining," in *Proc. of the 2006 IEEE Conference on Cybernetics and Intelligent Systems*. Bangkok, Thailand: IEEE, Jun. 7–9 2006, pp. 1–6.

[20] S. Basu and M. Meckesheimer, "Automatic Outlier Detection for Time Series: An Application to Sensor Data," *Knowledge and Information Systems*, vol. 11, no. 2, pp. 137–154, Feb. 2007.

[21] M. Desforges, P. Jacob, and J. Cooper, "Applications of Probability Density Estimation to the Detection of Abnormal Conditions in Engineering," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 212, no. 8, pp. 687–703, Jan. 1998.

[22] D.-Y. Yeung and C. Chow, "Parzen-Window Network Intrusion Detectors," in *Proc. of the 16th International Conference on Pattern Recognition*, vol. 4. Quebec City, Canada: IEEE, Aug. 11–15 2002, pp. 385–388.

[23] W. Lee and D. Xiang, "Information-Theoretic Measures for Anomaly Detection," in *Proc. of the IEEE Symposium on Security and Privacy*. Los Alamitos, CA, USA: IEEE, May 14–16 2001, pp. 130–143.

[24] M. Basseville, "Distance Measures for Signal Processing and Pattern Recognition," *Signal Processing*, vol. 18, no. 4, pp. 349–369, Dec. 1989.

[25] H. Shimazaki and S. Shinomoto, "A Method for Selecting the Bin Size of a Time Histogram," *Neural Computation*, vol. 19, no. 6, pp. 1503–1527, Jun. 2007.

[26] R. Krichevsky and V. Trofimov, "The Performance of Universal Encoding," *IEEE Transactions on Information Theory*, vol. 27, no. 2, pp. 199–207, Mar. 1981.

[27] D. H. Johnson, C. M. Gruner, K. Baggerly, and C. Seshagiri, "Information-Theoretic Analysis of Neural Coding," *Journal of Computational Neuroscience*, vol. 10, no. 1, pp. 47–69, Jan. 2001.

[28] H. Haas and S. McLaughlin, Eds., *Next Generation Mobile Access Technologies: Implementing TDD*. Cambridge University Press, ISBN: 13:9780521826228, Jan. 2008.

[29] Y. Benjamini and Y. Hochberg, "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *Journal of the Royal Statistical Society - Series B (Methodological)*, vol. 57, no. 1, pp. 289–300, 1995. [Online]. Available: http://www.jstor.org/stable/2346101

Radio Frequency Signature Correlation Based Speed Estimation for Indoor Positioning

Mostafa Afgani, Sinan Sinanović, Karim Khashaba*, Harald Haas
Institute for Digital Communications, School of Engineering & Electronics, The University of Edinburgh, UK
*School of Engineering & Science, Jacobs University Bremen, Germany
Email: h.haas@ed.ac.uk

***Abstract*—Dead reckoning represents a class of methods for relative position estimation based on a previously determined absolute reference position. The estimate is formulated from a combination of the known speed, time and heading information with the known reference position. One of the main obstacles to effective positioning of pedestrians via dead reckoning is the lack of accurate speed estimation algorithms. Existing methods are either complex or provide results that are unsatisfactory at the low velocities associated with pedestrians. In contrast, the two algorithms proposed in this paper are relatively simple to implement and provide accurate results at low velocities. In the first algorithm, a one-dimensional and unidirectional two-antenna solution is described where the speed can be easily estimated from a knowledge of the fixed inter-antenna distance and the time it takes for the trailing antenna to sense the same channel conditions (radio frequency (RF) signature) previously observed at the leading antenna. Computer simulations show that, with typical estimation errors of less than 2.67% around average pedestrian speeds, the approach is indeed effective and accurate. A by-product of the algorithm is an environment specific spatial correlation function which is used in the second algorithm to provide even better estimates. With the improvements offered by the latter algorithm, relative errors of merely around 0.15% on average are achievable. This improvement in performance over the first algorithm comes at the cost of slightly higher computational complexity. When subsequently used for user displacement estimation, a relatively small error of 24.5cm is observed after a duration of 60s.**

***Index Terms*—speed estimation, pedestrian dead reckoning, spatial correlation, radio frequency signature matching**

I. Introduction

Pedestrian dead reckoning (PDR) is a popular choice for positioning and navigation in areas (e.g. indoors) where Global Positioning System (GPS) based solutions cannot be used. Two pieces of information are essential before a valid location estimation can be made: a reference point with known coordinates and the velocity (speed and heading) at sufficiently close and successive intervals. Given the needed information, displacement of the user from the reference location can be approximated and, hence, an estimate of the new location coordinates can be obtained.

Accurate heading information is readily available from sensors such as a ring laser gyro [1]; it is the speed estimate that has been difficult to obtain with a sufficient degree of accuracy. Step length estimation based devices ([2], [3]) can only be used by pedestrians and must be mounted directly on the person. Frequent updates of the absolute position are also required as errors tend to accumulate with every step. While methods based on the level crossing rate (LCR) of the received Rayleigh fading envelope ([4], [5]) typically provide good results for high speeds, the accuracy drops considerably at speeds associated with pedestrians and other low velocity entities. Continuous wavelet transform (CWT) can also be used to extract speed information from the aforementioned envelope with a good degree of accuracy [6], however such a procedure is computationally complex and expensive.

In this paper, two novel methods of speed estimation are described. The first proposed method, termed relative radio frequency signature matching (RRFSM), correlates the RF signatures at two antennae separated by a known distance to determine the time it takes for the trailing antenna to experience the same channel conditions as that previously experienced by the leading antenna. As the antenna separation is predefined and known (e.g. in a MIMO (multiple-input-multiple-output) device), the speed is easily calculated from an estimate of the time delay. As will be shown later, the new speed estimation approach also employs an adaptive algorithm that allows accurate estimates at both high and low velocities. Furthermore, it is of low computational complexity as the main operation only involves the calculation of correlation values between two channel estimates which are based on the known existing radio frequency transmission in the environment.

The second algorithm uses the first to compute a spatial correlation function. Then, based on this knowledge, the correlation of the channel data at two different locations can be used to estimate the speed. This refinement, while adding some complexity to the speed estimation, is capable of lowering the estimation errors.

The rest of this paper is organised as follows. In Section II the wireless channel model considered is presented and in Section III speed estimation algorithms are

This paper is based on "Speed Estimation Using Relative Radio Frequency Signature Matching," by M. Afgani, and H. Haas, which appeared in the Proceedings of the 66th IEEE Vehicular Technology Conference (VTC), Baltimore, USA, September/October 2007.

This research was conducted in cooperation with MobilTec GmbH as part of the Mobile Positioning System (MPos) project, funded by the BIS (Bremerhavener Gesellschaft für Innovationsförderung und Stadtentwicklung mbH) within the T.I.M.E. program (grant 56023/2 VN) of the State of Bremen/Bremerhaven, Germany.

described. Section IV provides details of the realisation and the simulation results. Section V concludes the paper.

II. Wireless Channel Model

The multipath effect is a common phenomenon in typical terrestrial environments. The RMS (root mean square) delay spread of the power delay profile determines the coherence bandwidth of the channel. In future broadband wireless communications, the coherence bandwidth is typically smaller than the total channel bandwidth which results in frequency selective fading. In addition to frequency selective fading, signal frequencies can also experience spreading (frequency dispersion) caused by Doppler shifts. For antenna arrays, the impulse response vector $\mathbf{h}(t)$ of a multipath channel can be represented by

$$\mathbf{h}(t) = \sum_{l=0}^{L(t)-1} A_l(t)\, e^{j\phi_l(t)} \mathbf{a}(\theta_l(t))\, \delta(t-\tau_l(t)) \qquad (1)$$

where $L(t)$ is the number of multipath components, A_l is the amplitude, ϕ_l is the carrier phase shift, θ_l is the angle of arrival (AoA) of the l^{th} multipath component, $\delta(\cdot)$ is the unit impulse function and τ_l is the time delay of the l^{th} multipath component. $\mathbf{a}(\theta_l(t))$ is known as the array response vector. When the signal and the antenna array (containing m antennae) at the receiver are restricted to a two-dimensional (2D) space, the array response vector is given by

$$\mathbf{a}(\theta_l(t)) = \begin{bmatrix} \exp(-j\Psi_{l,1}) \\ \exp(-j\Psi_{l,2}) \\ \exp(-j\Psi_{l,3}) \\ \cdots \\ \exp(-j\Psi_{l,m}) \end{bmatrix} \qquad (2)$$

where $\Psi_{l,i}(t) = [x_i \cos(\theta_l(t)) + y_i \sin(\theta_l(t))]\,\beta$. The spatial coordinate pair (x_i, y_i) represents the location of the antenna element i in 2D space. $\beta = \frac{2\pi}{\lambda}$ is the wavenumber [7], with λ being the carrier wavelength.

For line-of-sight (LoS) channels, the amplitude can be modeled by a Rician random variable, with the Rician parameter representing the relative strength of the LoS component. A Rayleigh distributed random variable is used instead when there is no single dominant multipath component. The phase shift is often assumed to be uniformly distributed within the interval $[0, 2\pi]$. The AoA is highly dependent on the relative geometry of the environment and the heights of the scatterers and the receiver. Although the assumption that the AoA is uniformly distributed within the interval $[0, 2\pi]$ is valid under certain circumstances, there are other models that are more suitable for a given channel model. An overview is provided in [7]. The delay associated with each multipath component is generally assumed to be exponentially distributed [8].

The maximum Doppler shift, $f_{d(\max)}$, experienced by a receiver is dependent on the speed, v, and is given by [9]:

$$f_{d(\max)} = \frac{v}{\lambda}. \qquad (3)$$

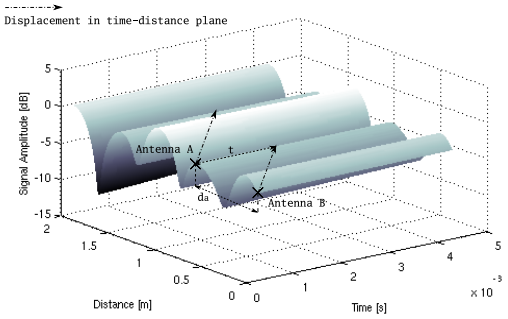


Fig. 1. Multipath channel in time and space simulated for a carrier frequency of 474MHz (DVB-T) and a speed of 1.5ms^{-1}. Assuming that the movement is restricted to the single direction parallel to the antenna array A-B, after t seconds *antenna B* is where *antenna A* was and experiences the same channel response that *antenna A* experienced t seconds ago. As the distance, d_a, between the antennae is known, the speed is easily estimated as $|v| = \frac{d_a}{t}$.

The presence of Doppler spread in a multipath channel causes it to display variations in time: the higher the Doppler frequency, the shorter is the coherence time [9],

$$T_c = \frac{0.423}{f_{d(\max)}}, \qquad (4)$$

of the channel. The terrestrial digital video broadcast (DVB-T) service is ubiquitous and wideband – allowing the frequency selective nature of the channel to be captured over a wide range of locations. Each channel is 8MHz wide and the first has a carrier frequency of 474MHz [10]. A DVB-T receiver travelling at a typical pedestrian velocity of approximately 1.5ms^{-1} experiences a rather insignificant maximum Doppler frequency of 1.6Hz and hence a relatively long channel coherence time of 0.26s. Fig. 1 shows the space-time characteristics of a multipath Rayleigh fading channel as experienced by such a receiver.

It is clear from the plot that although the channel stays relatively unchanged over the duration of the coherence time, it shows rapid variations in space. The spatial correlation, $\rho(d, \theta)$, between signal envelopes separated by a distance d, along some azimuthal direction θ is approximated by

$$\rho(d,\theta) \approx \exp\left[-23\Lambda^2 \left(1 + \gamma\cos\left[2\left(\theta - \theta_{\max}\right)\right]\right)\left(\frac{d}{\lambda}\right)^2\right] \qquad (5)$$

as derived in [11]. Λ is the angular spread defined as

$$\Lambda = \sqrt{1 - \frac{|F_1|^2}{F_0^2}}, \qquad (6)$$

γ is the angular constriction defined as

$$\gamma = \frac{|F_0 F_2 - F_1^2|}{F_0^2 - |F_1|^2}, \qquad (7)$$

$\theta_{\max}$ is the azimuthal direction of maximum fading defined as

$$\theta_{\max} = \frac{1}{2}\arg\left(F_0 F_2 - F_1^2\right), \qquad (8)$$

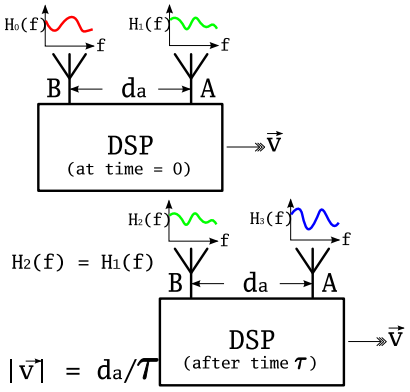


Fig. 2. Speed estimation using relative RF signature matching. The time τ required by the device to traverse the fixed distance d_a can be estimated by correlating the channel responses obtained at the two antennae. From an estimate of τ, the speed $|v|$ is easily calculated using $|v| = \frac{d_a}{\tau}$.

and F_n is the n^{th} complex Fourier coefficient of the angular distribution of multipath power (ADP), $p(\theta)$:

$$F_n = \int_0^{2\pi} p(\theta_x) \exp(jn\theta_x)\ d\theta_x. \tag{9}$$

As stated earlier, temporal variations in the channel are quantified by a measure of the coherence time, T_c. A similar measure of the spatial channel variations can be developed using (5). If the coherence distance, D_c, is defined as the distance at which the spatial correlation coefficient drops to 0.5, i.e. $\rho(D_c) = 0.5$, it can be approximated by [11]

$$D_c \approx \frac{\lambda\sqrt{\ln 2}}{\Lambda\sqrt{23\,(1+\gamma\cos[2\,(\theta-\theta_{\max})])}}. \tag{10}$$

For an omnidirectional Rayleigh fading channel, the coherence distance is often approximated by [11]

$$D_c \approx \frac{9\lambda}{16\pi}. \tag{11}$$

Applying this equation to the DVB-T system mentioned earlier produces a coherence distance of approximately 11cm. This is in agreement with the spatial variations observed in Fig. 1.

III. Speed Estimation Algorithms

The spatial and temporal correlation properties of a wireless channel makes it ideal for exploitation in mobile speed estimation. Given a source of RF (radio frequency) signal and a receiver equipped with multiple antennae, relative RF signature matching can be used to estimate the speed of the receiver unit. The following sections describe the algorithms that utilise the fading characteristics of the received RF signal to provide an estimate of the speed.

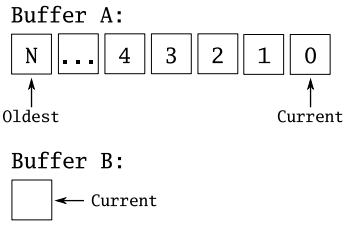


Fig. 3. Structure for the database used to store the channel response estimates.

A. Method I: Relative RF Signature Matching (RRFSM)

It is possible to estimate the speed of a MIMO mobile wireless device by comparing the channel response observed at each of the antenna array elements. The simplest such device consists of a receiver with two antennae as illustrated in Fig. 2. Motion is assumed to be restricted to a single spatial dimension and the antenna array is aligned parallel to the direction of motion. This results in a simplification of $\Psi_{l,i}(t)$ in (2) to $[x_i \cos(\theta_l(t))]\,\beta$.

In operation, the speed estimator traces a diagonal path in the spatio-temporal plane as illustrated in Fig. 1. Labelling the antenna array elements by the letters "A" and "B" respectively, the starting position, $(A, B)_{t_0}$, of the array elements can be denoted by the spatio-temporal coordinate pair $[(x_1, t_0), (x_0, t_0)]$. The radio channel sensed at the array elements, $(H_A, H_B)_{t_0}$, can be similarly represented by $[(H_1, t_0), (H_0, t_0)]$.

Assuming a non-zero velocity, the coordinates are $[(x_3, t_0+\tau), (x_2, t_0+\tau)]$ and the channel responses are $[(H_3, t_0+\tau), (H_2, t_0+\tau)]$ after some time τ. For some value of $\tau = \tilde{\tau}$, $x_2 = x_1$ and $H_2 \approx H_1$ due to the relatively large coherence time of the channel. This implies that the device travels a distance $d_a = x_1 - x_0$ in that time. Since d_a is the inter-antenna separation distance, it is constant and known *a priori*. Therefore, the speed of the device is easily estimated from an estimate of $\tilde{\tau}$ using

$$|\overrightarrow{v}| = \frac{d_a}{\tilde{\tau}}. \tag{12}$$

It is possible to obtain an estimate of the time delay, $\tilde{\tau}$ by comparing the channel responses observed at the antenna array elements. Accurate channel estimates or knowledge of the underlying fading process is not required as the proposed method only looks for a relative match between channel responses. This also implies that a reasonably low amount of uncertainty and errors in the estimates can be safely ignored as long as all the estimates are affected in a similar fashion.

Clearly, a history of the channel responses observed at the "leading" antenna (A in this case) must be maintained. Fig. 3 illustrates the structure of a database that stores the current and N previous channel response estimates (CREs) and associated timestamps for antenna A. The instantaneous CRE of antenna B is also stored for the duration of the signal correlation computation.

The operation used to find the the CRE leading to maximum correlation is:

$$\mathrm{idx} = \underset{k}{\operatorname{argmax}} \left(B \star A_k\right);\ k = 1, \ldots, N. \quad (13)$$

idx is the database index corresponding to the CRE from antenna element A yielding maximum correlation and can be used to determine the elapsed time from the associated timestamp. The binary operator "$\star$" represents cross-correlation at a lag of zero. It is unnecessary to compute $B \star A_0$ since it implies a time delay of zero and hence infinite speed.

The cross-correlation of two signals at a lag of zero is identical to their *dot-product*. As a result, (13) can be simplified to a series of normalised dot-product calculations followed by maximum detection:

$$\mathrm{idx} = \underset{k}{\operatorname{argmax}} \left(\frac{B \cdot A_k}{||B|| \cdot ||A_k||}\right);\ k = 1, \ldots, N. \quad (14)$$

The range of speeds detectable with such a device ultimately depends on the estimation rate and the CRE history buffer maintained. With $v_{\min}$ and $v_{\max}$ as bounds to the detectable speed range, the time delay between CRE estimates must be less than $\frac{d_a}{v_{\max}}$ for some inter-antenna separation distance d_a. On the other hand, correct estimation of $v_{\min}$ requires the ability to store CRE estimates spanning a time of at least $\frac{d_a}{v_{\min}}$. Assuming that the CRE estimation interval is short enough to detect $v_{\max}$, the total number of past CREs to store are:

$$N = \left\lceil \frac{\left(\frac{d_a}{v_{\min}}\right)}{\left(\frac{d_a}{v_{\max}}\right)} \right\rceil = \left\lceil \frac{v_{\max}}{v_{\min}} \right\rceil. \quad (15)$$

Since the time delay is estimated from a correlation between CREs, the instantaneous channel response must be estimated periodically. For a wireless system with symbol duration T_s, the CRE estimation interval $\hat{i}$ can be defined as the number of complete symbols that elapse between the CRE estimates:

$$\hat{i} = \left\lfloor \frac{\alpha}{\mu} \right\rfloor, \quad (16)$$

where α is a constant defined as

$$\alpha = \frac{d_a}{v_{\max} \times T_s} \quad (17)$$

and μ is an adaptive scaling parameter. By adjusting the CRE estimation interval in accordance with the estimated speed of the device, it attempts to provide higher accuracy in the time delay estimates.

The parameter μ is a real number bounded by the constant α and 1, allowing $\hat{i}$ to assume integer values within the range $\hat{i}_{\min} = 1$ and $\hat{i}_{\max} = \lfloor \alpha \rfloor$. A CRE estimation interval lower than $\hat{i}_{\min}$ is impossible as it implies an interval shorter than a symbol. An interval longer than $\hat{i}_{\max}$ is unnecessary as it would provide speed estimates lower than $v_{\min}$ for a given value of N.

Initially, the adaptive parameter μ is set to 1 so that $\hat{i} = \hat{i}_{\max}$. This allows for a rough estimate, v_{est}, to be obtained for any speed between $v_{\min}$ and $v_{\max}$. Given that estimate, linear interpolation is employed to obtain the new value of μ and hence $\hat{i}$:

$$\mu = \begin{cases} m \cdot (v_{\mathrm{est}} - v_{\min}) + 1, & v_{\min} \le v_{\mathrm{est}} \le (v_{\max} + \epsilon) \\ 1, & \text{otherwise} \end{cases} \quad (18)$$

where

$$m = \frac{\alpha - 1}{v_{\max} - v_{\min}} \quad (19)$$

and ϵ is a small number required to allow the adaptive procedure to converge at the upper bound of the detectable speed range. Defining the maximum speed detectable with $\mu = 1$ as

$$\tilde{v}_{\max} = \frac{d_a}{T_s \times \hat{i}_{\max}} \quad (20)$$

and $\Delta v_{\max}$ as the absolute difference between $\tilde{v}_{\max}$ and $v_{\max}$

$$\Delta v_{\max} = \left|\tilde{v}_{\max} - v_{\max}\right|, \quad (21)$$

ϵ is defined as

$$\Delta v_{\max} \le \epsilon \le 1.1 \Delta v_{\max}. \quad (22)$$

Since only a lower bound to ϵ is logically defined, an arbitrary upper bound of $1.1\Delta v_{\max}$ is introduced to provide a sensible limit to the possible choices of ϵ.

While the first case in (18) is useful for adapting the value of μ while the device is in motion, the second case is necessary to reset the value of μ to 1 in case of erroneous estimates (e.g. $v_{\mathrm{est}} \gg v_{\max}$) or when the device is stationary ($v_{\mathrm{est}} = 0$).

Due to the integer nature of $\hat{i}$, any time delay estimated is generally an integer multiple of T_s. As a result, the speeds estimated between $v_{\min}$ and $v_{\max}$ do not form a continuous range:

$$v_{\mathrm{est}} = \frac{d_a}{k \times T_s \times \hat{i}}. \quad (23)$$

k is an integer related to the idx in (14).

Example Let $v_{\min} = 0.1\mathrm{ms}^{-1}$, $v_{\max} = 15\mathrm{ms}^{-1}$, $T_s = 224\mu\mathrm{s}$, $\hat{i} = \hat{i}_{\max} = 14$, $d_a = 0.05\mathrm{m}$, and $N = 150$. The speeds detectable are then $15.94\mathrm{ms}^{-1}$, $7.97\mathrm{ms}^{-1}$, $5.31\mathrm{ms}^{-1}$, $\ldots$, $0.106\mathrm{ms}^{-1}$ corresponding to $k = 1, 2, 3, \ldots, N$ respectively.

The example cited also indicates that the performance is quite poor in the vicinity of $v_{\max}$ with $\hat{i} = \hat{i}_{\max}$. This is due to the fact that the CRE estimation interval is not short enough to provide the time resolution needed at higher speeds. Table I compares the result of using $\hat{i} = \hat{i}_{\max}$ and $\hat{i} = \hat{i}_{\min}$ for the example provided.

The tabulated data indicates that a longer estimation interval works well for speeds in the vicinity of $v_{\min}$ while higher speeds require shorter estimation intervals for more accurate estimates. The improved accuracy at in the vicinity of $v_{\max}$ comes at the cost of the minimum speed detectable – it increases to $1.4881\mathrm{ms}^{-1}$ when $\hat{i} = \hat{i}_{\min}$.

From (18) it is clear that the adaptive parameter μ always tries to provide the most suitable $\hat{i}$ for a given database size – higher values of v_{est} result in higher

TABLE I
THE EFFECT OF THE CRE ESTIMATION INTERVAL ON SPEEDS DETECTABLE.

$\hat{i} = \hat{i}_{\max} = 14$		$\hat{i} = \hat{i}_{\min} = 1$	
k	**Speed (m/s)**	**k**	**Speed (m/s)**
1	15.944	1	223.21
2	7.9719	...	...
3	5.3146	14	15.944
4	3.9860	15	14.881
...	...	16	13.951
148	0.10773	...	...
149	0.10701	149	1.4981
150	0.10629	150	1.4881

values of μ which in turn lead to a reduction in the CRE estimation interval. The opposite is true for lower values of v_{est}.

While it is obvious that the setting $\hat{i} = \hat{i}_{\min}$ yields the finest time resolution possible and hence the most accurate results, it is inefficient in practice as the number of past CRE estimates that must be stored in memory to detect $v_{\min}$ is prohibitively large ($N = 4465$ for a DVB-T receiver with $T_s = 224\mu$s, $d_a = 10$cm, $v_{\min} = 0.1\text{ms}^{-1}$). Dynamically adjusting $\hat{i}$ to the estimated speed allows the RRFSM algorithm to be more memory efficient ($N = 150$ for $v_{\min} = 0.1\text{ms}^{-1}$ and $v_{\max} = 15\text{ms}^{-1}$).

Table I also shows that with the initial value of $\hat{i} = 14$ and hence $\mu = 1$, a true speed of 15ms^{-1} is most likely to be detected as 15.944ms^{-1}. At the absence of the parameter ϵ, μ will always remain at that initial value (since the condition $v_{\text{est}} \leq (v_{\max} + \epsilon)$ in (18) would not be satisfied) and hence the speed will continue to be estimated as 15.944ms^{-1}. With $\epsilon = 1$, however, that condition will be satisfied and μ will be increased accordingly – reducing the estimation interval and yielding an estimate that is more precise than the last.

To avoid erroneous speed estimates, the maximum value produced by (14) must be above a sufficiently high threshold. If that value is below the threshold, the CREs at the antenna elements are assumed to be uncorrelated and hence the device is assumed to be at a standstill. It is important to pick a suitable threshold as a value too low may result in erroneous results while a value too high will reduce the sensitivity of the algorithm. It may be possible to determine a sensible threshold from spatial correlation models of the channel or real-world experiments.

