ucc

Coláiste na hOllscoile Corcaigh, Éire
University College Cork, Ireland

FINAL YEAR PROJECT REPORT

YANN DONNELLY

Supervisor: Dr. Colin Murphy
School of Engineering
University College Cork

March 2014 – version 1.0

DECLARATION

This report was written entirely by the author, except where stated otherwise. The source of any material not created by the author has been clearly referenced. The work described in this report was conducted by the author, except where stated otherwise.

Yann Donnelly, March 24, 2014

ABSTRACT

This report describes a final year project undertaken by the author. The project examined the effects of non-deterministic timing error on digital amplitude-modulated receivers. A background to the rationale behind the project is given, and the research methodology is discussed. Both an analytical description of the problem and a numerical simulation were explored. It was found that timing error had an attenuating effect on the received signal, and significant performance degradation was observed using traditional receivers. By defining the optimum decision region boundaries for a given timing offset variance, an improved decoder was described. This decoder demonstrated reduced error rates in the presence of timing error compared to traditional decoders. Finally, the author suggests practical implementation and smarter decoding as two possible continuations to the work described.

CONTENTS

LIST OF FIGURES

ACRONYMS

AWGN Additive White Gaussian Noise

BPSK Binary Phase Shift Keying

DRB Decision Region Boundary

EGC Equal Gain Combining

ISI Inter-Symbol Interference

MAP Maximum A Posteriori detector

ML Maximum-Likelihood detector

MRC Maximal-Ratio Combining

PAM Pulse Amplitude Modulation

PDF Probability Density Function

PSK Phase Shift Keying

SNR Signal to Noise Ratio

TP Teaching Period

1

INTRODUCTION AND OBJECTIVES

This project is focused on examining the effects of non-deterministic timing error in digital amplitude-modulation radio signal receivers. Timing errors remain an important issue in modern digital receivers, despite considerable research into improved clock generation and recovery circuits. This project takes a novel approach to the problem, as instead of trying to reduce the level of timing error, it assumes statistical knowledge of the timing error present and attempts to mitigate its effects at the decoding stage.

Work was divided into two parts:

- The effects of timing error on the received signal at the input to the decoder were examined, in order to determine the probability density function seen by the decoder.
- Using knowledge of the nature of the signal at the input of the decoder, communications theory was used to determine the optimum decoder for the given signal.

The ultimate goal was to demonstrate an optimum decoder that would perform significantly better than traditional decoders designed assuming perfect timing synchronisation. Numerical simulation work was backed up with a mathematical analysis.

2

BACKGROUND INFORMATION

2.1 THE COMMUNICATIONS SYSTEM MODEL

A typical radio communications system consists of a transmitter, a receiver and a communications channel, that may contain any number of non-idealities typically caused by stochastic processes. By modeling these non-idealities using fitting statistical probability distributions, a statistical description of the received signal can be derived and used to inform correct design of the receiver.

One could imagine a binary transmission system that sends one of two possible signals: a $1V_{RMS}$ wave if a '0' is to be sent, and a $3V_{RMS}$ if a '1' is to be sent. After being distorted by the communications channel, the receiver could in theory see a signal of any amplitude, and must make a decision as to which amplitude was originally sent. It would be helpful for the designer to know the probability density function (PDF) of the received signal, ie. the probability of receiving a signal amplitude if a known amplitude was sent. If one assumes the communications channel is memoryless and distorts the signal by adding zero-mean white Gaussian noise, the received signal PDF will be a Gaussian distribution centred on the sent signal amplitude, as shown below. If both symbols are equiprobable, their PDF's will be of equal height, and therefore shifted copies of each other.

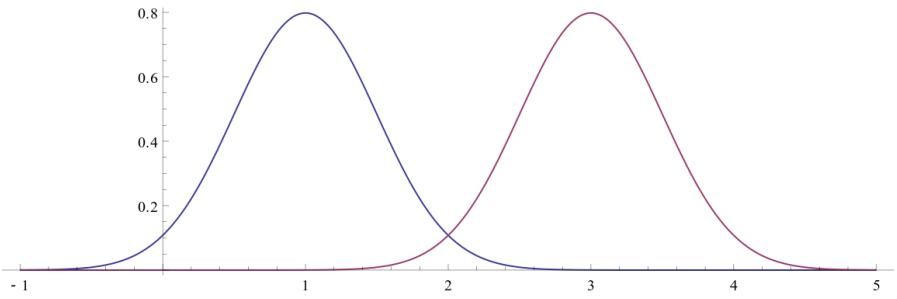


Figure 1: Gaussian noise corrupts a sent signal, resulting in a probability density function for each possible sent symbol