A flow diagram of the one dimensional and unidirectional algorithm is shown in Fig. 4. The method can be extended for multi-dimensional speed estimation by intelligently combining the results from additional antenna arrays oriented parallel to each of the spatial dimensions desired.

B. Method II: Modified RRFSM

The algorithm described in Section III-A produces discrete speed estimates and as such is generally not expected to be exact. However, the method can be further

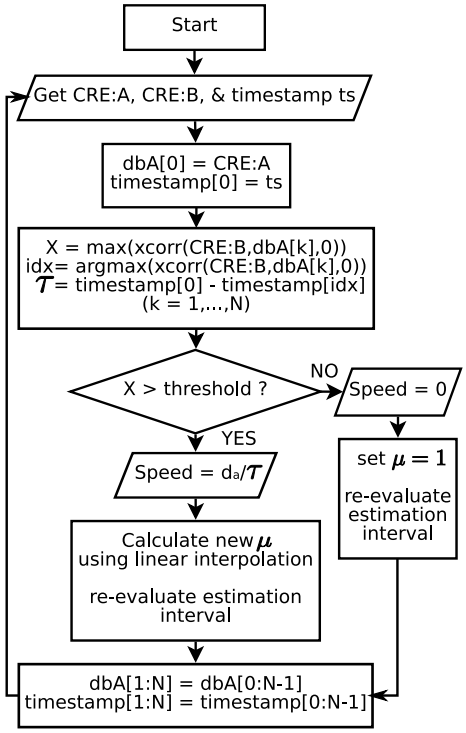


Fig. 4. 2-antenna speed estimation algorithm. "dbA[]" is the CRE buffer associated with antenna A. The function `xcorr(a,b,L)` provides the normalised cross correlation product at delay = L (i.e. it performs the operations necessary to compute the parenthesised expression in (14)).

extended by utilising the signal spatial correlation model described by (5).

Once again, assuming that the motion of the antenna array is restricted to a single constant direction θ_0, (5) can be simplified to:

$$\rho(d) \approx \exp\left[-23\Lambda^2\left(1+\gamma\cos\left[2\left(\theta_0-\theta_{\max}\right)\right]\right)\left(\frac{d}{\lambda}\right)^2\right]. \tag{24}$$

As Λ, γ, and $\theta_{\max}$ are also constants for a given ADP (see Section II), the equation can be further simplified to:

$$\rho(d) \approx \exp\left[-23K\left(\frac{d}{\lambda}\right)^2\right], \tag{25}$$

where the constant K is defined as

$$K = \Lambda^2\left(1+\gamma\cos\left[2\left(\theta_0-\theta_{\max}\right)\right]\right). \tag{26}$$

Therefore, given the inverse function of (25):

$$d \approx \lambda\sqrt{-\frac{\ln(\rho)}{23K}}, \tag{27}$$

it is possible to estimate the distance d between two signal envelopes given their correlation factor ρ, the signal wavelength λ, and the channel dependent constant K. If the time delay τ between the envelopes is known, an estimate of the speed can be obtained from the estimate

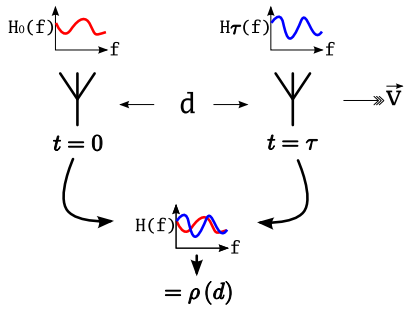


Fig. 5. Speed estimation using signal spatial correlation (modified RRFSM). Knowledge of the channel parameters and the correlation ρ between two signal fading envelopes sampled τ seconds apart allows the distance traversed to be estimated using (29). The speed, $|v|$, can then be estimated using $|v| = \frac{d}{\tau}$.

of d using (12) by substituting d for d_a. The concept is illustrated in Fig. 5. Assuming that the value of K is known, a single antenna can be used for speed estimation using (27).

Unfortunately, the parameter K is directly related to the ADP which is highly dependent on the geometry of the radio environment. Therefore it is virtually impossible to obtain an analytical expression for the instantaneous value of K. Hence it must be estimated before the speed estimation algorithm can be applied. Reordering of the variables in (27) provides an approximation

$$K \approx -\frac{\ln(\rho)}{23}\left(\frac{\lambda}{d}\right)^2 \tag{28}$$

in terms of the envelope correlation ρ and the distance d.

This is where the scheme described in Section III-A can be exploited. While evaluating (14), the CRE at antenna B is correlated against all the CREs in the database to yield a set of correlation factors $(\rho_1, \rho_2, \rho_3, \ldots, \rho_N)$. Then, using the rough speed estimate, v_{est}, each correlation factor ρ_k can be mapped to a distance d_k using

$$d_k = d_a - (t_k \cdot v_{\text{est}})\,;\; k = L, L+1, \ldots, M \tag{29}$$

where t_k is the time delay associated with ρ_k and L and M define a subset of the coefficients that meet some suitability criterion as explained later in this section. Given the set of correlation coefficients and their associated distance estimates, a set of estimates for K can be obtained using (28). The final estimate K' is then obtained by taking the arithmetic mean over that set:

$$K' = \frac{1}{M-L+1}\sum_{k=L}^{M} K'_k. \tag{30}$$

Since K' is an estimate of the true value of K, it is necessary to evaluate its accuracy. Given the procedure employed to obtain the estimate, the most likely source of errors are the correlation coefficients ρ_k and v_{est}. As v_{est} already represents the "best-effort" of the algorithm in Section III-A, an attempt is made to pick the most suitable set of correlation coefficients. The relative error

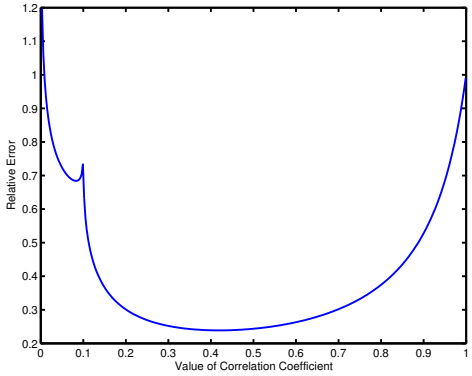


Fig. 6. Relative error in the estimate of K for a relative error of 10% in the correlation coefficients. The error is below 30% for coefficients between 0.2 and 0.7. The singularity observed at a correlation of 0.1 is due to the choice of a relative error of 10% or 0.1. This causes the logarithm in the numerator of (31) to approach infinity at that point.

e_K in the estimate of K for any correlation factor ρ' is given by:

$$e_K = \left|\frac{K - K'}{K}\right| = \left|\frac{\ln\left(\frac{\rho'}{\rho' - e_{\rho'}}\right)}{\ln(\rho' - e_{\rho'})}\right|, \tag{31}$$

where $e_{\rho'}$ is the relative error in the correlation coefficient ρ'. The derivation is provided in the Appendix A.

Since it is not possible to determine the relative error in each correlation coefficient computed, $e_{\rho'}$ is assumed to be the same for all coefficients. Fig. 6 plots (31) for $e_{\rho'} = 0.1$ (i.e. 10%). It can be seen that the lowest relative errors in K are obtained for coefficients that lie within the 0.2 to 0.7 range. The reason for this becomes clear from a plot of the spatial correlation functions for various channel conditions (Fig. 7). The figure shows that the rate of change of the distance with respect to the correlation coefficient is the least over the 0.2 to 0.7 range – implying that a large uncertainty in the value of ρ leads to relative little error in the value of d, minimising the relative error in K. Therefore, L and M in (29) are chosen such that only coefficients with values within a certain range $[\rho_{\min}, \rho_{\max}]$ are considered.

Once an estimate of K is available, subsequent correlation coefficients ρ_k estimated at intervals of Δt, where $\Delta t \ll T_c$, can be used to obtain the associated set of distances d_k using (27). The corresponding set of speed estimates can then be obtained as follows:

$$v_{\text{est},k} = \frac{d_k}{k \times \Delta t};\; k = L, L+1, \ldots, M. \tag{32}$$

The final estimate is obtained by taking the arithmetic mean of the set of estimated speeds:

$$v_{\text{est}} = \frac{1}{M-L+1}\sum_{k=L}^{M} v_{\text{est},k}. \tag{33}$$

For the algorithm to remain accurate and effective, the estimate of K must be periodically updated. Fig. 8 shows

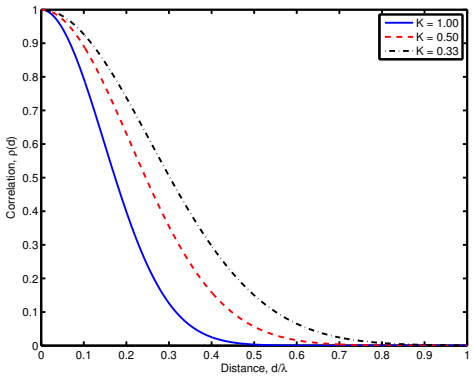


Fig. 7. Plot of the correlation function for various channel conditions. The gradient is steepest roughly between $\rho = 0.7$ and $\rho = 0.2$. Therefore, the rate of change of the distance with respect to the correlation coefficient is the lowest over that range. This implies that large errors in the value of ρ leads to relatively small errors in the estimate of d.

a flow chart of the algorithm.

IV. Simulation & Results

MATLAB simulations are used to evaluate the performance of the speed estimation algorithms. The speed profile utilised is designed to emulate the motion of a pedestrian as illustrated by Fig. 9. The duration of the profile is 60s.

A. Method I: RRFSM

For simulation purposes, the RF signal source is assumed to be a DVB-T transmitter with a carrier frequency of 474MHz and OFDM symbols consisting of 1705 subcarriers (2K mode of DVB-T). The multipath channel is modeled as Rayleigh fading [8] with component phase offsets (ϕ_l) and time delays (τ_l) obtained from [10]. For simplicity, the AoAs (θ_l) are assumed to be time-invariant and uniformly distributed in $[0, 2\pi]$. Furthermore, it is also assumed that the channel response estimates are error free and displacement is in one dimension only. The correlation threshold is maintained at 0.95. The use of such a high threshold is justified since the CRE estimates are assumed to be perfectly known and error-free.

Fig. 10 shows the performance of the estimation algorithm at $d_a = 0.10\text{m}$ and 0.30m. A moving average filter with a memory of 4ms is applied to the estimates for a smoother output. With $d_a = 0.10\text{m}$, a good match between the real and estimated speeds is obtained while a larger d_a shows relatively poorer performance. The reason for this becomes clear once the coherence time of the channel is taken into account. At a separation distance of 0.30m, the time required by antenna B to traverse the inter-antenna separation distance is 0.6s for a constant speed of 0.5ms^{-1}. However, at that speed, the coherence time of the channel is only about 0.54s. Consequently,

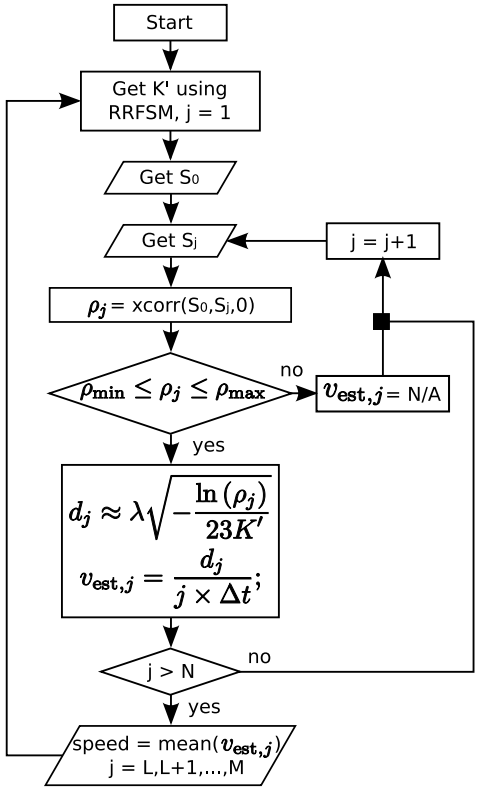


Fig. 8. Speed estimation algorithm utilising spatial correlation (modified RRFSM). S_j represents the j^{th} envelope. $\rho_{\min}$ and $\rho_{\max}$ define the coefficient range considered for speed estimation.

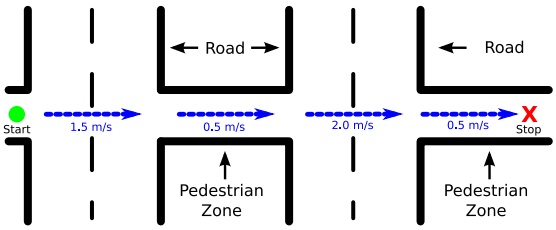


Fig. 9. Model of pedestrian behaviour. Starting from a standstill, the person crosses the street at an average speed of 1.5ms^{-1}. On reaching the pavement on the other side, the speed reduces to a leisurely 0.5ms^{-1}. When at the second road, the average speed increases to a brisk 2.0ms^{-1} for a fast crossing. Once on the other side, the speed returns to an average of 0.5ms^{-1} and is maintained till the pedestrian reaches the destination and stops.

the relevant CREs are no longer strongly correlated and a reliable speed estimate cannot be obtained.

The estimated profiles show that the algorithm tends to momentarily lose track of the speed when the device undergoes a sudden change in speed. This can be attributed to a radical change in the behaviour of the channel model. At constant speeds, the channel response is a function of linear time, t. However, when the device undergoes constant acceleration, the response becomes a function of the time squared, t^2, as shown in the Appendix B. This implies that any CRE recorded at some time t_0 is in fact

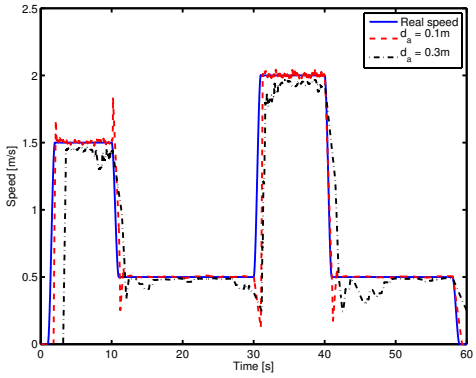


Fig. 10. Simulation of the relative RF signature matching algorithm for antenna separation distances (d_a) of 0.10m and 0.30m. The estimates are passed through a 4ms long moving average filter for smoothing. The speed profile used is the one illustrated in Fig. 9.

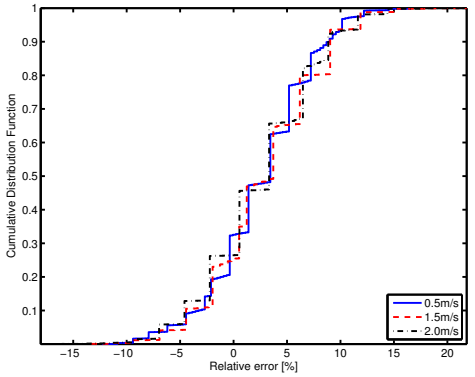


Fig. 11. CDF of relative errors in estimating speeds of 0.5ms^{-1}, 1.5ms^{-1} and 2.0ms^{-1} at an antenna separation distance of 0.10m. The time resolution issue described in section III-A manifests itself in the form of discrete steps in the error CDF.

similar to a CRE at time t_0^2. This fundamental change in the recorded CREs leads to a disruption of algorithm and produces erroneous results at the points where there is a sudden change in the speed. Once the speed stabilises, however, the algorithm is quick to recover and continues tracking the real speed with a low error margin.

Fig. 11 shows cumulative distribution function (CDF) of the percentage errors ($\frac{v_{\text{est}}-v_{\text{real}}}{v_{\text{real}}} \times 100\%$) in estimating speeds of 0.5ms^{-1}, 1.5ms^{-1} and 2.0ms^{-1} at an antenna separation distance of 0.10m. From the plot it is evident that the performance of the algorithm is independent of the actual speed of the device. On average, the relative error is approximately 2.67% with a standard deviation of 5%. The quantisation of detectable speeds due to time resolution available (section III-A) also leads to discretisation of the relative error. This is the reason behind the stair-like CDF.

It may be noted that the average error is positive – indicating a tendency to overestimate the speed. Since

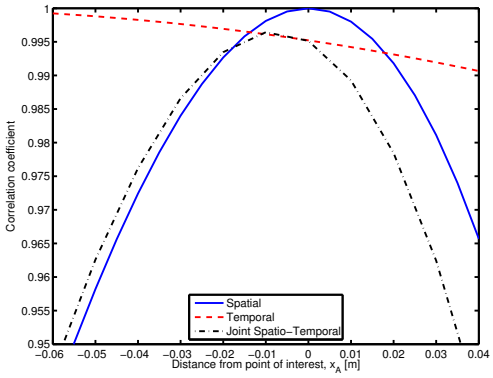


Fig. 12. Spatial and temporal correlation coefficients for a device with $d_a = 0.10\text{m}$ moving at 15ms^{-1}. As antenna B approaches the position previously occupied by antenna A (point of interest, x_A), the spatial correlation increases while the temporal correlation decreases. This leads to a joint maximum at a position shortly before x_A – leading to an underestimate of the time delay and hence an overestimate of the speed.

d_a is constant, this implies that the time delay is underestimated. This is due to the time-varying nature of the channel and a difference in the rate at which the spatial and temporal correlation functions evolve. Fig. 12 shows how the spatial and temporal correlation changes for a device with $d_a = 0.10\text{m}$ moving at 15ms^{-1}. Although a time axis, t, is not explicitly shown, it is related to the displacement axis, s, by the simple equation $t = \frac{s}{15} + c$, where c is some constant. It can be seen that as antenna B approaches the desired spatial location, x_A, the spatial correlation rapidly increases while the temporal correlation gradually decreases. As a result, the joint maximum of the correlation functions is not at the desired location but shortly before it – leading to an underestimate of the time delay and hence an overestimate of the speed.

B. Method II: Modified RRFSM

The simulator implemented is similar in design to the one described in Section IV-A.

Fig. 13 shows the performance of the modified RRFSM algorithm under different channel conditions as represented by the parameter K. It is assumed that perfect knowledge of the channel is available and hence the correlation coefficients ρ and the estimates of the parameter K are error free. From the plot, it is immediately clear that there is a near-perfect match between the real speed profile and that estimated by the algorithm – showing that the method described works as expected under ideal circumstances. The lack of any substantial difference between the estimates indicates that the results are not affected by the channel conditions. This is due to the assumption that both ρ and K are error free.

To investigate the impact of small uncertainties in the estimates of the parameter K, a uniformly distributed random relative error between $\pm 10\%$ is introduced. Fig.

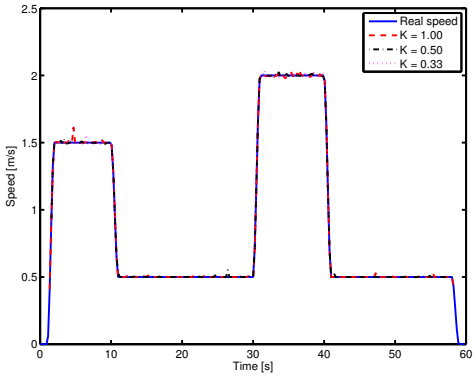


Fig. 13. Simulation of the spatial correlation algorithm for various channel conditions. Perfect knowledge of the channel parameters is assumed.

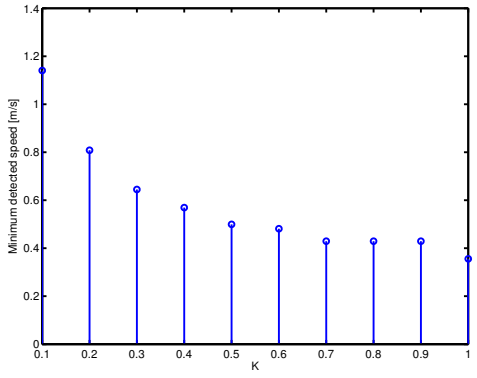


Fig. 15. Minimum speed detectable by the spatial correlation algorithm under various channel conditions for the speed profile shown in Fig. 13.

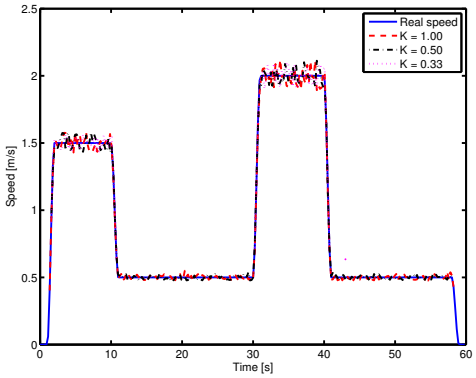


Fig. 14. Simulation of the spatial correlation algorithm for various channel conditions. The estimate of the parameter K is not exact and contains a random relative error uniformly distributed between $\pm 10\%$.

14 shows the results for different channel conditions as represented by the parameter K. As exact knowledge of the channel is no longer available, larger errors in the estimates appear. An analysis of the error statistics is provided later.

The plot reveals that no estimates are obtained for speeds that are below approximately 0.3ms^{-1}. A comparison of the minimum speeds detectable with the algorithm under various (exact) values of the parameter K is shown in Fig. 15. The general trend appears to be that the higher the value of K, the lower is the speed detectable. There are three contributing factors to this observation: the first is the relationship between ρ and K as described by (25), the second is related to the duration of each "run" of the algorithm and the third is tied to the choice of the "admissible" range of correlation coefficient values as shown in (29). The following attempts to explain the role each of these factors:

1) For a fixed value of d, (25) shows that $\rho \approx \exp[-XK]$ where X is a constant greater than zero. This implies that the larger the value of K, the smaller the ρ for a given d.
2) At each run of the algorithm, the signal envelope is estimated at intervals of Δt seconds for a total duration of T seconds where T is predetermined and fixed. At low speeds, a relatively small distance is travelled by the device in T seconds and as a result, the channel still remains highly correlated after that run.
3) As explained in Section III-B, only those values of ρ that fall within a certain range are utilised for speed estimation. Values that are too high or too low are ignored to minimise errors.

The consequence of the above is that at low speeds, the correlation coefficient for smaller values of K is higher than the upper bound defined and hence no estimate can be obtained. For the same speed, (25) implies that a larger K may perform better as it leads to a lower value of ρ. If that value is lower than the upper bound to admissible correlation coefficients, an estimate of the speed can be obtained. Since the minimum detectable speed clearly depends on a number of user-set parameters, it may be possible to lower the bound by increasing the time between runs and relaxing the $[\rho_{\min}, \rho_{\max}]$ limits placed on the correlation coefficients considered for speed estimation.

To investigate the error performance of the algorithm, simulations are performed at typical pedestrian speeds of 1.0ms^{-1}, 1.5ms^{-1} and 2.0ms^{-1}. The percentage relative error in the estimates (where available) are computed using $\frac{v_{\text{est}} - v_{\text{real}}}{v_{\text{real}}} \times 100\%$. When the channel parameters are known exactly, the mean error in the estimates is less than 0.04% with a standard deviation of approximately 0.86%.

Fig. 16 shows the CDF of the errors when the estimate of the parameter K contains a random relative error uniformly distributed between $\pm 10\%$. As the error CDFs are very similar to one another, it is reasonable to conclude that the relative error in the estimates is independent of

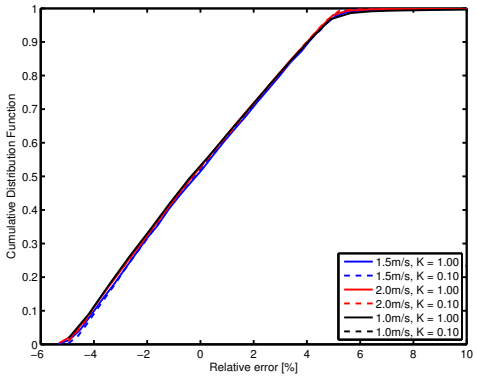


Fig. 16. CDF of relative errors in estimating speeds of 1.0ms^{-1}, 1.5ms^{-1} and 2.0ms^{-1}. The estimate of the parameter K has a random relative error uniformly distributed between $\pm 10\%$.

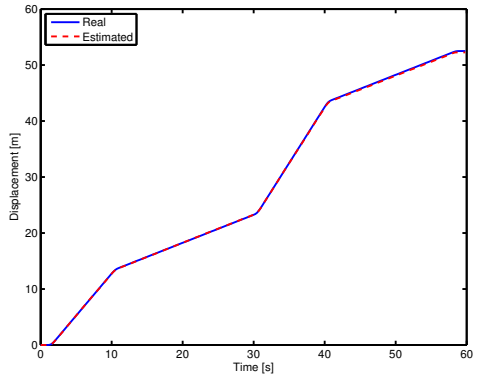


Fig. 17. Displacement (real and estimation) of a device moving according to Fig. 9 using the modified RRFSM algorithm with a 10% uncertainty in the parameter K. Despite the errors in K, the estimate is still excellent (error of 24.5cm after 60s) due to the fact that overestimates and underestimates in the speed cancel each other (as seen in Fig. 14).

both the speed and the actual channel characteristics. With a 10% uncertainty in the parameter K, the average error in the estimates is less than 0.15% with a standard deviation of approximately 3%.

Fig. 17 shows how the estimate of the displacement is affected by the errors in the speed estimates for a device moving according to the model in Fig. 9. After 60s, the error in displacement estimate is a mere 24.5cm. The error in the estimate is very low due to the fact that overestimates and underestimates in the speed cancel each other – resulting in a mean error that is very close to zero. This is particularly beneficial in context of PDR where the accumulation of error is a common problem.

To summarise, the basic RRFSM algorithm provides good results (2.67% error around pedestrian speeds on average) with a relatively low complexity. The modified RRFSM scheme requires additional computations but produce excellent speed estimates (0.15% error on average) even when accurate estimates of the model parameters are unavailable.

V. Summary and Conclusion

The spatial correlation properties of a multipath fading channel are exploited for speed estimation. If, in addition, heading information is available through an electronic compass, for example, indoor pedestrian dead reckoning can be performed. This enables the determination of the position of a mobile device, and hence, allows mobile navigation at places where GPS fails to work.

First, it is demonstrated that the RRFSM algorithm based on two antennas separated by a known distance provides speed estimation errors of less than 2.67% on average at typical pedestrian velocities. A by-product of this algorithm is a metric that characterises the actual propagation environment which, in turn, helps establish an environment specific relationship between the spatial correlation coefficient and the antenna displacement. This relationship is exploited by a modified RRFSM algorithm, with a further improvement of the accuracy (relative error of 0.15% or less on average even when the propagation environment cannot be estimated exactly) at the cost of a slightly increased computational complexity. Both algorithms can be combined intelligently to provide the best estimates possible. Where applicable, the modified RRFSM algorithm can be utilised to improve the speed estimate. Otherwise, the basic RRFSM algorithm is used as a fall-back solution since it generally also provides good estimates.

Simulations also show that when the modified RRFSM method is used for speed and subsequently displacement estimation, the error accumulated is very low. After 60s, the estimate differs by only 24.5cm for a typical pedestrian speed profile. This indicates that the algorithm is ideal for use in PDR applications where a high rate of accumulation of error is a common problem.

As no assumptions are made about the environment, the schemes are completely self-sufficient. The only requirement is a source of RF signal that leads to a fading channel at the receiver. Since the methods do not rely on the properties of any particular wireless standard, any wideband system capable of effectively capturing the effect of a fading channel is usable. In this paper, DVB-T is chosen for its wideband nature (8MHz channels) and ubiquity.

With the recent influx of MIMO capable devices, the low computational complexity of the proposed algorithms and the fact that some of the required information such as the channel estimates is simply recycled from the data detection process, the speed estimation algorithms can easily be integrated into current mobile communication devices.

At this point, one of the largest open issues is the extension of the algorithms to multiple dimensions. As mentioned earlier in Section III-A, this can be achieved with at least two antennae in each of the spatial dimensions desired. Preliminary investigations with multi-

antenna arrays have produced promising results but require further research. The multi-antenna array is also useful for heading estimation – opening up the possibility of "one-step" location estimates using the proposed basic algorithms.

Other issues that merit further research are assessment and mitigation of errors due to acceleration, investigations of the robustness of the algorithms under real-world conditions and the development of a mechanism for the determination of the level of accuracy in any estimate of the channel dependent parameter K.

APPENDIX A
DERIVATION OF THE RELATIVE ERROR IN K

Given a correlation coefficient ρ' and the associated distance d', an estimate K' can be obtained using (28):

$$\begin{aligned} K' &= -\frac{\ln(\rho')}{23}\left(\frac{\lambda}{d'}\right)^2 \\ &= C\frac{\ln(\rho')}{(d')^2}, \end{aligned} \tag{34}$$

where C is a constant. Assuming that the real value of the coefficient is ρ, the true K is then:

$$\begin{aligned} K &= -\frac{\ln(\rho)}{23}\left(\frac{\lambda}{d'}\right)^2 \\ &= C\frac{\ln(\rho' - e_{\rho'})}{(d')^2}, \end{aligned} \tag{35}$$

where $e_{\rho'}$ is the relative error in the correlation coefficient.

Therefore, the error in the estimate for K is given by:

$$\begin{aligned} K' - K &= C\frac{\ln(\rho')}{(d')^2} - C\frac{\ln(\rho' - e_{\rho'})}{(d')^2} \\ &= C\frac{\ln\left(\frac{\rho'}{\rho' - e_{\rho'}}\right)}{(d')^2}. \end{aligned} \tag{36}$$

Rewriting (27) as

$$\begin{aligned} (d')^2 &= \frac{\lambda^2}{-23} \times \frac{\ln(\rho)}{K} \\ &= C\frac{\ln(\rho' - e_{\rho'})}{K} \end{aligned} \tag{37}$$

and substituting into (36), an expression for the relative error in K, e_K, is obtained:

$$e_K = \left|\frac{K - K'}{K}\right| = \left|\frac{\ln\left(\frac{\rho'}{\rho' - e_{\rho'}}\right)}{\ln(\rho' - e_{\rho'})}\right|. \tag{38}$$

APPENDIX B
ACCELERATION AND CHANNEL RESPONSE

For a receiver with a single antenna, the spatio-temporal transfer function, $H(t, f, x)$, with time invariant multipath components can be written as

$$H(t, f, x) = \sum_{l=0}^{L-1} A_l e^{j2\pi f_{d(l)} t + j\phi_l - j2\pi\tau_l f} a(\theta_l) \tag{39}$$

where $f_{d(l)}$ is the Doppler spread associated with the component l and all other parameters are as previously defined in Section II. Substituting the one-dimensional version of (2) into (39) and rewriting the Doppler spread as a function of the speed v and angle ω_l,

$$\begin{aligned} H(t, f, x) &= \sum_{l=0}^{L-1} A_l \exp\Big[j2\pi\frac{v}{\lambda}\cos(\omega_l)\, t \\ &\quad + j\phi_l - j2\pi\tau_l f - \beta\cos(\theta_l)\, x\Big]. \end{aligned} \tag{40}$$

Since $v = v_0 + bt$ and $x = x_0 + v_0 t + \frac{1}{2}bt^2$, where v_0 is the initial speed, x_0 is the initial position and b is the constant acceleration,

$$\begin{aligned} H(t, f, x) &= \sum_{l=0}^{L-1} A_l \exp\Bigg[j2\pi\frac{v_0 + bt}{\lambda}\cos(\omega_l)\, t \\ &\quad + j\phi_l - j2\pi\tau_l f \\ &\quad - \beta\cos(\theta_l)\left(x_0 + v_0 t + \frac{1}{2}bt^2\right)\Bigg]. \end{aligned} \tag{41}$$

From (41) it is clear that the response is non-linear in t when the acceleration is non-zero.

REFERENCES

[1] J. Collin, O. Mezentsev, and G. Lachapelle, "Indoor Positioning System Using Accelerometry and High Accuracy Heading Sensors," in *Proc. of ION GPS/GNSS 2003 Conference*. Portland, OR: Institute of Navigation, Sep. 9–12, 2003.

[2] J. Kappi, J. Syrjarinne, and J. Saarinen, "MEMS-IMU Based Pedestrian Navigator for Handheld Devices," in *Proc. of ION GPS 2001 Conference*. Salt Lake City, UT: Institute of Navigation, Sep. 11–14, 2001, pp. 1369 – 1373.

[3] C. Randell, C. Djiallis, and H. Muller, "Personal Position Measurement Using Dead Reckoning," *Proceedings of the 7th IEEE International Symposium on Wearable Computers 2003*, pp. 166–173, Oct.18–21, 2005.

[4] A. Abdi and M. Kaveh, "Level Crossing Rate in Terms of the Characteristic Function: a New Approach for Calculating the Fading Rate in Diversity Systems," *IEEE Transactions on Communications*, vol. 50, no. 9, pp. 1397–1400, Sep. 2002.

[5] L. Zhao and J. W. Mark, "Mobile Speed Estimation Based on Average Fade Slope Duration," *IEEE Transactions on Communications*, vol. 52, no. 12, pp. 2066–2069, Dec. 2004.

[6] R. Narasimhan and D. C. Cox, "Speed Estimation in Wireless Systems Using Wavelets," *IEEE Transactions on Communications*, vol. 47, no. 9, pp. 1357–1364, Sep. 1999.

[7] R. B. Ertel, P. Cardieri, K. W. Sowerby, T. S. Rappaport, and J. H. Reed, "Overview of Spatial Channel Models for Antenna Array Communication Systems," *IEEE Personal Communications [see also IEEE Wireless Communications]*, vol. 5, no. 1, pp. 10–22, Feb. 1998.

[8] P. Höher, "A Statistical Discrete-Time Model for the WSSUS Multipath Channel," *IEEE Transactions on Vehicular Technology*, vol. 41, no. 4, pp. 461–468, Nov. 1992.

[9] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, ISBN: 0130422320, Dec. 2001.

[10] ETSI EN 300 744 v1.5.1 (2004-06), *Digital Video Broadcasting (DVB): Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television*, European Telecommunications Standards Institute (ETSI) Std., Jun. 2004.

[11] G. Durgin and T. Rappaport, "Theory of Multipath Shape Factors for Small-Scale Fading Wireless Channels," *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 5, pp. 682–693, 2000.

[12] M. Afgani and H. Haas, "Speed Estimation Using Relative Radio Frequency Signature Matching," in *Proc. of the 66th IEEE Vehicular Technology Conference (VTC)*, Baltimore, USA, Oct. 1–3, 2007, pp. 1970–1974.

[13] S. Beauregard and H. Haas, "Pedestrian Dead Reckoning (PDR) and GPS for Indoor Positioning," in *Proc. of 3rd Workshop on Positioning, Navigation and Communication (WPNC)*, Hannover, Germany, Mar. 16, 2006.

Mostafa Afgani received his B.Sc. in Electrical Engineering and Computer Science (2004) and M.Sc. in Electrical Engineering (2006) from Jacobs University Bremen, Germany. He is currently working towards his PhD degree at the Institute for Digital Communications (IDCOM) in the University of Edinburgh.

Sinan Sinanović has obtained his Ph.D. in electrical and computer engineering from Rice University, Houston, Texas in 2006. In the same year, he has joined Jacobs University Bremen in Germany as a post doctoral fellow. In 2007, he has joined the University of Edinburgh in the UK where he currently works as a research fellow in the Institute for Digital Communications. He has previously worked for Texas Instruments in Dallas, Texas on ASIC for the central office modem. While working for Halliburton Energy Services in Houston, Texas, he developed acoustic telemetry receiver for which he was awarded US patent (# 7158446). He is a member of the Tau Beta Pi engineering honour society and a member of Eta Kappa Nu electrical engineering honour society. He earned his M.S. from Rice University and his B.S.E.E. (summa cum laude) from Lamar University in Texas. He won national mathematics contest in Bosnia and was awarded honourable mention at the International Mathematics Olympiad in Hong Kong, both in 1994.

Karim Khashaba received his B.Sc. in Electrical Engineering and Computer Science in 2007 from Jacobs University Bremen. He is currently working towards his M.Sc. in Communication Systems at the Technical University Munich.

Harald Haas received the PhD degree from the University of Edinburgh in 2001. His main research interests are in the areas of wireless system engineering and digital signal processing, with a particular focus on interference aware MAC protocols, multiuser access, link adaptation, scheduling, dynamic resource allocation, multiple antenna systems and optical wireless communication. From 2001 to 2002, Dr. Haas was project manager at Siemens AG (Information and Communication Mobile Networks) for an international research project on new radio access technologies. Dr. Haas joined International University Bremen (Germany), now Jacobs University Bremen, in September 2002 where he has since been Associate Professor of Electrical Engineering. In June 2007, Dr. Haas joined the University of Edinburgh (Scotland/UK) where he is Reader in the Institute for Digital Communications (IDCOM). Dr. Haas received a best paper award at the International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) in Osaka/Japan in 1999 and holds more than 12 patents in the area of wireless communications. Dr. Haas contributed a chapter to the "Handbook of Information Security" entitled "Air Interface Requirements for Mobile Data Services" by John Wiley & Sons, Inc. He co-authors a book entitled "Next Generation Mobile Access Technologies: Implementing TDD" with Cambridge University Press. His work on optical wireless communication was selected by a jury of renowned scientists installed by the German ministry of education and research (BMBF) for publication in "100 Produkte der Zukunft (100 Products of the Future)" authored by Nobel Laureate T. W. Hänsch.

Hardware Implementation of a Kullback-Leibler Divergence Based Signal Anomaly Detector

Mostafa Afgani, Sinan Sinanović and Harald Haas
Institute for Digital Communications,
The University of Edinburgh, Edinburgh EH9 3JL, UK
Email: {m.afgani, s.sinanovic,h.haas}@ed.ac.uk

***Abstract*—A Field-Programmable Gate Array (FPGA) based implementation of the Kullback-Leibler Divergence (KLD) algorithm for real-time detection of anomalies in periodic signals is presented. To aid in the development of the hardware platform, an optimised, application specific, differential-equation version of the KLD equation is also proposed. Successful tests of the implementation against 10 MHz wireless signals show that the algorithm is indeed capable of real-time analysis of high speed test vectors.**

I. Introduction

Anomaly detection refers to the process of locating unusual and unexpected events that may exist alongside nominal samples in a dataset. It is a process that is utilised in large number of diverse application domains, including communications and biomedical sciences. One example of the use of anomaly detection in the field of communications is in cognitive radio [1], [2], [3] where it can be utilised to detect interference. In context of biomedical sciences, it can be used to detect irregularities in vital signs such as electrocardiogram (ECG) traces [4]. An extensive survey of current anomaly detection techniques and application domains is provided in [5].

The aforementioned survey reveals that there are many different approaches to solving the anomaly detection problem, each with its own set of advantages and disadvantages. However, there is one drawback that is shared by most algorithms: computational complexity that is quadratic in the number of data points. As a result of this setback, it is difficult to adapt these techniques for real-time and online processing of the input data. To overcome this challenge, a simple algorithm based on Kullback-Leibler Divergence [6] was developed and presented previously in [2] and [7].

KLD is a convenient and robust method of measuring the difference between two data sets in a statistical sense. Due to its versatility and general appeal, it finds use in fields as diverse as economics [8] and computational neuroscience [9]. As a statistical comparison tool, KLD can be also employed for the *automatic* and *real-time* detection of *unusual* (anomalous) data segments.

While the aforementioned papers provide a thorough discussion of the algorithm, they utilise a MATLAB based software implementation that can only post-process the data. In order to perform real-time analysis of high-speed, high-bandwidth data, a dedicated hardware implementation is required. This paper presents a Xilinx Virtex-4 FPGA based hardware platform that meets that goal.

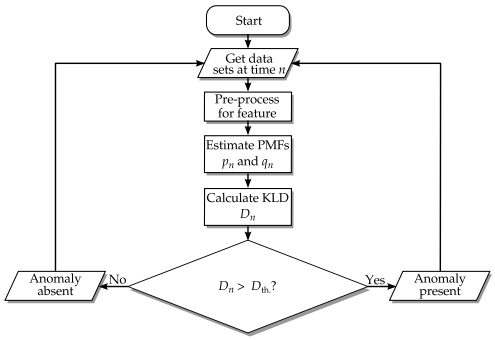


Fig. 1. The algorithmic flowchart. KLD is used to compare the statistical distribution of a test data set against that of a reference. If the divergence, D_n, is greater than some predefined threshold, $D_{\text{th.}}$, the test set may be anomalous.

The rest of this paper is organised as follows: Section II briefly introduces the algorithm and provides a description of the optimisations and approximations that must be made for an efficient implementation. Details of the actual design and the hardware platform are provided in Section III. Some experimental results are shown in Section IV and Section V concludes the paper.

II. The Algorithm & Practical Considerations

Given two data sets P_n and Q_n, at time n, that contain samples from "domain" X, it is possible to obtain empirical estimates of the associated Probability Mass Functions (PMFs) p_n and q_n from a non-parametric model such as a histogram. Once the PMF estimates are available, the KLD between them can be calculated using [6]

$$D\left(p_n \parallel q_n\right) = \sum_{x \in X} p_n\left(x\right) \log_2 \frac{p_n\left(x\right)}{q_n\left(x\right)}, \tag{1}$$

where $x \in X$. Since base-2 logarithm is used, the divergence is measured in *bits*. KLD between two PMFs is generally asymmetric: i.e. $D\left(p_n \parallel q_n\right) \neq D\left(q_n \parallel p_n\right)$. When $p_n = q_n$, the KLD is zero; otherwise it is a positive real number ($\mathbb{R}_+$). For brevity and convenience, $D\left(p_n \parallel q_n\right)$ will also be referred to as D_n in this paper.

TABLE I
COMPLEXITY ANALYSIS OF KLD

Operations	Σ	$\times$	$\div$	log
	$2\|P_n\| + N$	N	$3N$	N
Memory	$\|P_n\| + T_p + 2N$			

A flowchart of the KLD based anomaly detection algorithm is shown in Fig. 1. The algorithm starts with the acquisition and processing of the two data sets to be compared. Generally, one set is the reference that is known to be "good" while the other is the one under test. The associated PMFs are then estimated by means of histograms. Once the PMF estimates are available, the KLD is computed using (1). If the divergence obtained is greater than a predefined threshold, the test set is deemed to be anomalous.

When applied to the problem of detecting anomalous events in periodic signals, the data sets are obtained by applying two non-overlapping acquisition windows of the same size to the signal. The window centres are separated by an amount equivalent to the signal period. Since signals with periodic structures are expected to have statistical characteristics that are also periodic, analysis of signal segments that are separated by a duration equivalent to the period is reasonable [2].

If the KLD at each time step (i.e. sample index) is to be computed directly as prescribed by (1), the following steps must be taken:

1) Estimate the PMFs associated with each of the windows by means of histograms. The histograms are constructed using bins B which partition X:

$$X = \bigcup_{l=1}^{N} B_l.$$

N is the number of bins used to construct the histogram.
2) Compute the base-2 logarithm of the ratio of the PMFs.
3) Multiply and compute a summation with N terms.

While the steps involved are simple and straightforward, the storage (data buffers) and the number of arithmetic operations required grow linearly with the size of input data sets. As these data sets can be very large, it can easily lead to scenarios where it is no longer possible to provide for the resources required by the algorithm. An analysis of the complexity follows and is summarised in Table I.

The input data sets themselves require a buffer capable of holding at least $|P_n| + T_p$ samples, where T_p the number of samples corresponding to the signal period and $|P_n|$ $(= |Q_n|)$ is the size of the data windows. Only a single buffer is required for the input data since one of the data sets is essentially a T_p-delayed version of the other. Computing the frequency count over the bins (B), for the purpose of estimating the histograms, requires up to $|P_n|$ **additions** each for the two windows. Once the histograms are available, the PMFs are obtained by **dividing** the frequency count in each of the N bins by $|P_n|$. Two buffers of size N each are then required to store the resulting PMFs. Computation of the KLD from the PMFs then require a further N **divisions**, **logarithms**, **multiplications** and **additions** respectively.

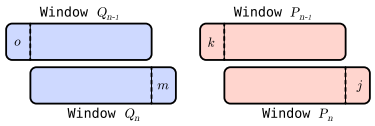


Fig. 2. At any given time, only a maximum of four histogram bins need to be updated (two per window). Bins incremented (fresh samples) are denoted by j and m while bins decremented (old samples at the end of the window) are denoted by k and o.

The analysis reveals that a direct interpretation of the equation to hardware would be inefficient, inflexible and computationally expensive. The inefficiency arises from the fact that at each successive time instance, the PMFs and the KLD are completely recalculated; even though it is only a single sample that changes in each of the data sets. The inflexibility comes from the fact that the computational complexity depends on N, implying that a direct interpretation would be limited by the initial choice of the histogram bin resolution. Finally, logarithms and divisions can be very costly to implement in hardware. Fortunately, there are several well-known methods that can be adopted to overcome each of these challenges.

A. Application-Specific Reinterpretation of KLD

With knowledge of the problem at hand, the KLD equation (1) can be manipulated to yield significant efficiency and cost improvements. The first of these is achieved through the use of the identity $\log\left(\frac{a}{b}\right) = \log(a) - \log(b)$:

$$\begin{aligned} D_n &= \sum_{x \in X} p_n(x) \log_2 \frac{p_n(x)}{q_n(x)} \\ &= \sum_{x \in X} p_n(x) \left\{ \log_2 [p_n(x)] - \log_2 [q_n(x)] \right\}. \end{aligned} \quad (2)$$

This simple modification exchanges the costly division operation for a logarithm. Since the logarithms of probabilities can be obtained in a very simple and efficient manner (as described in Section II-C), a great reduction in complexity and hardware resource requirements is achieved.

The next set of changes involve making application-specific modifications to the equation. Since the purpose of the algorithm is to analyse periodic signals by means of two sliding windows, it holds that at any given instance, only one sample in each of the data sets changes. This in turn implies that only a maximum of 4 histogram/PMF bins need to be updated at that instant – two for each data set/window. The two bins per window account for the freshly acquired sample (bin frequency count incremented by one) and the sample that is dropped at the end of the window (bin frequency count reduced by one). An illustration is provided in Fig. 2.

This also means that the KLD values change very little between subsequent time steps for this particular application; suggesting that it is possible to re-write the equation (2) in the

TABLE II
COMPLEXITY ANALYSIS OF DKLD

Operations	Σ	$\times$	$\div$	log
	20	8	16	16
Memory	$\lvert P_n\rvert + T_p + 2N$			

form of a differential (updater) equation:

$$\begin{aligned} D_n &= \sum_{x\in X} p_n(x)\left\{\log_2\left[p_n(x)\right] - \log_2\left[q_n(x)\right]\right\} \\ &= D_{n-1} \\ &\quad -p_{n-1}(j)\left\{\log_2\left[p_{n-1}(j)\right] - \log_2\left[q_{n-1}(j)\right]\right\} \\ &\quad +p_n(j)\left\{\log_2\left[p_n(j)\right] - \log_2\left[q_n(j)\right]\right\} \\ &\quad -p_{n-1}(k)\left\{\log_2\left[p_{n-1}(k)\right] - \log_2\left[q_{n-1}(k)\right]\right\} \\ &\quad +p_n(k)\left\{\log_2\left[p_n(k)\right] - \log_2\left[q_n(k)\right]\right\} \\ &\quad -p_{n-1}(m)\left\{\log_2\left[p_{n-1}(m)\right] - \log_2\left[q_{n-1}(m)\right]\right\} \\ &\quad +p_n(m)\left\{\log_2\left[p_n(m)\right] - \log_2\left[q_n(m)\right]\right\} \\ &\quad -p_{n-1}(o)\left\{\log_2\left[p_{n-1}(o)\right] - \log_2\left[q_{n-1}(o)\right]\right\} \\ &\quad +p_n(o)\left\{\log_2\left[p_n(o)\right] - \log_2\left[q_n(o)\right]\right\} \end{aligned} \tag{3}$$

where the four bin indices j, k, m, and o are assumed to be unique. If not, the duplicate terms in the equation are set to zero.

The differential equation form of KLD (DKLD) shows that its computational complexity is no longer dependant on the number of histogram bins utilised in evaluating the PMFs. Assuming that D_{n-1} is available, only 16 additions/subtractions, 16 logarithms, and 8 multiplications are needed to calculate D_n, regardless of the value of N. This opens the path for a fixed complexity, flexible and efficient implementation that can be easily updated to accommodate a wide range of histogram resolutions.

The computational complexity and storage requirement of DKLD is shown in Table II. Comparisons against the unmodified, direct interpretation version of KLD (Table I) reveals that while memory utilisation remains unchanged, there is a vast difference in the number of operations required. Regardless of the window size and histogram bin count, 60 operations are needed to compute the KLD. In addition to the 16 additions/subtractions, 16 logarithms and 8 multiplications required for the DKLD (3), 4 more additions/subtractions are required to update the affected histogram bins and 16 divisions are required to obtain the necessary PMFs at times $n-1$ and n.

B. RAM-based Histograms & PMF

The most obvious and straightforward approach to designing a histogram function/module involves the use of a number of parallel comparison and accumulation functions – one for each of the histogram bins. This is clearly inefficient and leads to a cumbersome implementation that is incapable of accommodating a bin resolution higher than the number of parallel comparison/accumulation paths available. As a result, it is necessary pursue alternative designs.

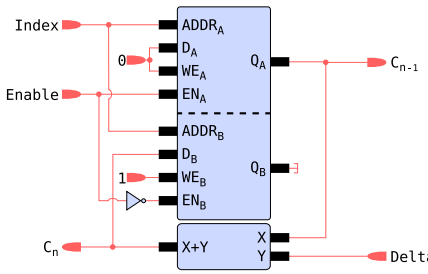


Fig. 3. Basic architecture of a dual-port RAM based histogram module. Bin indices are used as addresses and an externally supplied integer, "Delta", is used to modify the frequency count associated with a given histogram bin. The module simultaneously outputs the frequency count associated with supplied bin index at times $n-1$ and n. The "Enable" signal ensures that only one port of the RAM is active at a time – avoiding read-write collisions.

One possibility is to use Random-Access Memory (RAM) blocks; with bins mapped to individual memory locations. This way, an arbitrary number of bins can be supported by simply changing the size of the addressable space – allowing for a truly flexible implementation. An example of such a design is shown in Fig. 3. Dual-port RAM (DPRAM) blocks are convenient since they provide dedicated read and write access to the same memory space, simplifying the design.

The DPRAM is addressed using the index of the bin to be updated and the adjustment is provided via the port labelled "Delta". Port A is used to read the frequency count before the update, essentially providing the means to compute p_{n-1} and q_{n-1} as required by (3). Updating the count by the adjustment factor then provides the values necessary to compute p_n and q_n. The modified bin frequency count is stored back into the memory via port B. Since port A is used exclusively for reading while port B is used exclusively for writing, the "Write Enable" (WE) pins are held at logical `LOW` (Boolean *false*) and `HIGH` (Boolean *true*) respectively. The data bus of port A is also held `LOW` since it is not used for writing to the memory. Joining the port "Enable" (EN) pins by means of a NOT gate allows for exclusive read or write access to the device using a single control signal. This is necessary to prevent the read/write collisions that would otherwise arise due to simultaneous read/write access to a single memory address. Since each input requires a read followed by a write operation, it is necessary to operate the DPRAM module at twice the clock speed of the input.

To compute the PMF, the frequency counts from the histogram must be divided by $|P_n|$. As stated earlier in Section II, general-purpose, full-precision division operations can be very costly to implement in hardware. However, with a simple constraint on the denominator, that cost can be reduced to virtually zero. If the data window size, $|P_n|$, is always chosen to be a power of two (PoT), i.e.

$$|P_n| = 2^\sigma, \quad \sigma = 0, 1, 2, \ldots$$

a division is simply a left shift in the position of the binary point in the fixed-point representation of the numerator. As

this operation does not require any hardware, it is essentially "free". Therefore, through the imposition of a simple constraint on an algorithm parameter, the number of divisions required by the algorithm can be reduced zero.

C. Logarithm via Lookup Tables

Apart from the division operation, the only other function that poses a challenge is the base-2 logarithm. Although it can be implemented by means of the well-know CORDIC [10] algorithm, that approach is unnecessarily complex for this particular application. Instead, the use of a single pre-computed look-up table is proposed.

The purpose of the module is to compute the base-2 logarithm of PMFs. As a result, the input is guaranteed to have a value between 0 and 1. This tight bound on the input implies that a lookup table of $\log_2(\rho)$, where ρ is in [0,1], addressed by the input itself is viable. The size of the table then dictates the precision available.

Since the Most Significant Bit (MSB) is always zero and the remaining Lowest Significant Bits (LSB) are never all zero (due to bin prefilling [2]) in an M-bit fixed-point representation of the input, a table with $2^{M-1}-1$ entries is sufficient. The table that needs to be pre-computed is then:

$$\log_2\left(\frac{i}{2^{M-1}}\right); \quad i = 1, 2, \ldots, 2^{M-1}-1.$$

The $M-1$ LSB of the input can be directly used to address the table. Since the table is pre-computed, it reduces the run-time computational complexity of the function to zero at the expense of memory.

Read-Only Memory (ROM) blocks are typically utilised for lookup tables. However, it is more efficient to use DPRAM blocks for this particular application as a single DPRAM block can substitute for two ROM blocks by allowing simultaneous read operations. To use a DPRAM as a ROM block, the data and WE ports should be held at `LOW` while the EN pins are held at `HIGH`.

D. Resource Sharing

Resource sharing (RS) is another common trick that can lead to vast improvements in hardware utilisation efficiency when multiple parallel but identical execution paths exist in the design. It is the practice of serialising the parallel execution paths so that fewer resource units are required at the expense of a higher system clock speed. Since some additional control logic is also required, it is advisable to use this technique only when a significant reduction in resource utilisation is foreseen.

The concept of RS is illustrated in Fig. 4. If a design running at a speed of F_{clk} MHz contains r identical and parallel execution paths to compute some function $f()$, then it is possible to collapse the design to a single execution path by serialising the input and running $f()$ at a speed of $r \cdot F_{\text{clk}}$ MHz. The parallel/serial and serial/parallel conversion (time division multiplexing/demultiplexing) blocks provide the necessary control logic and interface the different clock domains by appropriately adjusting the I/O sample rates.

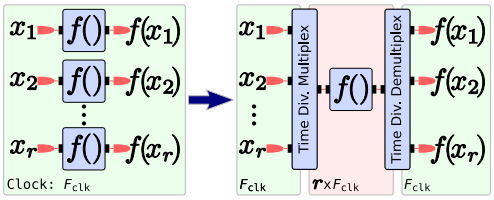


Fig. 4. Resource sharing for improved hardware utilisation efficiency. When multiple parallel but identical execution path exist in a design, it is possible to collapse them into fewer serial paths – resulting in a reduction in the resources required. However, higher clock speed and control logic in the form of time division multiplexing and demultiplexing blocks are required to accommodate the modified design.

TABLE III
COMPLEXITY ANALYSIS OF FP-DKLD

Operations	Σ	$\times$	$\div$	log
	20	8	0	0
Memory	$\lvert P_n\rvert + T_p + 2N + L$			

Looking back at (3), it is clear that the implementation will benefit from RS. At any given time, it is necessary to compute 16 independent base-2 logarithms – requiring as many ROM-based lookup tables if the data paths are kept independent and parallel. This highly inefficient use of the available resources can be remedied by employing DPRAM based lookup tables in conjunction with RS. With a RS factor of eight ($r = 8$), a single DPRAM module is all that is needed.

The computational complexity and storage requirement of Fixed-Point DKLD (FP-DKLD) with PoT constraint on $|P_n|$, factor eight RS, and a DPRAM based lookup table is shown in Table III. L is the size of the DPRAM used and depends on the precision required. Comparisons against Table I reveal that the application of some simple tricks and optimisations can lead to a tremendous reduction in the computational complexity of the algorithm for the price of a small increase in the memory utilised.

III. IMPLEMENTATION AND HARDWARE TEST CHAIN

For the hardware implementation, the Virtex-4 ML402 SX XtremeDSP Evaluation Platform is chosen as the target board. To facilitate and accelerate code development, Xilinx SystemGenerator 10.1 is used in conjunction with MATLAB R2007a for the primary design flow.

The design implemented is configurable and any changes made to the window size ($|P_n|$), signal period in samples (T_p), and number of histogram bins (N) to use require a simple and straightforward recompilation of the bit-stream file. Unlike the aforementioned parameters, the KLD threshold ($D_{\text{th.}}$) is configurable at run-time and does not require a recompilation of the design. This is achieved by using the on-board DIP switches as an 8-bit address to a lookup table containing 256 predefined threshold levels. The design has a single output in the form of a logical trigger signal that is `HIGH` when D_n is higher than $D_{\text{th.}}$.

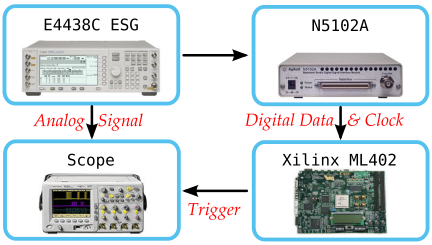


Fig. 5. Block diagram of the hardware test-bed. The E4438C ESG signal generator produces the signal under test in both analogue and digital format. The digital signal is passed to the FPGA platform via the N5102A digital signal interface module while the analogue signal is fed into the oscilloscope. The trigger signal from the FPGA core is also connected to the oscilloscope via a digital probe for comparison.

The module that computes the bin index, b, from the input sample value, u, utilises the following relationship

$$b = \left\lfloor \frac{u}{\Delta} \right\rfloor - \left\lfloor \frac{u_{\min}}{\Delta} \right\rfloor \tag{4}$$

where $u_{\min}$ is the lower bound of the input values and Δ is the bin width. Since the input is assumed to be a 2s complement number with zero fractional bits and a fixed word size, w, the lower bound of the input is

$$u_{\min} = -2^{w-1}$$

and the bin width is

$$\Delta = \frac{2^{w-1} - u_{\min}}{N}.$$

For a simpler and more efficient implementation, only the absolute value of the input is considered ($u_{\min} = 0$) and the number of histogram bins used is a PoT. These constraints reduce (4) to $b = u/\Delta$. Since Δ is also a PoT, the index is then easily obtained by bit-shifting the input by an appropriate number of bits. As stated earlier in Section II-B, bit-shift operations do not require any hardware; implying that the bin index is obtained from input sample essentially for free.

Since the DKLD algorithm (3) requires the probabilities associated with four (possibly different) indices from each PMF at any given time, a RS factor of four is used for each of the two PMF modules. Two DPRAM based log-lookup tables are then used to process the output. Since the RAM based histogram module must be operated at twice the clock speed of the module input which is already at $4F_{\text{clk}}$ due to RS, parts of the design must operate at $8F_{\text{clk}}$ MHz. To accommodate this requirement, the implementation is such that the input sample rate is F_{clk} MHz while the top level clock rate is $8F_{\text{clk}}$ MHz – providing eight clock cycles per sample (CCPS). Slower versions of the clock are then derived from the top level clock using appropriate clock-enable signals.

The hardware chain used to test and verify the algorithm is shown in Fig. 5. The Agilent E4438C ESG signal generator simultaneously provides analogue and digital versions of the signal under test. The digital data stream is connected to the FPGA platform via the Agilent N5102A Digital Signal Interface Module (DSIM) while the analogue signal is connected to a oscilloscope for display. The DSIM conditions the data (word size, bit alignment, clock relationship settings) and provides a synchronous clock signal that is used to drive the FPGA core. The trigger output from the FPGA platform is also connected to the oscilloscope via a digital probe so that it can be directly compared against the signal under test.

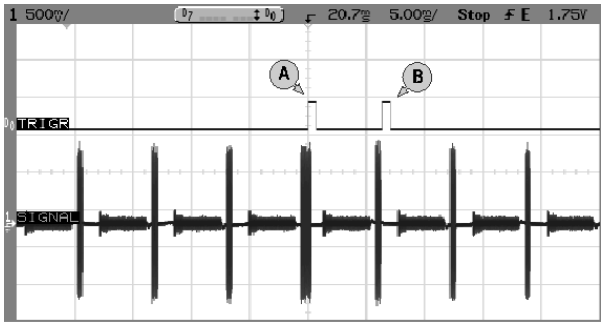


Fig. 6. Oscilloscope trace of a WiBro signal with a single anomalous frame and the associated trigger events. The KLD threshold is 0.0313. Trigger event A marks the start of the unusual segment of the anomalous frame. A second trigger event, B, is also obtained in the subsequent frame due to the disappearance of the anomalous feature.

Unfortunately, the DSIM only supports a maximum of four CCPS. Since the implemented design requires eight CCPS, it is not possible to directly connect the DSIM clock output to the top level clock input of the design. The solution is to produce a "wrapper" for the design that accepts the DSIM clock and multiplies it by two before feeding it to the core implementation. A Digital Clock Manager (DCM) module is ideal for this task as it is capable of digitally synthesising a range of frequencies. There are several DCMs available for use on the Virtex-4 SX chip targeted by the design. An added benefit of using the DCM based wrapper is that it also provides clock buffering and de-skewing.

IV. Empirical Results

A pair of Wireless Broadband (WiBro) signals known to contain a number of different anomalous data segments are used to test the hardware platform. The design is configured with $|P_n| = 4096$, $N = 8$, and $T_p = 64000$ to match the input signal period. The input word size is set to 12-bits in 2s complement format. The sample and DSIM clocks are set at 10 MHz and 40 MHz respectively to provide the four CCPS required by the top level DCM wrapper. The DCM then synthesises a phase-locked 80 MHz clock signal from the 40 MHz input to produce the eight CCPS needed by the core algorithm implementation.

A. Test Signal I

Fig. 6 shows the result of analysing the first WiBro signal using the FP-DKLD implementation. WiBro utilises time-division duplexing (TDD) and as a result the signal consists of a series of uplink (UL) and downlink (DL) frames. The UL frames appear to have greater signal strength due to the recording being made at the mobile terminal (MT) end. It is clear

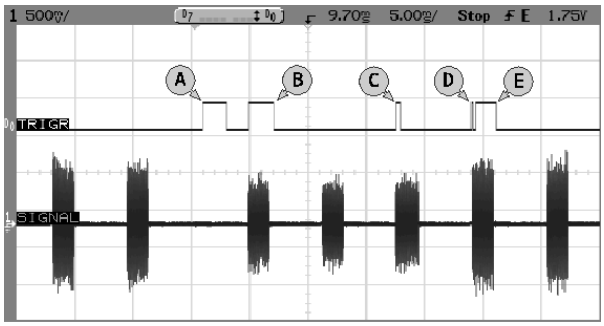


Fig. 7. Oscilloscope trace of a WiBro signal with multiple anomalous events and the associated trigger events. The KLD threshold is 0.0625. Triggers A and B are caused by a momentary change in the signal period. Trigger C flags the presence of a very brief command sequence that leads to a signal power level change. Trigger D marks the position in the subsequent frame where the power change inducing command was previously present and finally, trigger E corresponds to a sudden change in the overall signal power level.

from the oscilloscope trace that one of the UL frames is longer than the others and hence is anomalous. With $D_{\text{th.}} = 0.0313$, the FPGA implementation of the algorithm clearly succeeds in detecting the signal anomaly. The first trigger event obtained (A) coincides exactly with the anomalous segment of the unusual UL frame. A second trigger event (B) is observed when the UL frame structure subsequently returns to normal and the anomalous segment is no longer present.

B. Test Signal II

The second WiBro signal tested is shown in Fig. 7. Once again, UL power is much higher than DL power due to proximity to the MT. Analysing the signal with $D_{\text{th.}} = 0.0625$ is seen to produce five trigger events – corresponding to the three anomalous conditions known to be present in the signal.

Trigger events A and B are due to a momentary disruption in the natural frame period of the signal. The first event marks the position where the UL frame should have been but is not while the second event marks the opposite: finding a UL frame where there should be none.

Events C and D are caused by a very brief command sequence at the beginning of the fifth UL frame that causes the power control loop to be initiated – which is then responsible for event E. Although invisible to the naked eye, the algorithm succeeds in locating the anomalous command sequence as clearly demonstrated by trigger event C. Since that command sequence is no longer present in the subsequent UL frame, its disappearance is marked by trigger event D.

Once initiated, the power control loop causes a sudden increase in the signal power level. This behaviour can be considered to be anomalous and is flagged by trigger event E. There are no other events associated with the change in power level as the signal power is seen to remain high beyond this point.

While is not possible to show the result of different parameter combinations and test signals due to space constraints, experiments have shown that the implementation is indeed robust and capable of detecting a wide range of anomalous events in real periodic signals – as expected from the study performed in [2] and [7].

V. Conclusion

A hardware implementation of the KLD based anomaly detection algorithm for real-time analysis of periodic signals is described in this paper. An alternative, application specific, differential equation version of the KLD equation is developed to minimise the computational cost and implementation effort. A number of other resource saving optimisations such as RAM-based histograms, function (logarithm) lookup tables, and resource sharing are also described.

Experiments with real wireless test signals at 10 MHz have conclusively shown that the algorithm is robust, effective and can be used for the real-time detection of anomalies in high speed signals.

Acknowledgement

We would like to thank Roy Macnaughton and Peter Cain for the provision of test data. We greatly acknowledge the financial support of this project from Agilent and the University Relations Program.

References

[1] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[2] M. Afgani, S. Sinanović, and H. Haas, "Anomaly Detection Using the Kullback-Leibler Divergence Metric," in *Proc. of the First International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*, Aalborg, Denmark, Oct. 25–28, 2008.

[3] M. Afgani and H. Haas, "Information Content Analysis and Clustering for Signal Anomaly Detection," in *Proc. of the Vehicular Technology Conference (VTC)*. Anchorage, USA: IEEE, Sep. 20–23, 2009.

[4] J. Lin, E. Keogh, A. Fu, and H. V. Herle, "Approximations to Magic: Finding Unusual Medical Time Series," in *Proc. 18th IEEE Symposium on Computer-Based Medical Systems*, IEEE. Dublin, Ireland: IEEE, Jun. 23–24, 2005, pp. 329–334.

[5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[6] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed., ser. Wiley Series in Telecommunications, D. L. Schilling, Ed. John Wiley & Sons, Sep. 1991.

[7] M. Afgani, S. Sinanović, and H. Haas, "Information Theoretic Approach to Signal Feature Detection for Cognitive Radio," in *Proc. of the Global Telecommunications Conference (GLOBECOM)*. New Orleans, USA: IEEE, Nov. 30–Dec. 4, 2008.

[8] J. Robertson, E. W. Tallman, and C. H. Whiteman, "Forecasting Using Relative Entropy," Nov. 2002, FRB of Atlanta Working Paper No. 2002-22. [Online]. Available: http://ssrn.com/abstract=355460

[9] H. Nakahara and S.-I. Amari, "Information-Geometric Measure for Neural Spikes," *Neural Computation*, vol. 14, no. 10, pp. 2269–2316, Oct. 2002.

[10] J. S. Walther, "A Unified Algorithm for Elementary Functions," in *Proc. Spring Joint Computer Conference*, vol. 38, American Federation of Information Processing Societies. Atlantic City, New Jersey, USA: AFIPS Press, May 18–20, 1971, pp. 379–385.

Information Content Analysis and Clustering for Signal Anomaly Detection

Mostafa Afgani and Harald Haas
Institute for Digital Communications, Joint Research Institute for Signal and Image Processing,
The University of Edinburgh, Edinburgh EH9 3JL, UK
Email: {m.afgani, h.haas}@ed.ac.uk

***Abstract*—An information theoretic approach to detecting unusual events in radio signals is presented. Anomalies are detected through a measure of the events' information content. Clustering is utilised to reduce false-positives while allowing a lower discrimination threshold to be used for improved anomaly detection. Experiments with a real wireless local area network test signal shows that it is possible to achieve 100% detection rates while maintaining very low false discovery rate of 1%.**

I. Introduction

The term *cognitive radio* is used to describe smart, reconfigurable wireless communications systems with autonomous learning and decision making capabilities. Such devices aim to provide reliable communication links while making efficient use of valuable radio spectrum resources [1]. To achieve these goals, the system must monitor the radio environment and adapt its transceiver characteristics as necessary.

By virtue of its nature, the physical radio link provides a wealth of information about the radio environment to the cognitive radio system. Therefore, it is useful to analyse the signal envelope for changes or other unexpected events that may warrant a response. In order to maintain a seamless user experience, the analysis must be performed in real-time. This calls for sensing algorithms that are fast, simple and require minimal processing power.

Most existing techniques are inflexible, complex, or require a network of sensors ([2], [3]). By comparison, a robust and efficient method of statistically analysing a radio frequency (RF) signal for anomalies is presented in [4] and [5]. The algorithm proposed therein utilises Kullback-Leibler divergence (KLD) [6] and is applicable to signals with periodic characteristics (e.g. time division duplex (TDD) [7] based systems such as Wireless Broadband (WiBro)). Unfortunately, as the algorithm *requires* the signal under test to be periodic, it cannot be used to detect anomalies in random access systems such as wireless local area network (WLAN).

A complementary algorithm that does not require the signal to be periodic is presented in this paper. The proposed method utilises information content analysis to detect unusual events. Rare, low-probability events have a very high information content while regular events have close to zero. Therefore, information content can be used to discriminate between normal and anomalous events. Anomaly clustering is also employed to minimise false positives while allowing the use of a lower discrimination threshold for better anomaly detection. A detailed description of the proposed algorithm is provided in Section II.

A real WLAN signal with interference from a Bluetooth device is used to evaluate the performance of the algorithm. The effect of different parameter combinations on the detector is also investigated. The results are shown in Section III and used to develop guidelines for sensible parameter selection. Section IV highlights the contributions made and concludes the paper.

II. Information Content Analysis

The amount of information, $I\left(x\left[k\right]\right)$, conveyed by any discrete random event, $x\left[k\right]$, at time k, is directly related to its probability of occurrence, $p\left(x\left[k\right]\right)$ [8]:

$$I\left(x\left[k\right]\right) = -\log_2\left\{p\left(x\left[k\right]\right)\right\}; \quad k = 1, 2, \ldots \tag{1}$$

Since base-2 logarithm is used, information is measured in *bits*. The equation implies that an event with a very high probability of occurrence carries very little information. On the other hand, a large amount of information is conveyed by the occurrence of rare events (i.e. $I\left(x\left[k\right]\right) \to \infty$ as $p\left(x\left[k\right]\right) \to 0$). The quantity is always positive and monotonically increasing with decreasing values of event probability. Since unusual events are, by definition, rare (i.e. low probability), evaluating the information content associated with random events provides a convenient means of detecting anomalies. Information content analysis is essentially a non-linear scaling function that favours the unusual.

The radio spectrum is a limited resource that can be accessed by any suitably equipped device. As a result, interference is a problem that is inherent to any wireless communications system. Cognitive radio systems aim to detect interference and subsequently mitigate its effect through clever spectrum management. While regular, sustained interference is relatively straightforward to detect, spurious, randomly occuring cases are a challenge. Fig. 1 shows an example where a WLAN system experiences interference from a Bluetooth device sharing the same physical channel (2.4 GHz industrial, scientific and medical (ISM) band). The Bluetooth interferer shows the characteristic frequency hopping pattern while the WLAN system has a regular, repetitive frame structure. This particular example depicts a property that holds true for most random interference scenarios – interferers (signal anomalies) momentarily disrupt the regular structure of the intended signal.

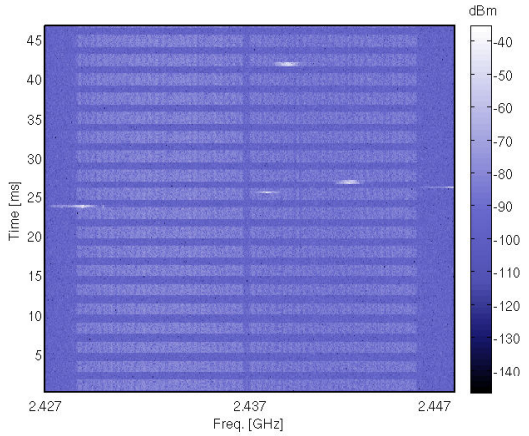


Fig. 1. Spectrogram of a WLAN signal experiencing interference from a Bluetooth device. The regular pattern is a single WLAN frame repeatedly transmitted by a signal generator every 1ms. The frequency hopping nature of the Bluetooth transmission is clearly visible in the plot.

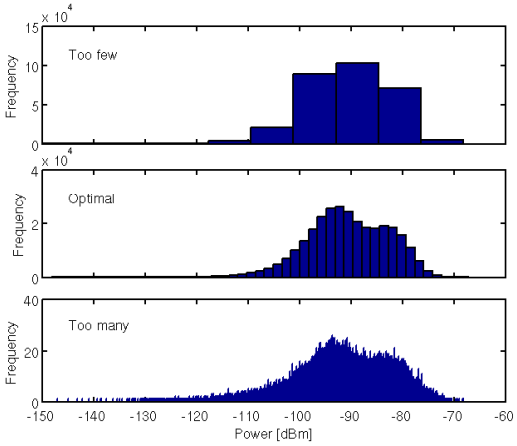


Fig. 2. Impact of the number of bins utilised, B, on the histogram of a random process (instantaneous power density at any time-frequency point of the signal in Fig. 1). When B is too small, the histogram is insensitive to small changes and does not effectively capture the subtleties of the process. At the other extreme, it is too sensitive and therefore susceptible to noise. The optimal B is with respect to some minimum error criterion [9].

Since anomalies lead to irregularities (unusual events) in the received signal, it is possible to devise a detector that utilises the information content measure introduced earlier. It may be done in the following manner:

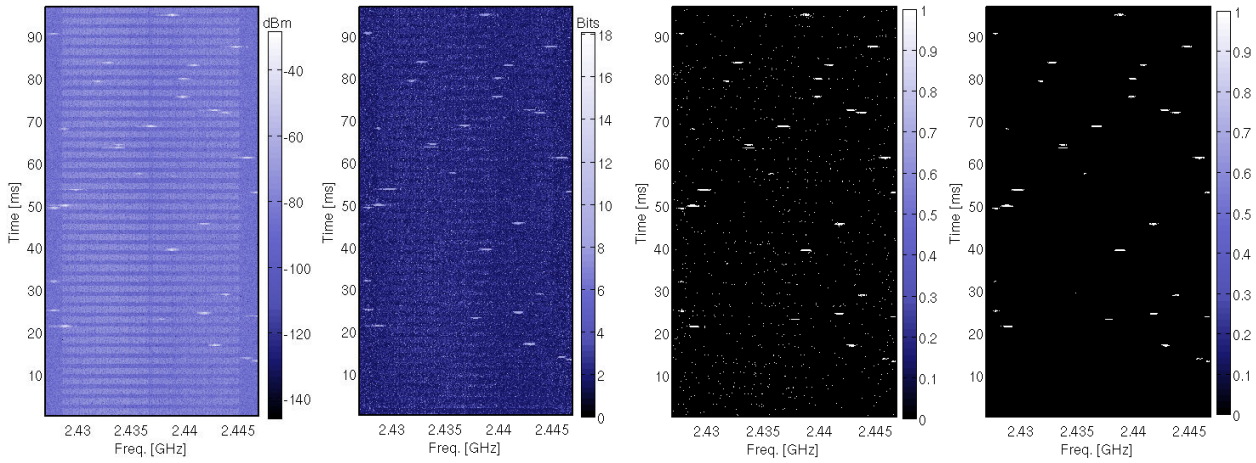
1) First and foremost, the type of event under observation must be established. This can be any property that is associated with the signal under test (e.g. instantaneous amplitude, phase, or power). If the event type chosen is measurable directly from the RF envelope, demodulation of the signal is unnecessary for anomaly detection.
2) A clean training signal is then used to determine reference event probabilities by means of an event histogram.
3) Once the reference histogram is obtained, analysis of the signal under test can commence. Events from the test signal are extracted and used to update the reference histogram. This yields updated event probabilities and hence the associated information content.
4) If the information content of any event is above some predefined threshold, a signal anomaly may be present.

It is clear that event histograms play a central role in the proposed algorithm. For events that arise from a continuous random process, the histogram bin width, Δ, is an important parameter as it defines the resolution and hence the sensitivity of the detector. Fig. 2 shows the effect of various bin widths on a histogram of instantaneous power density for the first 10ms of the signal in Fig. 1. The number of bins, B, utilised in a histogram is inversely proportional to the bin width, Δ. When the number of histogram bins utilised is too low, the histogram fails to capture subtleties in the behaviour of the random process. Sensitivity to spurious, anomalous events is also low and as a result, there is a high probability of missed detections. With too many bins, the histogram is noisy – increased sensitivity comes at a cost of much increased susceptibility to noise and false positives. The optimum number of bins shown in Fig. 2 is with respect to some minimum error criterion [9]. The effect of the histogram resolution on the detector performance is further explored in Section III-B.

The event histograms shown in Fig. 2 also reveal a potential challenge for the anomaly detection algorithm. It can be seen that the histograms have long tails with numerous low probability (i.e. high information content) events even when the signal is behaving nominally. Although this is expected for any analogue signal transmitted over a lossy physical channel, it raises the possibility that numerous false positives are observed at a detector that employs a simple information content threshold. It is certainly possible to reduce the number of tail events by using a smaller number of bins, but that leads to reduction in sensitivity and hence an increase in the probability of missed detections.

A closer look at the interference scenario in Fig. 1 reveals an important distinction between the anomalous events and the underlying signal – anomalies tend to appear in clusters while nominal low-probability signal events are decidedly "singular". This difference is the key to reducing the number of false positives while still maintaining a low rate of missed detections. The proposed algorithm is easily augmented to benefit from this insight: instead of triggering on individual high information content events, the detector should search for contiguous groups of events that exceed the predefined information content threshold.

A flowchart of the algorithm with simplified clustering is shown in Fig. 3. The general approach is the same as before, with the exception of the last step. With the simple clustering extension, detection of an anomaly is signalled only when a contiguous sequence of N previous events and the current event exceeds a predefined information content threshold. Sequence detection is used rather than full two-dimensional clustering to minimise the complexity of the algorithm. This is permissible since a sequence can be considered as a one-



(a) Test signal (b) Information content (c) Anomalies ($N = 0$) (d) Anomalies ($N = 2$)

Fig. 4. Analysis of a WLAN signal with interference from a Bluetooth device. (a) shows the spectrogram of the original signal. Data bursts from the Bluetooth device are clearly visible as high power, lightly shaded patches. (b) shows the information content of events (power density at any given time-frequency point) in the signal. (c) shows the anomalies detected (light patches) using a threshold of $1.25\sigma_{\mathrm{I}(16)}$ (7.15 bits) and a cluster length of 1. The result is noisy and there are a lot of false positives (appearing as singular, lightly shaded spots). (d) shows the anomalies detected when the threshold is left unchanged at $1.25\sigma_{\mathrm{I}(16)}$ and the cluster length is increased to 3. The outcome is now much cleaner with virtually zero false positives.

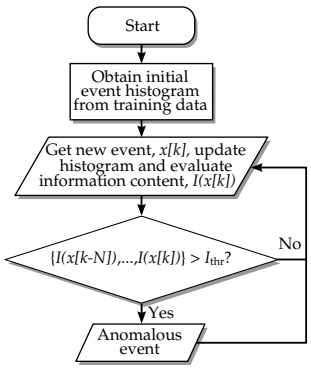


Fig. 3. Flowchart of information content analysis algorithm with clustered anomaly detection. First, clean data is used to initialise the reference event histogram and event probabilities. Then, events in the signal under test are used to update the reference histogram and event probabilities. The updated values are used to estimate the information content of the events. If the information content of $N+1$ contiguous events exceed a predefined threshold, I_{thr}, an anomaly may be present in the signal under test.

dimensional cluster. The effect of the cluster size utilised on the detector performance is examined in Section III-C.

The discrimination threshold is an important aspect of any detector. While the optimum threshold is problem and cost function specific, it is generally chosen to minimise missed detections while still maintaining a low rate of false positives. For the proposed anomaly detection algorithm, it is not possible to define a single information content threshold, I_{thr}, that is suitable for use with any arbitrary signal. I_{thr} is signal specific and may be set automatically using information obtained from the clean training data. After the reference event histogram and probabilities have been estimated, the reference information content associated each event type can be easily computed using (1). For B bins, the standard deviation, $\sigma_{\mathrm{I(B)}}$, of the reference information content provides a measure of the spread and may be used to obtain the threshold:

$$I_{\mathrm{thr}} = m \cdot \sigma_{\mathrm{I(B)}}. \tag{2}$$

m is a multiplicative factor greater than 1. The effect of I_{thr} on detector performance is investigated in Section III-A.

In order to evaluate the impact of B, I_{thr}, and N on detector performance, various combinations of the parameters are used to analyse the test signal shown in Fig. 4a. Performance is measured in terms of the detector *true positive rate* (R_{tp}) and *false discovery rate* (R_{fd}). R_{tp} is defined as the ratio of the number of correctly detected anomalous events (Σ_{tp}) to the total number of anomalous events present (Σ_{ta}):

$$R_{\mathrm{tp}} = \frac{\Sigma_{\mathrm{tp}}}{\Sigma_{\mathrm{ta}}} = 1 - \frac{\Sigma_{\mathrm{md}}}{\Sigma_{\mathrm{ta}}} \tag{3}$$

where Σ_{md} is the number of anomalous events that missed detection. R_{fd} is the ratio of false positives (Σ_{fp}) to the total number of anomalies detected (includes both Σ_{fp} and Σ_{tp}) [10]:

$$R_{\mathrm{fd}} = \frac{\Sigma_{\mathrm{fp}}}{\Sigma_{\mathrm{fp}} + \Sigma_{\mathrm{tp}}}. \tag{4}$$

R_{fd} is preferred over the more common *false positive rate* (R_{fp}) as it is more useful in this context. R_{fp} is defined as the ratio between Σ_{fp} and all non-anomalous events (Σ_{tn}) in the signal:

$$R_{\mathrm{fp}} = \frac{\Sigma_{\mathrm{fp}}}{\Sigma_{\mathrm{tn}}}. \tag{5}$$

Since Σ_{tn} is a very large number, R_{fp} is close to zero for most parameter combinations and therefore does not adequately reflect the variations observed in detector performance.

III. Results

In order to test the proposed algorithm, a signal similar to the one in Fig. 1 is used. Spectrogram of the test signal is shown in Fig. 4a. It is a much longer signal with numerous interference events to provide a statistically significant sample size. The plot depicts a real WLAN signal with Bluetooth interference captured over the air-interface. The WLAN signal is a single frame that is repeated every 1ms by a vector signal generator.

The signal spectrogram is estimated from the time series using non-overlapping Hamming windows that are 64μs long. A 1024 point FFT (fast Fourier transform) is used to obtain a frequency resolution of approximately 20kHz. The signal event under observation is the instantaneous power density at any given time-frequency coordinate. The first 10ms of the signal is free from interference and is therefore used for training purposes. 16 equally spaced histogram bins divide the range between the maximum and minimum power densities observed in the training data. The $\sigma_{\mathrm{I}(16)}$ for the training data is 5.72 bits.

Fig. 4b shows the information content of events in the test signal. As expected, the anomalous events are highlighted while the regular underlying structure is suppressed. The plot also shows that there is a lot of noise (tiny spots of high information content) from individual low-probability signal events that are otherwise "normal". The reason behind this is outlined in Section II.

Anomalies detected using a threshold of $1.25\sigma_{\mathrm{I}(16)}$ and a cluster length of 1 (i.e. only the current event) is shown in Fig. 4c. It is immediately obvious from the large number of small, lightly shaded spots that there are a lot of false positives. Again, singular low-probability signal events are responsible since they can potentially have higher information content than actual anomalous events. Keeping I_{thr} the same and increasing the cluster length to 3 yields the result shown in Fig. 4d. The plot reveals that a simple change in the cluster length is sufficient for reducing the number of false positives to virtually zero.

A. Effect of Threshold

As stated previously in Section II, the discrimination threshold (2) is an important aspect of any detector. Its impact on R_{tp} and R_{fd} is investigated using a cluster length of 3 and histogram bin sizes of 8, 16, and 32. The result of the analysis is shown in Fig. 5.

The plot shows that there is a hard I_{thr} boundary for each B after which R_{tp} drops rapidly. This implies that the majority of the anomalous events share similar characteristics and convey information equivalent to that boundary. When I_{thr} is increased further through the use of a larger threshold factor m, R_{tp} approaches zero due to an ever increasing number of missed detections.

At low information content thresholds, R_{fd} is also high – specially for high values of B. As explained earlier, a higher resolution makes the detector more susceptible to noise, leading to an increase in the number of false positives and hence the R_{fd}.

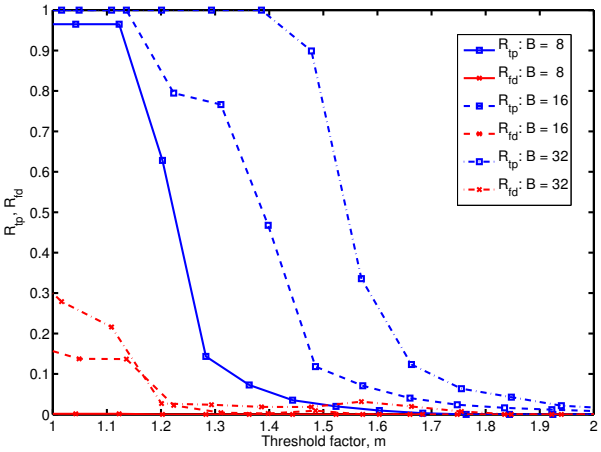


Fig. 5. Effect of distinction threshold (2) on detector R_{tp} and R_{fd}. The cluster length is 3. Histogram bin sizes of 8, 16, and 32 are used for comparison. R_{tp} for each B shows a sharp decrease when I_{thr} (i.e. $m{\cdot}\sigma_{\mathrm{I(B)}}$) is increased above a certain limit. Majority of the anomalous events have an information content less than this and miss detection. $m = 1.35$ and $B = 32$ yields the best performance.

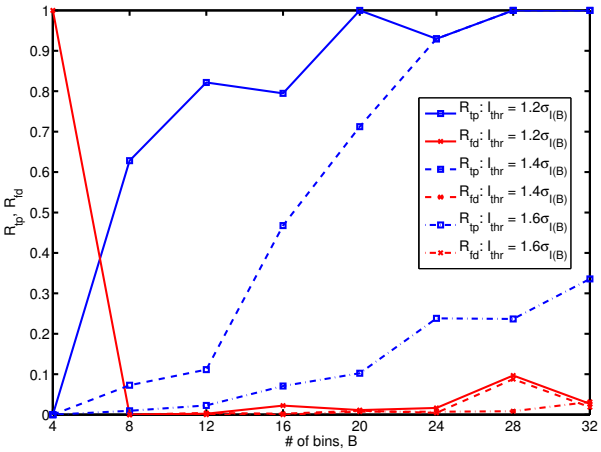


Fig. 6. Effect of histogram resolution on detector R_{tp} and R_{fd}. The cluster length is 3 and thresholds are $1.2\sigma_{\mathrm{I(B)}}$, $1.4\sigma_{\mathrm{I(B)}}$ and $1.6\sigma_{\mathrm{I(B)}}$. R_{tp} improves with resolution while R_{fd} deteriorates. 20 bins with a threshold of $1.2\sigma_{\mathrm{I(20)}}$ yields the best performance.

B. Effect of Histogram Resolution

Fig. 6 shows how the R_{tp} and R_{fd} varies for the test signal (Fig. 4a) with the number of histogram bins utilised (B). The cluster length utilised is 3 and I_{thr} of $1.2\sigma_{\mathrm{I(B)}}$, $1.4\sigma_{\mathrm{I(B)}}$ and $1.6\sigma_{\mathrm{I(B)}}$ are used for comparison.

The plot reveals that when $B = 4$, R_{tp} is zero and R_{fd} is unity for all thresholds tested. This is because the sensitivity is very low and no anomalies can be detected (R_{tp} = 0). Events exceeding the threshold are low-probability signal events and hence are all false positives (R_{fd} = 1). As B is doubled to 8, the resolution improves and there is a corresponding increase in the R_{tp}. The R_{fd} also drops to a negligibly small value. As

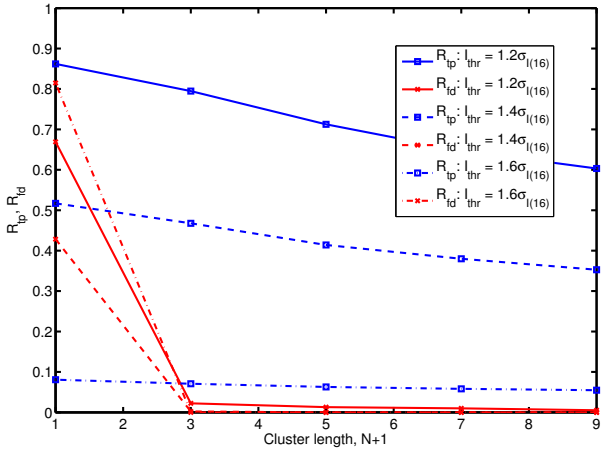


Fig. 7. Impact of cluster length on detector R_{tp} and R_{fd}. The number of histogram bins used is 16 and the thresholds utilised are $1.2\sigma_{\mathrm{I}(16)}$, $1.4\sigma_{\mathrm{I}(16)}$ and $1.6\sigma_{\mathrm{I}(16)}$. Even the smallest cluster length (N = 2) is shown to provide a significant improvement in the R_{fd}.

B is increased further, the R_{tp} increases due to better detector resolution. The R_{tp} improvements come at a cost however; the detector is more susceptible to noise at higher resolutions. This is evident from the gradual increase in the R_{fd}.

Comparison between the three detection thresholds reveals that a higher R_{tp} is achieved with a lower threshold. Unfortunately, this also leads to a higher R_{fd}. This behaviour is in accordance with the explanation provided in Section III-A.

C. Effect of Cluster Size

In order to investigate the impact of the cluster length on the detector, B is set at 16 and the analysis is performed for I_{thr} of $1.2\sigma_{\mathrm{I}(16)}$, $1.4\sigma_{\mathrm{I}(16)}$ and $1.6\sigma_{\mathrm{I}(16)}$. Fig. 7 shows the result.

The significance of clustered anomaly detection is immediately obvious. With $N = 0$, when clustering is not performed, there is an overwhelming number of false positives. This is indicated by the high R_{fd}. As soon as clustering is applied by setting $N = 2$, a dramatic drop in the R_{fd} is observed – showing that even minimal anomaly clustering is sufficient to yield a massive improvement in detector performance. By lowering the R_{fd}, clustering also allows a lower I_{thr} to be used to achieve a higher R_{tp}.

The impact of anomaly clustering on R_{tp} for a given I_{thr} is relatively low. As cluster size is increased, a gradual decrease is observed in the R_{tp}. This is expected since larger cluster sizes lead to missed detections around the edges of the interference patterns.

The plot also shows that higher thresholds lead to lower R_{tp}. This is also expected since a higher information content threshold leads to a higher number of missed detections.

From the analysis performed on the test signal, it is clear that it is not an easy task to determine a set of parameters that are inherently "optimal". Parameter values that increase the R_{tp} (good) often also lead to an increase in the R_{fd} (bad) and vice versa. Trade-offs must be made to meet the required detector performance characteristic (low R_{fd}, moderate R_{tp} or high R_{tp}, moderate R_{fd}). A moderate number of bins ($B = 20$), small cluster size ($N = 2$), and a threshold of $1.2\sigma_{\mathrm{I}(20)}$ bits ($m = 1.2$) provides a good balance between R_{tp} (1.0) and R_{fd} (0.01) for this particular test signal.

IV. Conclusion

The clustered anomaly detection technique proposed is effective, computationally simple, and does not require the signal to be demodulated. As it does not require the signal to be periodic, it provides a perfect complement to the algorithm described in [4] and [5].

Performance of the system clearly depends on a number of factors such as histogram resolution, threshold, and search cluster size. Impact of these parameters on the detector performance by means of R_{tp} and R_{fd} has been analysed and guidelines for appropriate values have been provided. It is shown that a R_{tp} of 100% and R_{fd} of 1% is possible for the signal tested with a suitable set of parameter choices.

Application of the proposed algorithm to other types of anomalous signals is beyond the scope of this paper and will be the subject of future research. Other open issues that merit further investigation are the effect of training data length and windowed event histograms on detector performance.

Acknowledgement

This project is in collaboration with Agilent Technologies (Edinburgh/Queensferry, UK). In particular, we would like to thank Roy Mcnaughton and Peter Cain for the provision of test data. We greatly acknowledge the financial support of this project from the Agilent University Relations Program.

References

[1] J. Mitola III, "Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio," Ph.D. dissertation, Royal Institute of Technology (KTH), May 2000.

[2] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[3] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation Issues in Spectrum Sensing for Cognitive Radios," in *Proc. of the Asilomar Conference on Signals, Systems and Computers*, vol. 1. Pacific Grove, CA, USA: IEEE, Nov. 7–10, 2004, pp. 772–776.

[4] M. Afgani, S. Sinanović, and H. Haas, "Information Theoretic Approach to Signal Feature Detection for Cognitive Radio," in *Proc. of the Global Telecommunications Conference (GLOBECOM)*. New Orleans, USA: IEEE, Nov. 30–Dec. 4, 2008.

[5] M. Afgani, S. Sinanović, and H. Haas, "Anomaly Detection Using the Kullback-Leibler Divergence Metric," in *Proc. of the First International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*, Aalborg, Denmark, Oct. 25–28, 2008.

[6] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed., ser. Wiley Series in Telecommunications, D. L. Schilling, Ed. John Wiley & Sons, Sep. 1991.

[7] H. Haas and S. McLaughlin, Eds., *Next Generation Mobile Access Technologies: Implementing TDD*. Cambridge University Press, ISBN: 13:9780521826228, Jan. 2008.

[8] J. G. Proakis, *Digital Communications*, 4th ed., ser. McGraw-Hill Series in Electrical and Computer Engineering, S. W. Director, Ed. McGraw-Hill Higher Education, Dec. 2000.

[9] H. Shimazaki and S. Shinomoto, "A Method for Selecting the Bin Size of a Time Histogram," *Neural Computation*, vol. 19, no. 6, pp. 1503–1527, Jun. 2007.

[10] Y. Benjamini and Y. Hochberg, "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *Journal of the Royal Statistical Society - Series B (Methodological)*, vol. 57, no. 1, pp. 289–300, 1995. [Online]. Available: http://www.jstor.org/stable/2346101

Information Theoretic Approach to Signal Feature Detection for Cognitive Radio

Mostafa Afgani, Sinan Sinanović and Harald Haas
Institute for Digital Communications,
The University of Edinburgh, Edinburgh EH9 3JL, UK
Email: {m.afgani, s.sinanovic, h.haas}@ed.ac.uk

***Abstract*—Cognitive radio (CR) systems need to be able to adjust the transceiver characteristics in response to stimuli received from the radio environment. Therefore, monitoring the wireless signal in real-time for stimuli such as unexpected changes due to sporadic interference in radio frequency band of operation is at the core of such systems. In this paper, a method of detecting anomalies in a periodic signal by means of statistical analysis of its envelope is described. The proposed scheme makes use of the Kullback-Leibler divergence between probability distributions drawn from analogous segments of the periodic signal to detect anomalous events. Experiments conducted on real wireless signals suggest that the method described is simple, robust and effective for the analysis of periodic signals.**

I. Introduction

Cognitive radios (CRs), by definition, are intelligent wireless communications devices built on top of software defined radio platforms. Inherently reconfigurable, such devices are able to change the operating parameters in response to usage patterns, network state and radio link conditions [1] – with the goals of providing the best possible service to the user as well as providing means to increase spectral efficiency.

Of all the input stimuli that affect the judgement of a CR system, the signal waveform prior to demodulation is sufficient for providing information related to the quality of the radio link. As a result, however, it is necessary to continuously monitor the wireless signal for events that may warrant a response. Such events are generally atypical of the signal and hence labelled as unusual. Incidents that can lead to unusual events range from interference to malfunctions in the transmitter.

The trivial, brute-force approach to the problem involves capturing a large set of signal data and comparing against "known-good" waveforms. The resource requirements are unacceptable and the algorithm cannot be implemented in real-time. A smarter approach is the interference temperature metric proposed in literature for the quantification of interference received at a terminal [2]. It is, however, computationally complex and requires the use of multiple sensors to take into account spatial variations in the radio environment. It is also limited in its use as it can only detect unusual events associated with interferers. This paper presents a novel statistical method that is both simple and capable of detecting a wide range unusual events in periodic signals in real-time. The proposed scheme also only requires the received radio waveform.

Data from segments of a periodic signal separated by the signal period are expected to have similar probability distributions; if not, an unusual event may be present in one of the segments. The Kullback-Leibler divergence (KLD) metric is a suitable tool for the quantisation of that difference as it measures the discrepancy between two sets of data in a statistical sense. While other metrics such as the L_1 distance (L1D) between the data probability distributions [3] can also be employed, the results obtained are poorer. The KLD metric, its properties and the application to signal anomaly detection is described in Section II.

Section III provides a discussion of time-frequency decomposition (spectrogram) and probability mass function (PMF) estimation techniques. Spectrograms provide a better starting point for the analysis compared to the time-domain signal. This is because it reveals unusual events that are localised in both time and frequency. PMF estimates are also integral to the algorithm as they are quantities compared by the divergence metric.

The algorithm developed is applied to real test cases obtained using signal measurement hardware and the results are discussed in Section IV. Section V concludes the paper.

II. Kullback-Leibler Divergence

The Kullback-Leibler (KL) divergence metric is a statistical measure that can be applied for the detection of irregularities within regular, periodic structures. It quantifies the difference between two probability distributions ($p(x)$ and $q(x)$) [3]:

$$D(p \parallel q) = \sum_{x \in X} p(x) \log_2 \frac{p(x)}{q(x)}. \quad (1)$$

As base-2 logarithm is used, the divergence can, thus, be measured in bits. It is, however, often not considered as a true distance metric as the divergence between a set of PMFs is generally not symmetric (i.e. $D(p \parallel q) \neq D(q \parallel p)$) and the triangle inequality is not satisfied. $D(p \parallel q) = D(q \parallel p) = 0$ if and only if $p(x)$ and $q(x)$ are identical; otherwise the divergence is always a positive, non-zero number. It belongs to a class of distance measures known as *f-divergence* (or *Ali-Silvey distances*) [4]. Although there are other divergence measuring metrics (e.g. Chernoff and Bhattacharyya distances), the KL metric has the lowest computational complexity and hence is the easiest to implement.

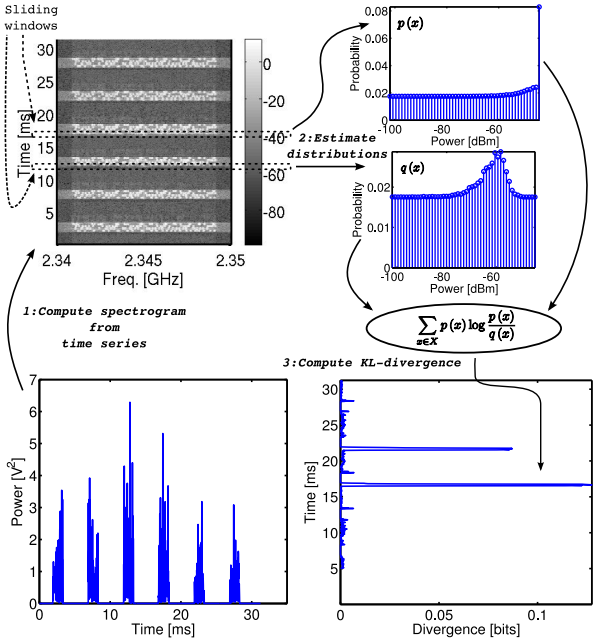


Fig. 1. Application of the Kullback-Leibler divergence metric in anomaly detection: if the divergence between two PMFs drawn from analogous portions of a periodic signal exceeds a threshold, detection of an unusual event is signalled. Each anomaly generally gives rise to two peaks (separated by T_{wc}) in the divergence – this is due to the twin windowing technique utilised.

It can be employed for the detection of anomalies in a periodic signal as follows:

1) Determine the period, T_s, of the signal under test.
2) Define two signal windows of the same size (T_w). The distance between the centres of the windows (inter-window separation, T_{wc}) must be the same as the signal period (i.e. $T_{wc} = T_s$) to ensure that the windows span over similar regions of the signal (i.e. for a like-for-like comparison). While there is much more freedom with respect to the choice of T_w, it is upper bounded by T_{wc}. The effect of the window size on detector performance is studied in Section IV-D.
3) Define what is meant by an "event". Events thus described are discrete and hence must have only non-overlapping "states" – i.e. there should not be any ambiguity in classifying a given occurrence of an event.
4) Estimate the PMF of events within the windows and compute the KLD.
5) If the divergence between the two windows is larger than some threshold, signal the detection of an unusual event. Otherwise, advance windows to the next set of data.

Fig. 1 illustrates the method for a signal utilising time division duplexing (TDD). First, the signal spectrogram is computed – revealing the TDD structure as a periodic series of light and dark segments. A pair of sliding windows separated by T_s are then applied to the spectrogram to collect the event (average power within a certain time-frequency grid) distribution statistics. The PMF in each window is represented by $p(x)$ and $q(x)$ respectively and used to compute the instantaneous KLD metric (KLDM) (1). The procedure is then repeated by moving the windows to the next data segment of the spectrogram.

Analogous segments of consecutive signal frames are compared using PMF estimating windows separated by T_s. Hence, the only *a priori* knowledge needed is the signal period.

The KLDM is also ideal for the detection of signal boundaries. Typically, the statistical distributions of the data on either side of a boundary are different from one another. This discrepancy can be exploited in conjunction with the KLDM by setting the inter-window gap to zero (i.e. $T_{wc} = T_w$). With PMF estimating windows thus positioned, the divergence is expected to reach a maximum at the moment when the signal boundary coincides with the inter-window border. This is explored further in Section IV-C.

III. Spectrum & PMF Estimates

Estimation of the time-frequency decomposition (spectrogram) of the signal is a first step for the detection algorithm. Although the divergence analysis can be carried out entirely in the time domain [5], the time-frequency representation is preferred as it also allows for band specific processing of the signal.

Amongst the numerous techniques of obtaining signal spectral estimates [6], the short-term Fourier transform (STFT) is taken to be the method of choice due to the simplicity of implementation offered. Schemes such as the *multitaper spectral estimation procedure* [7] offer better results, but at a much increased complexity. STFTs on the other hand are much simpler as it essentially only involves the evaluation of a series of fast Fourier transforms (FFT) which can be computed in a very efficient, rapid manner.

The STFT of a given time series is a one-to-many mapping with an infinite number of valid solutions. Therefore, any given representation is only an approximation of the true time-frequency decomposition of a signal. The resolution and amplitude accuracy are dictated by three important parameters: FFT size, windowing function, and the window length. The exact impact of each is throughly discussed in [8].

Computation of event histograms and eventually the PMFs is clearly at the heart of the proposed algorithm. Although the construction of a histogram is seemingly trivial, the choice of an appropriate bin size, Δ, is generally not – especially for events that arise from a continuous random variable. Shimazaki and Shinomoto show in [9] how the "optimum" Δ, according to some minimum error criterion and for certain classes of random variables, can be calculated. Although the method is simple and effective, it is too costly to implement in real-time as it requires the minimisation of a cost function through exhaustive search. As an alternative, a suitable Δ can be determined empirically. The choice of the bin width directly affects the number of bins, N, used ($N \propto \frac{1}{\Delta}$) and hence the complexity of (1): the number of terms in the summation is N. For real-time applications it is therefore desirable to pick the smallest N usable.

Information Theoretic Approach to Signal Feature Detection for Cognitive Radio

Mostafa Afgani, Sinan Sinanović and Harald Haas
Institute for Digital Communications,
The University of Edinburgh, Edinburgh EH9 3JL, UK
Email: {m.afgani, s.sinanovic, h.haas}@ed.ac.uk

***Abstract*—Cognitive radio (CR) systems need to be able to adjust the transceiver characteristics in response to stimuli received from the radio environment. Therefore, monitoring the wireless signal in real-time for stimuli such as unexpected changes due to sporadic interference in radio frequency band of operation is at the core of such systems. In this paper, a method of detecting anomalies in a periodic signal by means of statistical analysis of its envelope is described. The proposed scheme makes use of the Kullback-Leibler divergence between probability distributions drawn from analogous segments of the periodic signal to detect anomalous events. Experiments conducted on real wireless signals suggest that the method described is simple, robust and effective for the analysis of periodic signals.**

I. Introduction

Cognitive radios (CRs), by definition, are intelligent wireless communications devices built on top of software defined radio platforms. Inherently reconfigurable, such devices are able to change the operating parameters in response to usage patterns, network state and radio link conditions [1] – with the goals of providing the best possible service to the user as well as providing means to increase spectral efficiency.

Of all the input stimuli that affect the judgement of a CR system, the signal waveform prior to demodulation is sufficient for providing information related to the quality of the radio link. As a result, however, it is necessary to continuously monitor the wireless signal for events that may warrant a response. Such events are generally atypical of the signal and hence labelled as unusual. Incidents that can lead to unusual events range from interference to malfunctions in the transmitter.

The trivial, brute-force approach to the problem involves capturing a large set of signal data and comparing against "known-good" waveforms. The resource requirements are unacceptable and the algorithm cannot be implemented in real-time. A smarter approach is the interference temperature metric proposed in literature for the quantification of interference received at a terminal [2]. It is, however, computationally complex and requires the use of multiple sensors to take into account spatial variations in the radio environment. It is also limited in its use as it can only detect unusual events associated with interferers. This paper presents a novel statistical method that is both simple and capable of detecting a wide range unusual events in periodic signals in real-time. The proposed scheme also only requires the received radio waveform.

Data from segments of a periodic signal separated by the signal period are expected to have similar probability distributions; if not, an unusual event may be present in one of the segments. The Kullback-Leibler divergence (KLD) metric is a suitable tool for the quantisation of that difference as it measures the discrepancy between two sets of data in a statistical sense. While other metrics such as the L_1 distance (L1D) between the data probability distributions [3] can also be employed, the results obtained are poorer. The KLD metric, its properties and the application to signal anomaly detection is described in Section II.

Section III provides a discussion of time-frequency decomposition (spectrogram) and probability mass function (PMF) estimation techniques. Spectrograms provide a better starting point for the analysis compared to the time-domain signal. This is because it reveals unusual events that are localised in both time and frequency. PMF estimates are also integral to the algorithm as they are quantities compared by the divergence metric.

The algorithm developed is applied to real test cases obtained using signal measurement hardware and the results are discussed in Section IV. Section V concludes the paper.

II. Kullback-Leibler Divergence

The Kullback-Leibler (KL) divergence metric is a statistical measure that can be applied for the detection of irregularities within regular, periodic structures. It quantifies the difference between two probability distributions ($p(x)$ and $q(x)$) [3]:

$$D(p \parallel q) = \sum_{x \in X} p(x) \log_2 \frac{p(x)}{q(x)}. \quad (1)$$

As base-2 logarithm is used, the divergence can, thus, be measured in bits. It is, however, often not considered as a true distance metric as the divergence between a set of PMFs is generally not symmetric (i.e. $D(p \parallel q) \neq D(q \parallel p)$) and the triangle inequality is not satisfied. $D(p \parallel q) = D(q \parallel p) = 0$ if and only if $p(x)$ and $q(x)$ are identical; otherwise the divergence is always a positive, non-zero number. It belongs to a class of distance measures known as *f-divergence* (or *Ali-Silvey distances*) [4]. Although there are other divergence measuring metrics (e.g. Chernoff and Bhattacharyya distances), the KL metric has the lowest computational complexity and hence is the easiest to implement.

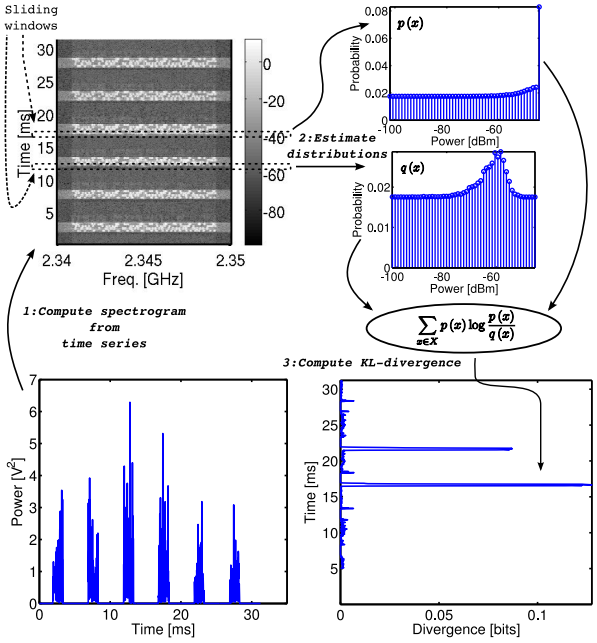


Fig. 1. Application of the Kullback-Leibler divergence metric in anomaly detection: if the divergence between two PMFs drawn from analogous portions of a periodic signal exceeds a threshold, detection of an unusual event is signalled. Each anomaly generally gives rise to two peaks (separated by T_{wc}) in the divergence – this is due to the twin windowing technique utilised.

It can be employed for the detection of anomalies in a periodic signal as follows:

1) Determine the period, T_s, of the signal under test.
2) Define two signal windows of the same size (T_w). The distance between the centres of the windows (inter-window separation, T_{wc}) must be the same as the signal period (i.e. $T_{wc} = T_s$) to ensure that the windows span over similar regions of the signal (i.e. for a like-for-like comparison). While there is much more freedom with respect to the choice of T_w, it is upper bounded by T_{wc}. The effect of the window size on detector performance is studied in Section IV-D.
3) Define what is meant by an "event". Events thus described are discrete and hence must have only non-overlapping "states" – i.e. there should not be any ambiguity in classifying a given occurrence of an event.
4) Estimate the PMF of events within the windows and compute the KLD.
5) If the divergence between the two windows is larger than some threshold, signal the detection of an unusual event. Otherwise, advance windows to the next set of data.

Fig. 1 illustrates the method for a signal utilising time division duplexing (TDD). First, the signal spectrogram is computed – revealing the TDD structure as a periodic series of light and dark segments. A pair of sliding windows separated by T_s are then applied to the spectrogram to collect the event (average power within a certain time-frequency grid) distribution statistics. The PMF in each window is represented by $p(x)$ and $q(x)$ respectively and used to compute the instantaneous KLD metric (KLDM) (1). The procedure is then repeated by moving the windows to the next data segment of the spectrogram.

Analogous segments of consecutive signal frames are compared using PMF estimating windows separated by T_s. Hence, the only *a priori* knowledge needed is the signal period.

The KLDM is also ideal for the detection of signal boundaries. Typically, the statistical distributions of the data on either side of a boundary are different from one another. This discrepancy can be exploited in conjunction with the KLDM by setting the inter-window gap to zero (i.e. $T_{wc} = T_w$). With PMF estimating windows thus positioned, the divergence is expected to reach a maximum at the moment when the signal boundary coincides with the inter-window border. This is explored further in Section IV-C.

III. Spectrum & PMF Estimates

Estimation of the time-frequency decomposition (spectrogram) of the signal is a first step for the detection algorithm. Although the divergence analysis can be carried out entirely in the time domain [5], the time-frequency representation is preferred as it also allows for band specific processing of the signal.

Amongst the numerous techniques of obtaining signal spectral estimates [6], the short-term Fourier transform (STFT) is taken to be the method of choice due to the simplicity of implementation offered. Schemes such as the *multitaper spectral estimation procedure* [7] offer better results, but at a much increased complexity. STFTs on the other hand are much simpler as it essentially only involves the evaluation of a series of fast Fourier transforms (FFT) which can be computed in a very efficient, rapid manner.

The STFT of a given time series is a one-to-many mapping with an infinite number of valid solutions. Therefore, any given representation is only an approximation of the true time-frequency decomposition of a signal. The resolution and amplitude accuracy are dictated by three important parameters: FFT size, windowing function, and the window length. The exact impact of each is throughly discussed in [8].

Computation of event histograms and eventually the PMFs is clearly at the heart of the proposed algorithm. Although the construction of a histogram is seemingly trivial, the choice of an appropriate bin size, Δ, is generally not – especially for events that arise from a continuous random variable. Shimazaki and Shinomoto show in [9] how the "optimum" Δ, according to some minimum error criterion and for certain classes of random variables, can be calculated. Although the method is simple and effective, it is too costly to implement in real-time as it requires the minimisation of a cost function through exhaustive search. As an alternative, a suitable Δ can be determined empirically. The choice of the bin width directly affects the number of bins, N, used ($N \propto \frac{1}{\Delta}$) and hence the complexity of (1): the number of terms in the summation is N. For real-time applications it is therefore desirable to pick the smallest N usable.

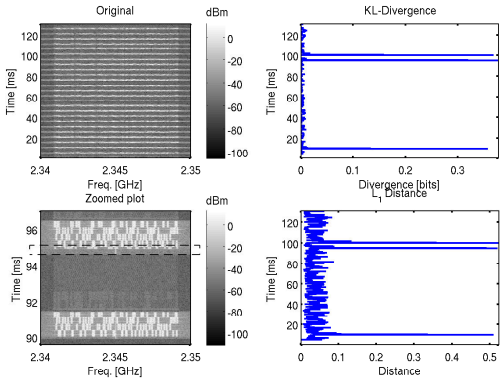


Fig. 2. *Left*: Spectrograms of a WiBro signal with power control inducing commands. The extra commands (located at 4.76ms and 94.76ms) in the original plot are impossible to detect with the unaided eye. Zooming in on the frames between 90ms and 97ms reveals the presence of additional data at the start of the second frame (as indicated by the dashed box). Darker shades represent lower power levels than lighter ones. *Right*: The presence of the unusual events are immediately revealed by both the KLD and L1D analysis of the signal. However, the L_1 plot is clearly noisier.

Once the histogram is constructed, the PMF must be evaluated. Empty bins in the histogram results in zeros in the PMF. This becomes a major concern when KLDs, as defined in (1), are to be evaluated: divisions involving zeros have to be handled in some manner. An easy workaround is to *preload* the histogram bins with an arbitrary number – such that there can be no empty bins. Krichevsky and Trofimov [10] reason that 0.5 is a good choice.

IV. Test Cases & Results

To evaluate the performance of the proposed scheme, periodic signals with two distinct classes of abnormalities are employed as test cases. Use of the technique for signal boundary detection and effect of the PMF window length on the detector are also investigated.

A. Test Signal A

Fig. 2 shows a measured signal that undergoes a sudden but subtle structural change. It depicts message exchanges between two entities using the WiBro (Wireless Broadband - IEEE 802.16e) standard. Although invisible in the original spectrogram, additional command sequences at 4.76ms and 94.76ms cause the power control loop to be initiated. Due to the presence of the extra command sequences, the associated frame is structurally different from the others – as seen in the lower spectrogram.

Table I lists the parameters used for the computation of the KLD and L1D between segments of the test signal. 25.6μs is a natural choice for the STFT window length as it provides individual symbol level resolution (symbols making up WiBro frames have a duration of 25.6μs). Empirical evidence suggests that the performance of the proposed algorithm is not significantly affected by the choice of the histogram bin size. Therefore, 32 bins, equally distributed between maximum and minimum power levels estimated from training data, are chosen. The PMF estimating window length is set to 256μs to take into account a statistically significant data set.

TABLE I
Parameter list for the analysis of signal A

Spectrogram:	
Windowing function	Hamming
Window length	25.6μs (one symbol)
FFT size	Next power of 2 larger than the window
Histogram:	
Number of bins, N	32
Divergence analysis:	
Window length, T_w	256μs
Window separation, T_{wc}	5ms (signal period)

The results of applying the algorithm to the signal are also shown in Fig. 2 next to the original signal spectrograms. With more than a 36-fold difference between the background divergence (BGD, represents the divergences obtained for "normal" segments of the signal) levels and the peaks produced by the events of interest, the proposed method in conjunction with KLD clearly succeeds at picking out the structural abnormalities in the signal. While the L1D analysis also detects the anomalies, the performance is poorer (only six-fold difference between BGD and the peak) compared to the KLD due to higher levels of BGD.

Every anomaly in the signal produces *two* peaks in the divergence, with the first peak being the "true" one. This is a side effect of the two windows needed for the calculations – each window produces a peak as it passes over the anomaly. There is only one peak associated with the first event as, at the beginning of the analysis, the "leading" window is positioned beyond it. The fact that each irregularity gives rise to two peaks in the divergence is not necessarily a disadvantage: detection of the second can be used to reinforce the detection of first as the peaks should always be separated by T_s.

B. Test Signal B

Fig. 3 demonstrates another distinct class of anomalies commonly encountered. It consists of interference that is both unpredictable and ephemeral in nature. The underlying periodic signal is a single wireless local area network (WLAN) frame repeated continuously by a vector signal generator. The highly localised interferer observed at around the 11.3ms mark is a data burst from a Bluetooth (BT) device.

The parameters used for the analysis are identical to those in Table I with the exception of the STFT window length and T_{wc}. The window length is set to 4μs to reflect the WLAN symbol duration and T_{wc} is set to 2.448ms in accordance with the frame repeat rate of the generator. The reasoning behind the choices are similar to those in Section IV-A.

Fig. 3 shows the result of the KLD analysis. Once again, there is a significant change in the divergence when the interferer is encountered. Although there should be two peaks associated with the BT interferer, only one is displayed. This

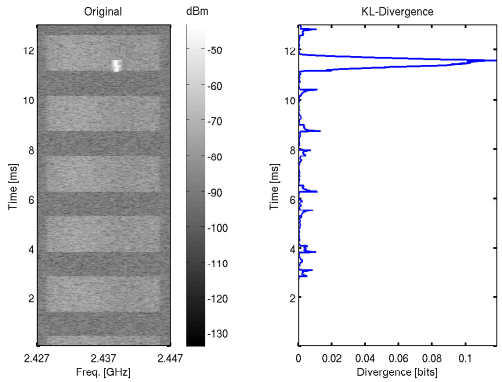


Fig. 3. *Left*: Spectrogram of a periodic signal with interfence from a Bluetooth device. The underlying signal is obtained through the repetitive transmission of a single wireless LAN frame from a signal generator. Darker shades represent lower power levels than lighter ones. *Right*: Analysis of the signal by means of the KLDM immediately reveals the presence of the interferer.

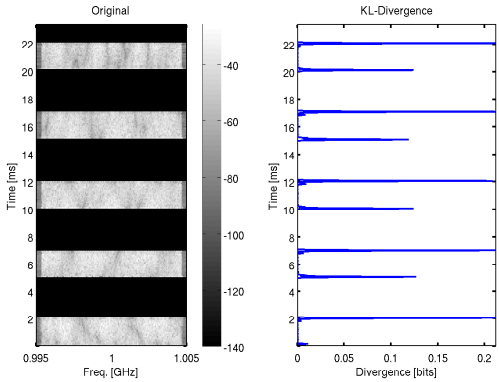


Fig. 4. Boundary detection using the KLDM. An anomaly detector is readily converted to a signal boundary detector by setting the windows adjacent to one another. The asymmetric nature of the metric manifests itself in the form of differing peak heights for the possible frame transitions (in and out).

is due to a lack of data beyond the displayed range (the second peak is expected at around the 13.74ms mark). It is also interesting to note that the BGD values appear to peak around the frame boundaries. This observation is further explored in the following section.

C. Boundary Detection

All parameters remaining the same as in Section IV-A, if the windows are placed adjacent to one another ($T_{wc} = T_w$), a signal boundary detector is obtained. Fig. 4 shows an example with $T_w = 78\mu$s. The signal displayed is a WiBro downlink session over a fading channel. Although the divergence peaks at every frame transition, the magnitude depends on the type of transition (noise $\rightarrow$ signal or signal $\rightarrow$ noise) due to the asymmetric nature of the KLDM. The detector resolution depends on the choice of T_w: it must be smaller than or equal to the length of the smallest detectable feature required.

The ability to detect signal anomalies and signal boundaries using the same framework is a valuable one. It is conceivable that the algorithm may be implemented as a self-learning scheme that uses boundary detection to determine the signal period prior to executing in the anomaly detection mode – nullifying the need for any user input.

D. Effect of PMF Window Length

As stated earlier, the length of the PMF estimating window has a clear impact on the detector. To quantise the performance under different values of T_w, a measure called *KLD ratio* (KLDR) is hereby defined. With the divergence plot from Fig. 3 as an example:

$$\mathrm{KLDR} = \frac{\mathrm{KLD}_{\mathrm{anom}}}{\mathrm{KLD}_{\mathrm{bg}}}, \tag{2}$$

where $\mathrm{KLD}_{\mathrm{anom}}$ represents the magnitude of the peak associated with the anomalous event and $\mathrm{KLD}_{\mathrm{bg}}$ represents the maximum observed in the noisy BGDs. This is a meaningful

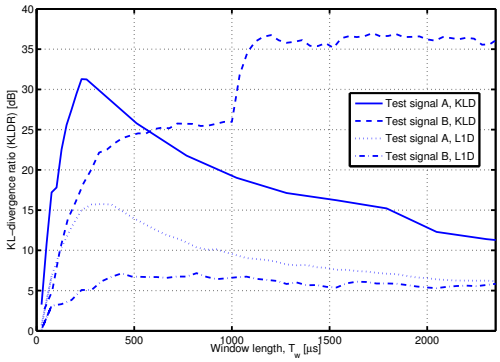


Fig. 5. Effect of the PMF window length, T_w, on the distinguishability of KLD and L1D peaks (i.e. the KLDR). *Test signal A* corresponds to the WiBro test signal (Section IV-A). *Test signal B* corresponds to the artificially generated periodic WLAN signal with the Bluetooth interferer (Section IV-B). The KLD metric performs better for both signals and all window lengths.

measure in context of detectors as a higher ratio implies a greater range over which a detection threshold can be applied. A similar metric called L_1 *distance ratio* (L1DR) can be defined for L_1 distance measures.

Fig. 5 shows the result of applying the KLDR measure to the test signals under a wide range of T_w values. The L1DR is also computed for performance comparison. For both test signals, the KLD performs significantly better than the L1D. The initial increase in the KLDR with increasing values of T_w is due to the fact that an increase in the amount of data used to compute the PMFs results in better estimates – helping to reduce BGDs and increasing the divergence associated with the abnormal event. As T_w is increased further, the KLDR reaches a maximum and then starts to drop as observed with test signal A. This behaviour is attributed to the fact that when the window is much larger than the abnormal event, the data associated with the event tends to become insignificant

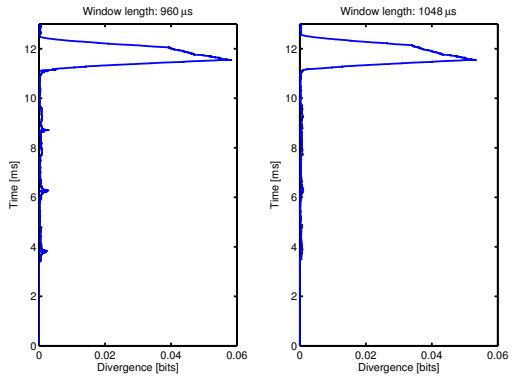


Fig. 6. KLD plots for test signal B with window lengths of 960μs and 1048μs respectively. The periodic peaks in the BGDs disappear once the window length is increased beyond the duration of the feature (inter-frame gap of 1000μs) responsible for those peaks.

in defining the shape of the anomalous PMF – resulting in a drop in the divergence and hence the KLDR. Test signal B is expected to behave similarly. However, it is not possible to observe the effect of $T_w > 2448\mu$s (period of the generated WLAN signal) as it would violate the condition $T_w \leq T_{wc}$ (Section II). The explanation for the L1DR plots follows the same line of reasoning as above.

From the plots, it is also clear that there is not a single value of T_w that is jointly optimal for both test cases. The choice maximising the KLDR is undoubtedly influenced by the dimensions of the anomalous entity. Nevertheless, it should be noted that the algorithm is still sufficiently robust with respect to the PMF window length; a satisfactory KLDR of 10dB or greater can be obtained for a wide range of values between 120μs and 2448μs for the test cases displayed.

The KLDR plot for test signal B shows an interesting behaviour as the the window length is pushed beyond 1ms – the KLDR suddenly increases and continues that trend before reaching a stable value at a T_w of around 1.25ms. This observation is explained by the boundary detecting capability of the detector.

The inter-frame spacing for the signal is 1ms. With T_w shorter than or equal to that, periodic peaks attributable to that gap are observed in the BGDs as seen in Fig. 3 and Fig. 6. Once T_w is increased beyond that "feature" length (e.g. 1048μs as in Fig. 6), the inter-frame gap can no longer be resolved (Section IV-C) and the peaks fade away – leading to large drop in the $\mathrm{KLD}_{\mathrm{bg}}$. As the $\mathrm{KLD}_{\mathrm{anom}}$ is not affected to the same extent by the change in T_w, there is a net increase in the KLDR.

Comparison between the divergence plots in Fig. 3 and Fig. 6 reveals that the peak associated with the anomaly widens with increasing T_w: the base of the peak is 0.72ms wide for $T_w = 256\mu$s, increasing to 1.64ms for $T_w = 1048\mu$s. The wider the peak, the larger the ambiguity regarding the location of the anomalous event. This is analogous to the time resolution problem of STFTs (Section III). A larger window provides a better KLDR at the expense of temporal resolution and vice-versa.

V. Conclusion

A new, robust statistical method for the detection of anomalies in periodic signals is presented. The "KLD detector" is shown to be effective against both localised and structural abnormalities as long as the test signal is periodic and an accurate estimate of the period is available. Comparison of KLD and L1D shows that with upto a 30dB difference in performance under certain circumstances, KLD is a significantly better choice.

The scheme is also easy to modify into a boundary detector capable of estimating crucial signal timing data such as the period, as required by the detector. The ability to detect spectral abnormalities and temporal boundaries has numerous applications even outside the area of CR such as the field of test & measurement (T&M). The proposed method can be used to detect and identify primary systems, to predict interference patterns, to synchronise frames, and to debug complex wireless communication systems - to name only a few.

Open issues that are subject to further research are: performance evaluation against a wider range of test signals, applicability to the use cases outlined above and the development of an autonomous detector combining the base algorithm with boundary detection.

Acknowledgement

This project is in collaboration with Agilent Technologies (Edinburgh/Queensferry, UK). In particular, we would like to thank Roy Mcnaughton and Peter Cain for the provision of test data. We greatly acknowledge the financial support of this project from the Agilent University Relations Program.

References

[1] J. Mitola III, "Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio," Ph.D. dissertation, Royal Institute of Technology (KTH), May 2000.

[2] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[3] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed., ser. Wiley Series in Telecommunications, D. L. Schilling, Ed. John Wiley & Sons, Sep. 1991.

[4] M. Basseville, "Distance Measures for Signal Processing and Pattern Recognition," *Signal Processing*, vol. 18, no. 4, pp. 349–369, Dec. 1989.

[5] M. Afgani, S. Sinanović, and H. Haas, "Anomaly Detection Using the Kullback-Leibler Divergence Metric," in *Proc. of the First International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL)*, Aalborg, Denmark, Oct. 25–28, 2008.

[6] L. Cohen, *Time-Frequency Analysis: Theory and Applications*, 1st ed., ser. Prentice Hall Signal Processing Series, A. V. Oppenheim, Ed. Prentice Hall, Jan. 1995.

[7] D. Thomson, "Spectrum Estimation and Harmonic Analysis," *Proc. IEEE*, vol. 70, no. 9, pp. 1055–1096, Sep. 1982.

[8] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing: Principles, Algorithms and Application*, 4th ed., T. Robbins, Ed. Prentice Hall, Apr. 2006.

[9] H. Shimazaki and S. Shinomoto, "A Method for Selecting the Bin Size of a Time Histogram," *Neural Computation*, vol. 19, no. 6, pp. 1503–1527, Jun. 2007.

[10] R. Krichevsky and V. Trofimov, "The Performance of Universal Encoding," *IEEE Transactions on Information Theory*, vol. 27, no. 2, pp. 199–207, Mar. 1981.

Anomaly Detection Using the Kullback-Leibler Divergence Metric

Mostafa Afgani, Sinan Sinanović and Harald Haas
Institute for Digital Communications,
The University of Edinburgh, Edinburgh EH9 3JL, UK
Email: {m.afgani, s.sinanovic,h.haas}@ed.ac.uk

***Abstract*—A method of detecting changes or anomalies in periodic information-carrying signals or any other sets of data using Kullback-Leibler divergence is described. Theoretical reasons for using this information-theoretic approach are briefly outlined and followed by its detailed application on disturbance/anomaly detection in wireless signals. Even though the concept is illustrated in a communications centric framework, it is more generally applicable in areas such as computational neuroscience, mathematical finance and others where it is important to statistically detect unexpected signal distortions. The results obtained show that the proposed approach is robust, highly effective, and has a low implementation complexity.**

I. Introduction

Random signals are ubiquitous in communications, neuroscience, economics and many other areas of science. In the field of communications, the traditional method of detecting the desired signal which is mixed with additive Gaussian noise is via matched filter. However, if during the communication, some sudden, anomalous event such as short-term power surge or irregular interferer appears, it is beneficial to detect this occurrence. This is especially true of cognitive radio systems that aim to intelligently avoid interference from secondary radio sources to provide the best service possible to the primary user [1]. Because of the unpredictable and destructive nature of the anomaly, demodulation of the signal is often infeasible. It is also generally unnecessary since the signal envelope itself can provide sufficient information regarding the link quality. As a result, a statistical approach to the problem of anomaly detection can provide a better and computationally simpler solution.

The Kullback-Leibler divergence (KLD) measure of the difference between two statistical distributions is ideally suited to the problem. It is a simple and versatile tool that reduces the often complex task of describing the statistical difference between two data sets to a single positive real number. Due to such qualities, it finds widespread use in diverse fields ranging from neuroscience to economics. In [2] it is used for image recognition, while in [3], the same metric is utilised for analysing neural firing patters. It also finds use in economics where it is used to develop models for financial forecasting [4].

It is possible to model any discrete process or data set by means of a probability mass function (PMF). Given anomaly-free data samples, an empirical PMF characterising "normal" behaviour can be estimated by means of a histogram. Then, subsequent PMF estimates of the data under observation can be compared with the "normal" by means of the KLD to establish the presence of an anomaly. Once the PMF estimates are available, little computational effort is required to evaluate the KLD as it only requires a number of multiplications, divisions, additions and logarithm operations proportional to the number of histogram bins utilised. Due to the low computational overhead, real-time implementation and integration into existing devices are feasible.

A method of employing the KLD for anomaly detection in periodic signals and other data sets is presented in Section II. The performance of the algorithm is then evaluated against a set of three real (physically recorded) radio frequency (RF) signals containing different types of anomalies. The findings are presented and discussed in Section III and Section IV concludes the paper.

II. Anomaly Detection

A. Kullback-Leibler Divergence

The natural statistical measure of difference between two probability distributions ($p(x)$ and $q(x)$) [5] is Kullback-Leibler divergence (KLD) and it is defined as:

$$D(p \parallel q) = \sum_{x \in X} p(x) \log_2 \frac{p(x)}{q(x)}, \tag{1}$$

where X is the set of all possible event outcomes. As base-2 logarithm is used, KLD can, thus, be measured in bits. It is, however, not considered a true distance metric as the divergence between two PMFs is generally not symmetric (i.e. $D(p \parallel q) \neq D(q \parallel p)$) and the triangle inequality is not satisfied. $D(p \parallel q) = D(q \parallel p) = 0$ if and only if $p(x)$ and $q(x)$ are identical; otherwise the KLD is always a positive, non-zero number. It belongs to a class of distance measures known as *f-divergence* (or *Ali-Silvey distances*) [6]. Furthermore, it has a number of desirable theoretical properties. Through Stein's Lemma, KLD is equal to the exponential rate of the optimal Neyman-Pearson classifier which is important result in detection theory. Furthermore, because KLD is proportional to the Fisher information for perturbational changes in the PMF parameter, it is obviously related to the fundamental Cramér-Rao bound on mean-square estimation error.

The general approach to quantise the difference between two sets of data using KLD is as follows:

1) Identify the data sets that are to be compared. For a time-series signal with periodic features, it is logical to

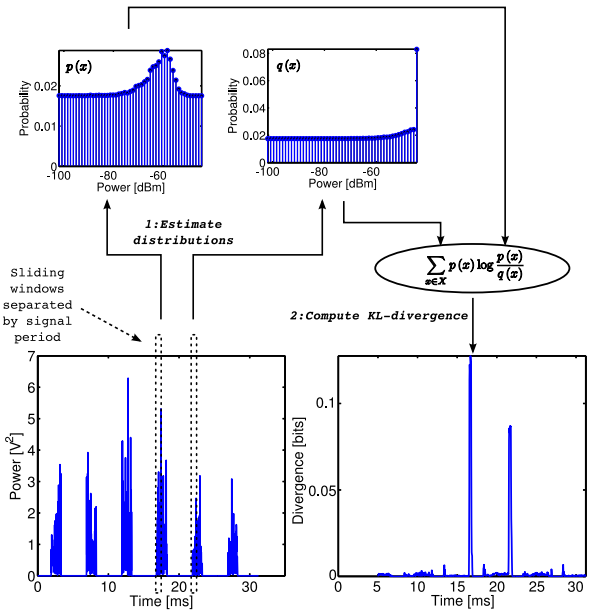


Fig. 1. KLD analysis procedure for a periodic signal. The windows compare analogous portions of neighbouring signal frames. Any discrepancy between statistical distributions in the windows leads to peaks in the KLD plot. Each anomaly generally gives rise to two peaks (separated by the signal period) in the divergence – this is due to the twin windowing technique utilised.

compare segments separated by the feature period since similar probability distributions are expected in both.

2) If the data is obtained from a continuous process, it is necessary to set a limit on the number of samples used to construct each set. For the periodic time-series case, this amounts to *windowing* the segments separated by the implicit period.
3) Once the data sets are available, the samples are processed to extract the feature targeted. The same feature definition must be employed for both groups of samples. Given data from a complex time-series, possible features are magnitude and phase.
4) Estimate the PMF of each data set over the feature chosen and computed the associated KLD using (1).
5) If the KLD obtained is larger than some predefined threshold, the difference between the data sets is significant and an anomaly may be present.

Fig. 1 illustrates how the aforementioned approach can be used to detect anomalies in a signal utilising time division duplexing (TDD). TDD implies that the signal has a frame structure that repeats with a period of T_S. Two sliding windows of length T_w separated by T_S are employed to collect the feature (square of the magnitude) distribution statistics. The PMF estimated in each window is denoted by $p(x)$ and $q(x)$ respectively and used to compute the instantaneous KLD. The procedure is then repeated by advancing the windows to the next data sets.

It is clear that the analysis can be performed on the signal envelope itself and no demodulation is necessary. This is a significant advantage since no knowledge of the underlying system parameters except the frame period, T_S, is required. Since the metric reacts directly to changes in the signal envelope, any peaks in the instantaneous KLD can be used to trigger a higher level analysis of that segment of the signal or a detector. Besides being insensitive to the values coded in the underlying signal, the algorithm reduces the complexity of an unknown anomaly to a scalar value which is the divergence from the true PMF.

B. PMF Estimation

From (1) it is obvious that the success of the procedure in detecting anomalies is directly affected by the quality of the PMF estimates which are in turn affected by the parameters used to construct the data feature histograms. The parameters in question are the sample size and the bins utilised.

The role of the sample size, Σ, in constructing histograms is straightforward – for a given number of bins, too few samples lead to an inadequate description of the the underlying distribution. Generally, larger values of Σ produce better results with diminishing returns. Section III-E shows how the performance of the proposed scheme is affected by the amount of data considered.

The choice of the number of histogram bins, N, and the bin-width, Δ, are also crucial for accurately estimating a given distribution. For a given value of Σ, too few bins may not adequately capture the subtle variations present in multi-modal distributions; too many will yield a histogram that is excessively noisy. Shimazaki and Shinomoto show in [7] how the "optimum" Δ, according to some minimum error criterion and for certain classes of random variables, can be calculated for a given data set. Although the method is simple and effective, it is too costly to implement in real-time as it requires the minimisation of a cost function through exhaustive search. At this point, it is important to note that an accurate representation of the underlying PMFs is not absolutely necessary for the purpose of anomaly detection using KLD. Since the proposed method performs a *relative* comparison between PMFs, estimates utilising any reasonable number of bins should be sufficient. The impact of the value of N on the performance of the algorithm is explored in Section III-F.

Empty bins in histograms result in zeros in estimated PMFs. This can lead to undefined behaviour when divisions with zeros are encountered while evaluating the KLD using (1). This apparent problem can be mitigated by *preloading* the histogram bins by a small number – such that there can be no empty bins in the constructed histograms. In [8] Krichevsky and Trofimov argue that 0.5 is a good choice.

III. Signal Analysis Results

In order to evaluate the performance of the proposed anomaly detection scheme, three distinct test signal are employed. The signals are real RF transmissions captured using spectrum analysis hardware. The effect of the amount of data, sampling rate, and the number of histogram bins on the performance of the algorithm is also investigated and the results are presented.

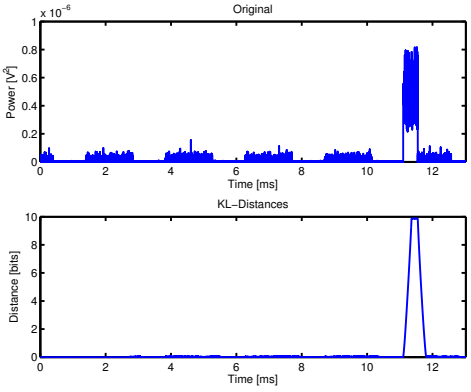


Fig. 2. *Top*: Wireless LAN signal with interference from a Bluetooth device. *Bottom*: KLD analysis of the signal readily reveals the presence of the interferer.

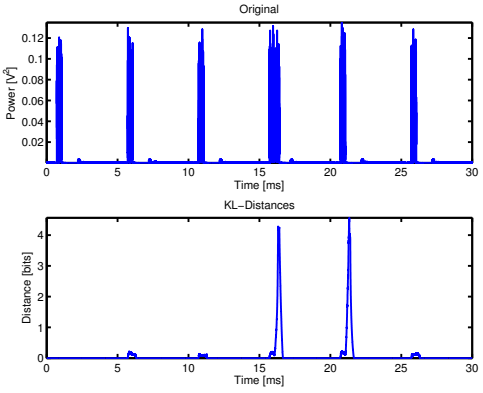


Fig. 3. *Top*: Wireless broadband signal. The uplink sub-frame at 16ms has a longer duration than others and is therefore unusual. *Bottom*: KLD analysis reveals the anomalous segment of the signal. A second peak is obtained when the signal returns to normal – this is due to the twin-windowing nature of the procedure.

For comparative purposes, performance is quantised using the KLD Ratio (KLDR) metric introduced in [9]. It is essentially the ratio between the KLD value of the anomaly and the maximum of the background KLD associated with normal signal segments:

$$\mathrm{KLDR} = \frac{\mathrm{KLD}_{\mathrm{anom}}}{\mathrm{KLD}_{\mathrm{bg}}}. \quad (2)$$

A. Test Signal A

Fig. 2 shows a signal that encounters a sudden strong burst of interference from a secondary source. The underlying periodic signal consists of a single Wireless LAN (WLAN) frame that is transmitted repeatedly with a period of 2.45ms. The interference observed at 11.5ms is a single burst of data from a Bluetooth (BT) device.

In order to perform KLD analysis of the signal, two windows with a duration of 256μs are employed. The window centres are separated by 2.45ms to match the WLAN frame repetition interval. The 20MHz signal is sampled at a rate of 2MHz and the number of histogram bins utilised is 32. As shown in a subsequent section (III-F), the number of histogram bins used does not have a significant impact on the outcome. Therefore, a bin count of 32 is simply an arbitrary but reasonable choice.

The result of the KLD analysis is also shown in Fig. 2. The presence of the anomaly (BT interferer) is immediately revealed. Even though the signal is under-sampled by a factor of 10, the algorithm is clearly successful with a large KLDR of 42.63 dB. The effect of the sampling rate on performance is further explored in Section III-D.

B. Test Signal B

Fig. 3 shows a wireless broadband (WiBro) signal. Due to the proximity of the recording equipment to the mobile terminal (MT), the uplink (UL) sub-frames show a higher power level then downlink (DL) sub-frames. From the plot, it can be seen that the UL sub-frame at 16ms is longer than any of the other UL sub-frames. In context of this particular signal

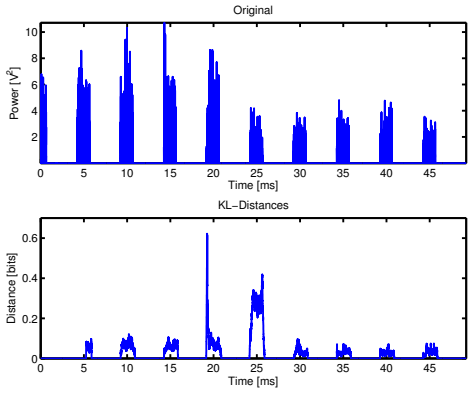


Fig. 4. *Top*: Wireless broadband signal. An extra command sequence in the uplink sub-frame at 20ms initiates the power control loop. *Bottom*: The extra command sequence (20ms) and power change (25ms) are both revealed by KLD analysis of the signal.

snapshot, this behaviour is unusual and hence anomalous. Once again, two windows with a duration of 256μs each are employed. The window centres are separated by 5ms – corresponding to the signal frame period. 32 histogram bins are utilised for the PMF estimates and the 10MHz signal is sampled at 1MHz.

KLD analysis of the signal is also shown in Fig. 3. A sharp peak in the KLD at 16ms reveals the presence of the unusual UL sub-frame. A second peak is obtained when the signal returns to normal in the following UL sub-frame. The KLDR observed is 26.62 dB.

C. Test Signal C

The final test signal is shown in Fig. 4. It depicts the message exchange between a mobile terminal and base station using the WiBro communication standard. Since the recording is made at the MT, there is significantly more power in the UL

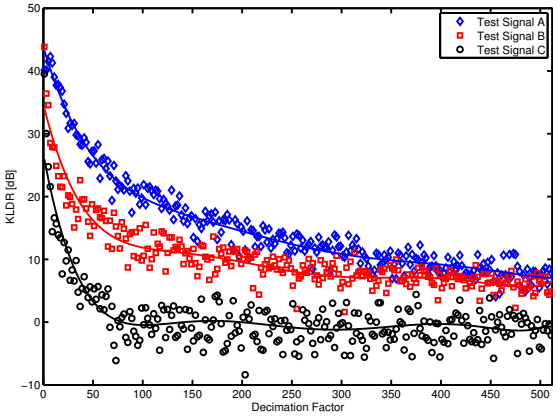


Fig. 5. Performance of the KLD based anomaly detection scheme under various data sampling rates. Decimation factor refers to the amount by which the input signal is under-sampled relative to the signal bandwidth. A factor of unity corresponds to a sampling frequency equal to the signal bandwidth. For two of the test cases, a KLDR well above 10 dB can be maintained for a window length of 256μs, 32 histogram bins, and a decimation factor of 100. The third signal is unable to accommodate high decimation rates due to the relatively short duration of the first anomaly and a clear distinction in the power levels of the second anomaly.

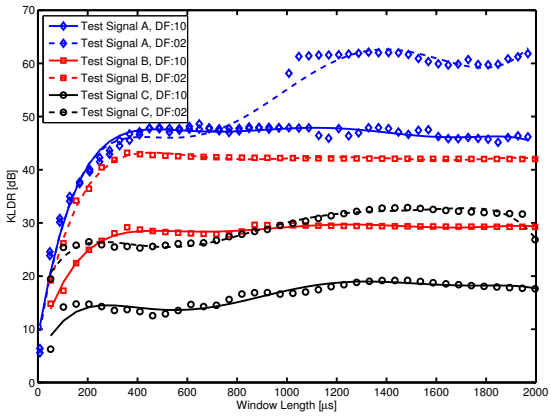


Fig. 6. Performance of the KLD based anomaly detection scheme under various window lengths and hence the input sample size. Decimation factors (DF) of 2 and 10 are utilised and the number of histogram bins is 32. In all cases, an increase in the amount of data leads to a better performance. However, beyond a certain window length, the performance is no longer strongly affected.

sub-frames. Although unnoticeable in the time-series, the UL sub-frame at 20ms contains an additional command sequence that triggers the subsequent change in the transmit power observed at 25ms. As a result, there are effectively two unusual events in the signal. The parameters utilised for KLD analysis of the signal are identical to those in Section III-B.

The KLD plot in Fig. 4 shows that both anomalies are detected. Since the width of a KLD peak corresponds to the temporal duration of the anomaly responsible, the first peak at 20ms is very sharp as it is due to the extra command sequence in the UL sub-frame. Since the subsequent change in power at 25ms affects the entire UL sub-frame, the second KLD peak is much broader and spans the entire sub-frame. The KLDR in this case is 13.68 dB.

D. Sampling Rate

Continuous processes such as time-series must be sampled before the anomaly detection algorithm can be applied. The sampling frequency employed is crucial as it directly influences the complexity of the scheme. For a given window length, a higher frequency implies that more data samples have to be stored and sorted to construct the histograms. If the frequency is too low, small scale signal features and anomalies may be overlooked. According to the Nyquist sampling criterion, a signal must be sampled with a frequency at least twice as large as its bandwidth to be reconstructible. For wideband signals this leads to a very high sampling frequency and hence a prohibitively large volume of data – heavily increasing the resource requirements of the proposed scheme. However, since the algorithm does not require the time-series to be reconstructible, a far lower sampling frequency can be used. Fig. 5 shows how the performance of the scheme is affected by under-sampling of the data. The window length utilised is 256μs and the histograms are 32 bins wide. The amount by which the input time-series is under-sampled relative to the bandwidth is defined as the decimation factor. Therefore, a factor of unity implies that the signal is sampled at the same frequency as the signal bandwidth.

The results indicate that decimation factors as large as 500 can be successfully employed depending on the type and duration of the anomaly present. For test signals A and B, a KLDR of more than 10 dB can be maintained even with a decimation factor of 100. This is an important result as it indicates that satisfactory performance levels can be maintained with little input data and hence low complexity implementations of the algorithm. The scheme is able to work with such small amounts of data due to the fact that the KLD performs a *relative* comparison between the PMFs. At high decimation factors, test signal C fails. This is because the first anomaly is temporally brief and is likely to be missed. As for the second anomaly in the signal, too few samples are used to characterise the PMF and hence the difference is not large enough.

E. Effect of Window Length

The effect of the window size and hence the amount of data utilised in constructing PMF estimates is shown in Fig. 6. The number of histogram bins utilised is 32 and the input time-series is under-sampled by factors of 2 and 10. At lower decimation factors, more data is available and the KLDR improves uniformly across all window sizes for signals B & C. This is the behaviour expected from Fig. 5. At smaller window sizes, performance for signal A is unaffected by the choice of the decimation factor due to the relatively long duration of the anomaly. This is because even at a decimation factor of 10, a sufficient number of anomalous samples are represented in the PMF.

As anticipated, the performance is poor at small window sizes where the amount of data available is insufficient. Increasing the window length leads to an improvement of the

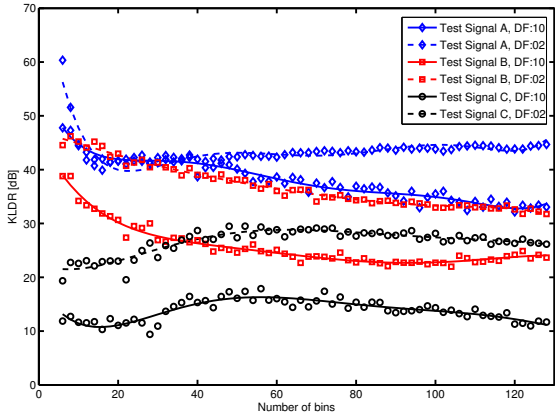


Fig. 7. Performance of the KLD based anomaly detection scheme under various histogram bin counts. $N-2$ of the bins are distributed equally between the maximum ($P_{\max}$) and minimum ($P_{\min}$) power levels encountered in the training data. The other two account for samples that fall between $(-\infty, P_{\min})$ or $(P_{\max}, \infty)$. The window size is set at 256μs and the decimation factors used are 2 and 10. Generally, a larger number of bins lead to poorer performance due to increased noise in the estimates. However, the rate of change is small and therefore the drop in performance is insignificant over a wide range of bin numbers.

performance. However, for signals B & C, the gains become marginal for windows larger than approximately 400μs. The sudden transition shown by signal A at a window length of 1ms for a decimation factor of 2 is attributed to a reduction in the background KLD at the frame edges. This is due to the boundary detecting capability of the algorithm [9].

The initial gain is due to improvements in the PMF estimates which lead to a reduction in the background KLD levels. At larger window sizes, the anomalous samples represent smaller fractions of the data and hence contribute less to the shape of the estimated PMF – resulting in a decrease of the KLD due to the anomaly. As the background levels are also reduced by an increase in the data size, the overall KLD ratio (i.e. the KLDR) remains relatively constant.

F. Effect of Histogram Bin Count

Fig. 7 shows how performance is affected by the choice of the number of histogram bins used to classify the input data and estimate the PMF. The window length is set at 256μs and the decimation factors utilised are 2 and 10. The behaviour observed and hence the analysis for lower decimation factors is identical to that in Section III-E.

Of the N bins, $N-2$ are distributed equally between the maximum ($P_{\max}$) and minimum ($P_{\min}$) power levels encountered in some anomaly-free training data. The remaining two are used to handle samples that fall between $(-\infty, P_{\min})$ or $(P_{\max}, \infty)$.

Although the general trend observed in the plot is that the performance degrades slowly with increasing bin numbers, test signal C appears to have an optimum in the vicinity of 55. This indicates that the behaviour observed is specific to the type of anomaly present in a signal. While the number of bins utilised does not appear to have a significant impact on the performance of the scheme for a fixed amount of data, the decrease observed is due to noisier PMF estimates that are obtained for larger bin counts. Noisy PMFs lead to larger background KLD values and hence a reduced KLDR.

IV. Conclusion

A simple, yet highly effective and robust method of detecting anomalies in periodic signals and general data sets by means of Kullback-Leibler divergence is presented. Extensive analysis of three distinct wireless test signals captured from the real world show that the scheme is successful at detecting all anomalies present under a wide range of conditions. Furthermore, it is possible to obtain satisfactory performance even when the input is severely under-sampled. This allows for low complexity implementations capable of monitoring wideband signals in real-time.

Although the algorithm is explored in context of wireless communications, it is envisioned that the approach described can be readily generalised to tackle a wide range of problems in fields ranging from computational neuroscience to mathematical finance where detection of hidden or otherwise unusual features are of interest. Application of the technique to such diverse problems will be the subject of future research.

Acknowledgement

This project is in collaboration with Agilent Technologies (Edinburgh/Queensferry, UK). In particular, we would like to thank Roy Macnaughton and Peter Cain for the provision of test data. We greatly acknowledge the financial support of this project from the Agilent University Relations Program.

References

[1] J. Mitola III, "Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio," Ph.D. dissertation, Royal Institute of Technology (KTH), May 2000.

[2] C. Liu and H.-Y. Shum, "Kullback-Leibler Boosting," in *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. Madison, Wisconsin, USA: IEEE, Jun. 16–22, 2003, pp. I–587–I–594.

[3] H. Nakahara and S.-I. Amari, "Information-Geometric Measure for Neural Spikes," *Neural Computation*, vol. 14, no. 10, pp. 2269–2316, Oct. 2002.

[4] J. Robertson, E. W. Tallman, and C. H. Whiteman, "Forecasting Using Relative Entropy," Nov. 2002, FRB of Atlanta Working Paper No. 2002-22. [Online]. Available: http://ssrn.com/abstract=355460

[5] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 1st ed., ser. Wiley Series in Telecommunications, D. L. Schilling, Ed. John Wiley & Sons, Sep. 1991.

[6] M. Basseville, "Distance Measures for Signal Processing and Pattern Recognition," *Signal Processing*, vol. 18, no. 4, pp. 349–369, Dec. 1989.

[7] H. Shimazaki and S. Shinomoto, "A Method for Selecting the Bin Size of a Time Histogram," *Neural Computation*, vol. 19, no. 6, pp. 1503–1527, Jun. 2007.

[8] R. Krichevsky and V. Trofimov, "The Performance of Universal Encoding," *IEEE Transactions on Information Theory*, vol. 27, no. 2, pp. 199–207, Mar. 1981.

[9] M. Afgani, S. Sinanović, and H. Haas, "Information Theoretic Approach to Signal Feature Detection for Cognitive Radio," in *Proc. of the Global Telecommunications Conference (GLOBECOM)*. New Orleans, USA: IEEE, Nov. 30–Dec. 4, 2008.

Speed Estimation Using Relative Radio Frequency Signature Matching

Mostafa Afgani and Harald Haas
Jacobs University Bremen,
School of Engineering & Science,
28759 Bremen, Germany,
Email: {m.afgani,h.haas}@jacobs-university.de

***Abstract*—One of the largest obstacles to effective navigation and positioning using pedestrian dead reckoning is the lack of accurate speed estimation algorithms. Existing methods are either complex or provide results that are unsatisfactory at low velocities associated with pedestrians and other slow moving entities. In contrast, the algorithm proposed in this paper is simple to implement and is able to provide accurate results at low velocities. A one-dimensional and unidirectional two-antenna solution is described where the speed can be easily estimated from a knowledge of the fixed inter-antenna distance and the time it takes for the trailing antenna to experience the same channel conditions (radio frequency (RF) signature) as the leading antenna. Computer simulations of the new adaptive algorithm show that with estimation errors of less than 1.2% around average pedestrian speeds the approach is indeed effective and accurate.**

I. Introduction

Pedestrian dead reckoning (PDR) is a popular choice for positioning and navigation in areas (e.g. indoors) where Global Positioning System (GPS) based solutions cannot be used. Two pieces of information are essential before a valid location estimation can be made: a reference point with known coordinates and the velocity (speed and heading) at sufficiently close and successive intervals. Given the needed information, displacement of the user from the reference location can be approximated and hence an estimate of the new location coordinates can be obtained.

Accurate heading information is readily available from sensors such as a ring laser gyro [1]; it is the speed estimate that has been difficult to obtain with a sufficient degree of accuracy. Step length estimation based devices ([2], [3]) can only be used by pedestrians and must be mounted directly on the person. Frequent updates of the absolute position are also required as errors tend to accumulate with every step. While methods based on the level crossing rate (LCR) of the received Rayleigh fading envelope ([4], [5]) typically provide good results for high speeds, the accuracy drops considerably at speeds associated with pedestrians and other low velocity entities. Continuous wavelet transform (CWT) can also be used to extract speed information from the aforementioned envelope with a good degree of accuracy [6], however such a procedure is computationally complex and expensive.

In this paper, a novel method of speed estimation using relative RF signature matching is described. The proposed method correlates the RF signatures at two antennae separated by a known distance to determine the time it takes for the trailing antenna to experience the same channel conditions as that experienced by the leading antenna. As the antenna separation is predefined and known (e.g. in a MIMO (multiple-input-multiple-output) device), the speed is easily calculated from an estimate of the time delay. As will be shown later, the new speed estimation approach also employs an adaptive algorithm that allows accurate estimates at both high and low velocities. Furthermore, it is of low computational complexity as the main operation only involves the calculation of correlation values between two channel estimates.

The rest of this paper is organized as follows. In Section II the wireless channel model considered is presented and in Section III the speed estimation algorithm is described. Section IV provides details of the realization and the simulation results. Section V concludes the paper.

II. Wireless Channel Model

The multipath effect is a common phenomenon in typical terrestrial environments. In addition to frequency selective fading, signal frequencies can also experience spreading caused by Doppler shifts. The signal received from a multipath channel by an antenna array can be represented by

$$\overrightarrow{h}(t) = \sum_{l=0}^{L(t)-1} A_l(t) \exp^{j\phi_l(t)} \overrightarrow{a}(\theta_l(t)) \delta(t - \tau_l(t)) \quad (1)$$

where $L(t)$ is the number of multipath components, A_l is the amplitude, ϕ_l is the carrier phase shift, τ_l is the time delay of the l^{th} multipath, and θ_l is the angle of arrival (AoA) of the l^{th} multipath. The amplitude is modeled as a Rayleigh distributed random variable and the AoA and phase shift are uniformly distributed [7]. Furthermore, the delay is exponentially distributed [8]. $\overrightarrow{a}(\theta_l(t))$ is known as the array response vector. When the signal and antenna array (containing m antennae) are restricted to a two-dimensional space, the array response vector is given by

$$\overrightarrow{a}(\theta_l(t)) = \begin{bmatrix} \exp(-j\Psi_{l,1}) \\ \exp(-j\Psi_{l,2}) \\ \exp(-j\Psi_{l,3}) \\ \cdots \\ \exp(-j\Psi_{l,m}) \end{bmatrix} \quad (2)$$

where $\Psi_{l,i}(t) = [x_i \cos(\theta_l(t)) + y_i \sin(\theta_l(t))] \cdot \beta$ and $\beta = \frac{2\pi}{\lambda}$ is the wavenumber [7]. λ is the carrier wavelength.

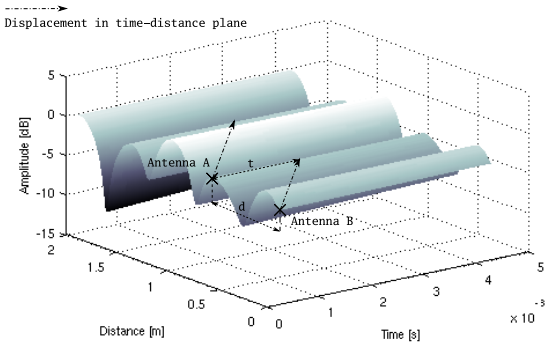


Fig. 1. Multipath channel in time and space simulated for a carrier frequency of 474MHz (DVB-T) and a speed of 1.5ms^{-1}. After t seconds, *antenna B* is where *antenna A* was and experiences the same channel response that *antenna A* experienced t seconds ago. As the distance, d, between the antennae is known, the speed is easily estimated as $v = \frac{d}{t}$.

The maximum Doppler shift, $f_{d(\max)}$, experienced by a receiver is dependent on the speed and is given by

$$f_{d(\max)} = \frac{v}{\lambda} \tag{3}$$

where v is the speed [9]. The presence of Doppler spread in a multipath channel causes it to display variations in time: the higher the Doppler frequency, the shorter is the coherence time [9],

$$T_c = \frac{0.423}{f_{d(\max)}}, \tag{4}$$

of the channel. Fortunately, at velocities typical of pedestrians (approx. 1.5ms^{-1}), the resultant Doppler frequency is small (1.6Hz for a carrier frequency of 474MHz) and hence the coherence time of the channel is quite large (approximately 0.26s). Fig. 1 shows the space-time characteristics of a multipath Rayleigh fading channel with a low Doppler spread.

It is clear from the plot that although the channel stays nearly constant over time, it shows rapid variations in space – the "coherence distance" is on the order of some tens of centimeters. The behavior displayed by the channel, in terms of coherence time and "coherence distance", is exactly as needed for the proposed speed estimation technique.

III. RF Signature Matching Algorithm

As stated in the previous section, a channel response that shows rapid variations in space but remains relatively unchanged in time is essential to the success of the speed estimation algorithm. It requires a multi-antenna setup and an accessible RF source such as a local digital television transmitter. For each spatial dimension, at least two antennae are required. For simplicity, let movement be confined to a single spatial dimension only (the idea applies to the general three dimensional case in a straight-forward manner) and let the antenna array be aligned parallel to the direction of motion as shown in Fig. 2.

At time $t = 0$s, the antennae are as shown by the markers in Fig. 1. As the array moves forward, it traces a diagonal

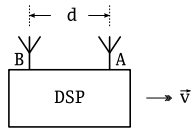


Fig. 2. Two-antenna speed estimator setup.

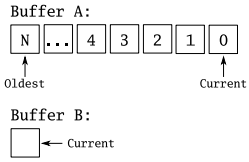


Fig. 3. Buffers to hold channel response estimates for *antenna A* and *antenna B*.

line in the space-time plane and after some t seconds, the trailing antenna (*antenna B*) will be at the very same point in space that the leading antenna (*antenna A*) occupied t seconds ago. As the coherence time of the channel is quite large, *antenna B* should experience a channel response that is very similar to that experienced by *antenna A* t seconds ago – i.e. the two responses would be highly correlated. Therefore, the time delay can be estimated by correlating the channel responses experienced by *antenna B* with a number of previous channel responses experienced by *antenna A*. As the antenna separation, d, is predefined and known, the speed is easily calculated using:

$$v = \frac{d}{t}, \tag{5}$$

once the time delay, t, has been determined. Note that knowledge of the fading process or even an accurate estimate of the channel is not necessary for this technique. As the algorithm compares the relative match between two channel estimates, any error or uncertainty in the estimates can be ignored as long as both estimates are affected similarly by the error.

Obviously, a database of channel responses for *antenna A* must be maintained. The size of the database will dictate the accuracy of the estimation algorithm. For the simple case where the movement is confined to a single spatial dimension and in the forward direction only, a database capable of holding N previous and the current channel response estimate (CRE) from *antenna A* will be required. The buffer associated with *antenna B* only needs to hold the instantaneous CRE as depicted in Fig. 3. In addition to the CRE database, a database of associated timestamps will also have to be kept.

At every instant of the speed estimation process, the current CRE at *antenna B* is cross-correlated against the CREs at *antenna A* to determine lag. Therefore, the detection process is:

$$\text{lag} = \underset{k}{\text{argmax}} \left(B \star A_k \right);\ k = 1, \ldots, N. \tag{6}$$

The $\star$-operator represents the cross-correlation at $\text{delay} = 0$. The cross-correlation product $B \star A_0$ is not considered since it corresponds to the lag being zero – implying infinite speed.

As the cross-correlation at $\text{delay} = 0$ is nothing more than the *dot-product* between the two vectors B and A_k, the entire

procedure simplifies to a series of normalized dot-product calculations followed by a maximum detection. The simplified detection rule is then

$$\text{lag} = \underset{k}{\text{argmax}} \left(\frac{B \cdot A_k}{||B|| \cdot ||A_k||} \right);\ k = 1, \ldots, N. \quad (7)$$

Once the lag has been determined, it can be used to look up the associated timestamp – allowing for the actual time lapse to be calculated.

In order to obtain further system parameters, certain physical constraints must be defined. Assuming that the average walking speed of an adult human is between $1 - 1.5\text{ms}^{-1}$, lower and upper speed detection limits of $v_{\min} = 0.1\text{ms}^{-1}$ and $v_{\max} = 15\text{ms}^{-1}$, respectively, are proposed. A speed of zero can be deduced from a lack of significant correlation in the CREs at *antenna A* and *antenna B*. With a given antenna separation distance, the shortest delay detectable (i.e. a lag of 1) is $\left(\frac{d}{15}\right)$. The longest delay that needs to be accommodated is then given by $\left(\frac{d}{0.1}\right)$. Therefore the minimum required size for the database is

$$\left\lceil \frac{\left(\frac{d}{0.1}\right)}{\left(\frac{d}{15}\right)} \right\rceil + 1 = 151, \quad (8)$$

i.e. $N = 150$. The one additional slot is needed to store the current CRE. At this point, the needed estimation frequency can be defined. Assume a DVB-T transmitter operating in the 2K mode with 8MHz channels is available as the RF source. The duration of each OFDM (orthogonal frequency division multiplexing) symbol, T_s, is then 224μs [10]. The *estimation frequency*, $\hat{i}$ (number of OFDM symbols that elapse between estimates) is defined as

$$\hat{i} = \left\lfloor \left(\frac{d}{v_{\max}} \times \frac{1}{T_s} \right) \times \frac{1}{\mu} \right\rfloor \quad (9)$$

where μ is an adaptive scaling parameter that has been introduced to improve the accuracy of the estimates. $\frac{d}{v_{\max}}$ provides the maximum time that can elapse between two estimates if the upper speed limit of $v_{\max}$ is to be detected; therefore, dividing that quantity by T_s and rounding it off to the lower integer yields the maximum number of complete OFDM symbols that can be allowed to pass between the two estimation instants.

μ is a real number that is upper bounded by $\left(\frac{d}{v_{\max}} \times \frac{1}{T_s}\right)$; this is because estimates cannot be obtained at a rate faster than for every OFDM symbol ($\hat{i} = 1$). It is lower bounded by 1 since we are not interested in detecting speeds that are below $v_{\min}$. Starting with an initial value of 1, the rough speed estimates, v_{est}, are used to find the new value of μ to be used. Linear interpolation is used for this purpose:

$$\mu = \begin{cases} m \cdot (v_{\text{est}} - v_{\min}) + 1, & v_{\min} \le v_{\text{est}} \le (v_{\max} + \epsilon) \\ 1, & \text{otherwise} \end{cases} \quad (10)$$

where

$$m = \frac{\left[\left(\frac{d}{v_{\max}} \times \frac{1}{T_s}\right) - 1\right]}{(v_{\max} - v_{\min})} \quad (11)$$

and ϵ is a small number that is needed to allow the adaptive procedure to converge at the upper end of speed detection range. Its is governed by the absolute difference between $v_{\max}$ and the highest speed detectable when $\mu = 1$:

$$\epsilon = \left| v_{\max} - \left(\frac{d}{T_s \times \left\lfloor \frac{d}{v_{\max}} \times \frac{1}{T_s} \right\rfloor} \right) \right|. \quad (12)$$

It should be clear from (10) that an increased speed estimate results in a higher value of μ which then causes the estimation frequency (9) to increase – allowing a finer granularity in the time offset estimates. As a result of the finer granularity, the resolution at high speeds is increased*. A lower speed results in a drop in the value of μ and causes the estimation frequency to decrease – allowing the accurate detection of lower speeds. Table I shows the effect of μ on the speeds that can be detected for $d = 0.05$m. $v_{\max}$, $v_{\min}$, and T_s are as described earlier. The speed is calculated from the lag, k, using

$$v_{\text{est}} = \frac{d}{k \times T_s \times \hat{i}}. \quad (13)$$

TABLE I
THE EFFECT OF μ ON SPEEDS DETECTABLE.

$\mu = \mathbf{1}$		$\mu = \mathbf{14}$	
Lag	**Speed (m/s)**	**Lag**	**Speed (m/s)**
1	15.944	1	223.21
2	7.9719	...	...
3	5.3146	14	15.944
4	3.9860	15	14.881
...	...	16	13.951
148	0.10773	...	...
149	0.10701	149	1.4981
150	0.10629	150	1.4881

Table I clearly shows that with $\mu = 1$, estimation is possible over the entire range – although the resolution at the high end is very poor. With $\mu = 14$, the resolution of the estimates at the high end is improved considerably at the expense of a reduced range – speeds lower than 1.48ms^{-1} can no longer be detected.

It can also be seen that with $\mu = 1$ (the starting condition), a true speed of 15ms^{-1} is most likely to be estimated as 15.944ms^{-1}. At the absence of the parameter ϵ, μ will always remain at that initial value (since the condition $v_{\text{est}} \le (v_{\max} + \epsilon)$ in (10) would not be satisfied) and hence the speed will continue to be estimated as 15.944ms^{-1}. With $\epsilon = 1$, however, that condition will be satisfied and μ will be increased accordingly – increasing the estimation frequency and yielding an estimate that is more precise than the last.

Table I also shows that the speed estimates available over the entire detection range is not continuous. This is due to the fact that the time offset estimates are always multiples of T_s and hence are themselves not continuous.

*At a distance $d = 0.05$m, a "granularity" of 0.001s would only allow the detection of 10ms^{-1} and 8.33ms^{-1} corresponding to $t = 0.005$s and $t = 0.006$s respectively. However, with a finer granularity of 0.0005s, a speed of 9.09ms^{-1} can also be detected (corresponding to $t = 0.0055$s) – i.e., the resolution is improved.

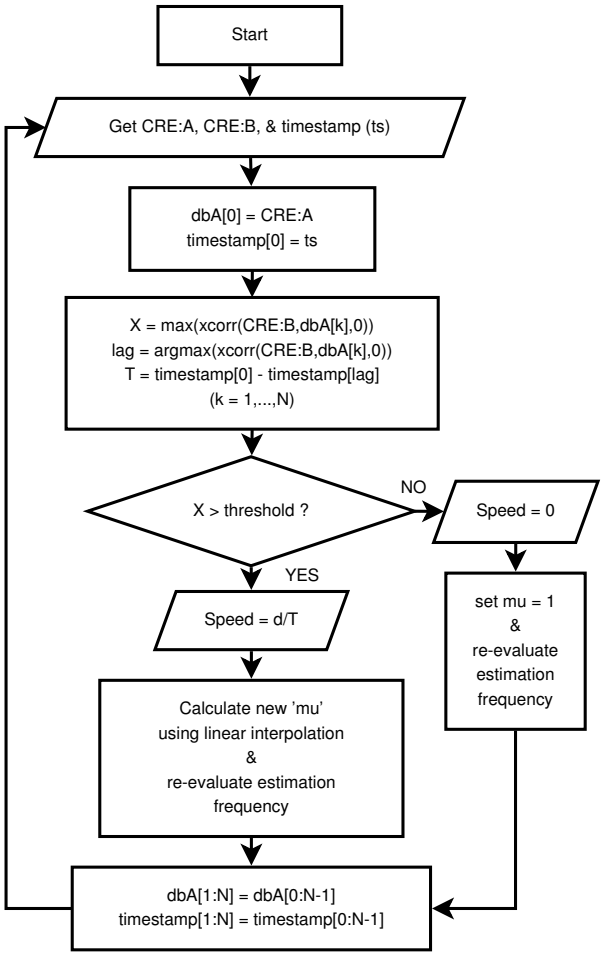


Fig. 4. Speed estimation algorithm. "dbA[]" is the CRE buffer associated with *antenna A*. The function `xcorr(a,b,L)` provides the normalized cross correlation product at lag = L (i.e. it performs the operations necessary to compute the parenthesized expression in (7)).

To prevent false measurements, the correlation-product with the highest value is compared against a predefined threshold (a suitable value can be determined empirically). Fig. 4 shows an algorithmic flowchart of the one-dimensional and unidirectional speed estimation algorithm. If the algorithm is to be extended to the three-dimensional and bidirectional case, an array with antenna pairs in each of the three dimensions will be required. For bidirectional support, CRE databases will have to maintained for each antenna in a pair. Then, *six* independent estimation processes (two for each antenna pair) will have to be run in parallel. The non-zero speed estimates from each pair can then be added as a vector sum to yield the velocity of the device in three-dimensions.

IV. MATLAB Implementation and Results

The following simplifying assumptions are made to ease the simulator implementation:

- The carrier phase offsets (ϕ_l), AoAs (θ_l), and time delays (τ_l) are random but time-invariant.
- The channel response estimate is exact and error free.
- Displacement is in one dimension only; therefore, $\Psi_{l,i}(t)$ can be simplified to

$$\Psi_{l,i}(t) = [x_i \cos(\theta_l(t))] \cdot \beta. \tag{14}$$

The multipath channel is modeled by ten independent time-varying Rayleigh fading processes (implemented using MATLAB `rayleighchan` channel objects) and has a maximum delay spread of 5μs. The AoA and carrier phase offsets are modeled as random variables uniformly distributed in $[0, 2\pi]$. The carrier frequency used is 474MHz and each OFDM symbol consists of 1705 subcarriers (DVB-T in 2K mode [10]). The threshold is maintained at 0.95. The use of such a high threshold is justified since the CRE estimates are assumed to be perfectly known and error-free.

Fig. 5a and 5b show the performance of the estimation algorithm at $d = 0.10\text{m}$ and 0.45m respectively. The first plot shows a very good match between the real and estimated speeds while the latter with a larger d shows relatively poor performance. The reason for this becomes clear once the coherence time of the channel is taken into account. At a separation distance of 0.45m, the time required by *antenna B* to reach the position previously occupied by *antenna A* is around 0.3s for a constant speed of 1.5ms^{-1}. However, at that speed, the coherence time of the channel is only about 0.18s. Consequently, the relevant CREs are no longer strongly correlated and a reliable speed estimate cannot be obtained.

Fig. 5c shows the percentage errors ($\frac{\text{abs. error}}{\text{real speed}} \times 100\%$) in the estimates over the prescribed speed range for values of d ranging from 0.05m to 0.15m. The results are extracted from simulations of a ramp-like speed profile over the prescribed range ($0.1-15\text{ms}^{-1}$). The acceleration is constant at 0.5ms^{-2}. At higher speeds, the smaller antenna separation distances show larger errors. The reason is that a smaller d allows a smaller range for μ (see (10)) and consequently leads to a very limited number of discrete speed levels that can be estimated. As shown in Table I, this leads to very large gaps at higher speeds. At lower speeds (less than 2ms^{-1}), there appears to be a singularity where the error suddenly drops before returning to a very high value and gradually decreasing again. This corresponds to the behavior seen at the start of Fig. 5a (shown zoomed in an inset). This is an artifact caused by the high level of spatial correlation between subsequent CRE estimates at the given antenna separation distance (0.10m). Starting with an estimate of 0ms^{-1}, as *antenna B* approaches a position previously occupied by *antenna A*, the correlation between the CREs gradually increases and eventually exceeds the threshold defined – although it has not yet reached the right coordinates. This results in an underestimate of the lag and yields an erroneous estimate slightly higher than the real speed – producing the singularities observed. The errors in the lag then dissipate and the errors due to the acceleration (as detailed below) dominate.

Fig. 5d is a plot of the percentage change in the speed ($\frac{|v_n - v_{n-1}|}{v_{n-1}} \times 100\%$) after *each* estimation instant. For constant

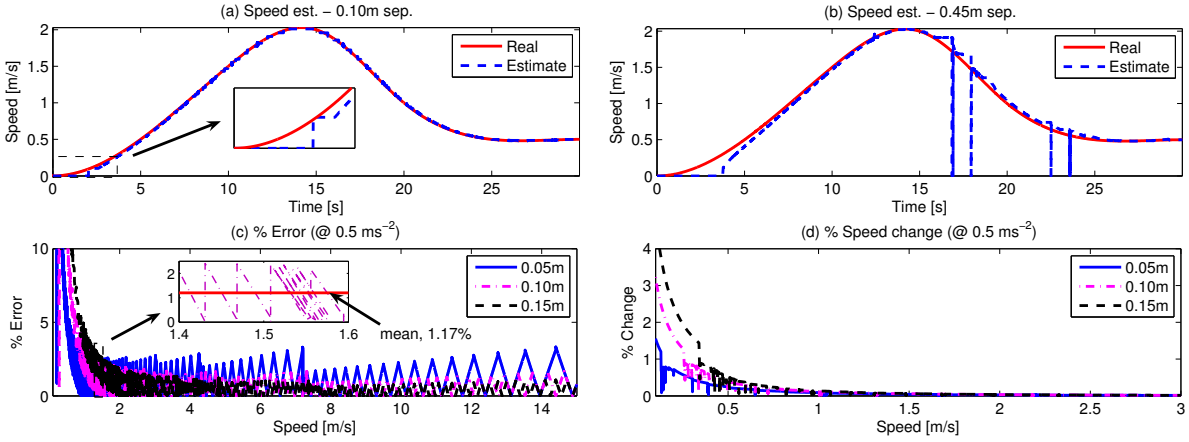


Fig. 5. MATLAB simulations of the one-dimensional and unidirectional speed estimation algorithm. The threshold is set at 0.95 and the antenna separation is varied from 0.05m to 0.45m. The first two plots (a,b) show typical simulation results for antenna separation distances of 0.10m and 0.45m respectively. Plot (c) shows the estimation error versus speed while plot (d) shows the percentage change in speed after each estimation instant for a constant acceleration of 0.5ms^{-2}.

acceleration, the change is higher at lower speeds. Furthermore, a larger d results in a larger change due to increased time difference between estimation instants (9). It should also be noted that at low speeds, a very large number of estimation instants pass before *antenna B* reaches a position previously occupied by *antenna A*. Ultimately, this means that the CREs to be compared have different characteristics due to different Doppler spreads and hence the speed estimate suffers. Nevertheless, the accuracy for low speeds is still very high. In order to determine typical performance levels at pedestrian speeds, the arithmetic mean of the percentage errors (Fig. 5c) between 1.4ms^{-1} and 1.6ms^{-1} is calculated for $d = 0.10\text{m}$. This yields a *mean* error of less than 1.2% (shown in the inset) – indicating that the scheme can indeed be effective and accurate.

V. Conclusion

This paper presents a simple adaptive speed estimation algorithm that is capable of producing high resolution estimates within a given range. It is possible to obtain estimates with an average error of less than 1.2% at typical pedestrian speeds. The scheme utilizes relative RF signature matching and hence does not require *a priori* information regarding the environment (no large databases of exact channel estimates are required). Also, no additional hardware would be required since future wireless devices are expected to be MIMO capable.

It should be noted that the proposed method can be easily extended by means of different antenna array geometries for speed estimation in any arbitrary direction. Such an extension is, however, beyond the scope of this paper and subject to future work. In addition, finding optimum values for the antenna separation and threshold will be the focus of further research.

Acknowledgement

This project is in collaboration with MobilTec GmbH & Co. KG, and funded by the BIS Bremerhaven as part of the Bremen in T.I.M.E. program under grant 56023/2-Z. We also would like to thank T-Mobile Germany, in particular Mr Ehmke for supporting this work. Moreover, the authors would like to thank Dr. Sinan Sinanović (Jacobs University) for his valuable technical contributions.

References

[1] J. Collin, O. Mezentsev, and G. Lachapelle, "Indoor Positioning System Using Accelerometry and High Accuracy Heading Sensors," in *Proc. of ION GPS/GNSS 2003 Conference*. Portland, OR: Institute of Navigation, Sep. 9–12, 2003.

[2] J. Kappi, J. Syrjarinne, and J. Saarinen, "MEMS-IMU Based Pedestrian Navigator for Handheld Devices," in *Proc. of ION GPS 2001 Conference*. Salt Lake City, UT: Institute of Navigation, Sep. 11–14, 2001, pp. 1369 – 1373.

[3] C. Randell, C. Djiallis, and H. Muller, "Personal Position Measurement Using Dead Reckoning," *Proceedings of the 7th IEEE International Symposium on Wearable Computers 2003*, pp. 166–173, Oct.18–21, 2005.

[4] A. Abdi and M. Kaveh, "Level Crossing Rate in Terms of the Characteristic Function: a New Approach for Calculating the Fading Rate in Diversity Systems," *IEEE Transactions on Communications*, vol. 50, no. 9, pp. 1397–1400, Sep. 2002.

[5] L. Zhao and J. W. Mark, "Mobile Speed Estimation Based on Average Fade Slope Duration," *IEEE Transactions on Communications*, vol. 52, no. 12, pp. 2066–2069, Dec. 2004.

[6] R. Narasimhan and D. C. Cox, "Speed Estimation in Wireless Systems Using Wavelets," *IEEE Transactions on Communications*, vol. 47, no. 9, pp. 1357–1364, Sep. 1999.

[7] R. B. Ertel, P. Cardieri, K. W. Sowerby, T. S. Rappaport, and J. H. Reed, "Overview of Spatial Channel Models for Antenna Array Communication Systems," *IEEE Personal Communications [see also IEEE Wireless Communications]*, vol. 5, no. 1, pp. 10–22, Feb. 1998.

[8] P. Höher, "A Statistical Discrete-Time Model for the WSSUS Multipath Channel," *IEEE Transactions on Vehicular Technology*, vol. 41, no. 4, pp. 461–468, Nov. 1992.

[9] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, ISBN: 0130422320, Dec. 2001.

[10] ETSI EN 300 744 v1.5.1 (2004-06), *Digital Video Broadcasting (DVB): Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television*, European Telecommunications Standards Institute (ETSI) Std., Jun. 2004.