A *maximum-likelihood (ML) detector* seeks to minimise the probability of error by always picking whichever signal was most likely to be sent, given the received signal. The derived probability distributions can give us this information. The threshold between picking one value or the other will be where both symbols are equally likely to be sent, the point of intersection of both PDF's. Since the Gaussian distribution is symmetric about its mean, in this case the threshold (or *Decision Region Boundary*) is exactly midway between both sent

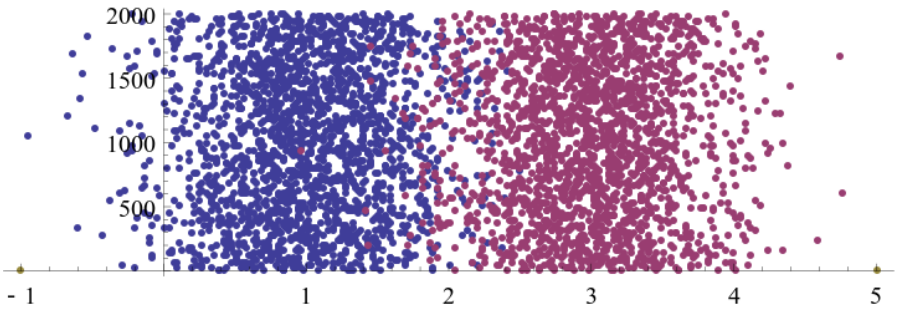


Figure 2: As an example, 2000 of each symbol has been send down the communications channel. The received values distorted by noise are plotted above. Note the overlap in received values corresponding to both symbols: it is impossible to detect the signal with 100% accuracy.

amplitudes. Intuitively this makes sense: it says that in the simplest case if one receives a particular signal, one should assume whichever possible sent signal is nearest. However in more complicated cases picking the point of intersection of both PDF's is the more general solution.

While applicable to cable channels, radio systems have to deal with numerous reflections from the environment, known as fading effects. A superposition of reflected signals, with different levels of attenuation, is therefore seen at the receiver. Traditional communications theory models fading as a random gain. Performance in this case can be improved by capturing multiple signals using multiple antennae, and combining these signals, either using equal weighting (Equal-Gain Combining), or by weighting each channel by its channel gain (Maximal-Ratio Combining).

2.2 TRANSMITTER-RECEIVER SYNCHRONISATION

One issue that complicates detection is *inter-symbol interference* (ISI). It is possible for signals representing symbols in the future or past to bleed into the current symbol clock period, distorting the signal further. For this reason transmitters and receivers are designed with filtering that applies a *raised cosine function* to the signal. This filter passes the signal at exactly the symbol's sample time, and attenuates completely at every other symbol's sample time, ie. at every interval of T_{clk} for $T \neq 0$. If both receiver and transmitter are perfectly synchronised, this ensures that the receiver will only see signals corresponding to the current transmitted symbol, after distortion via the communications channel.

If the receiver and the transmitter are poorly synchronised, however, the root cosine filter response is shifted in time with respect to the receiver; adjacent symbols bleed into the current symbol, and the

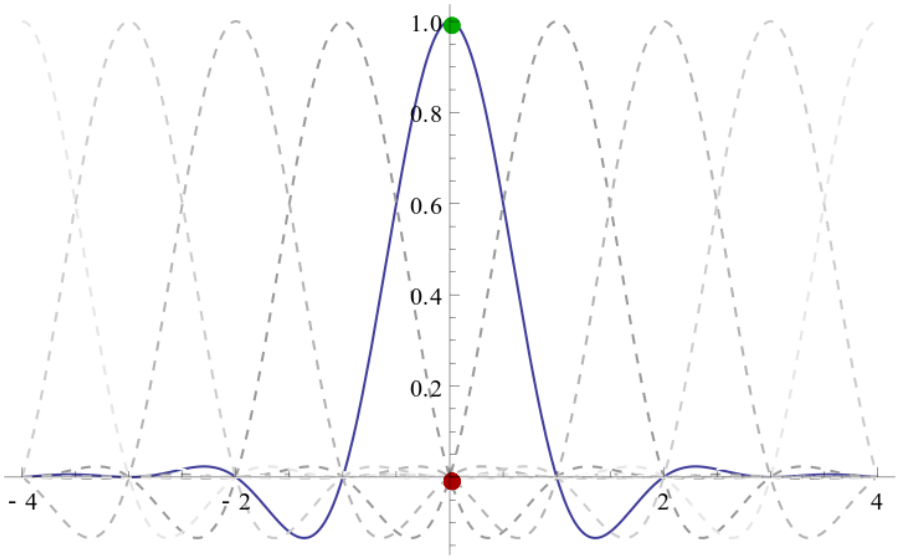


Figure 3: A filter with a root raised cosine function is often used, as it evaluates to 1 at the current sample time and 0 at all other sample times

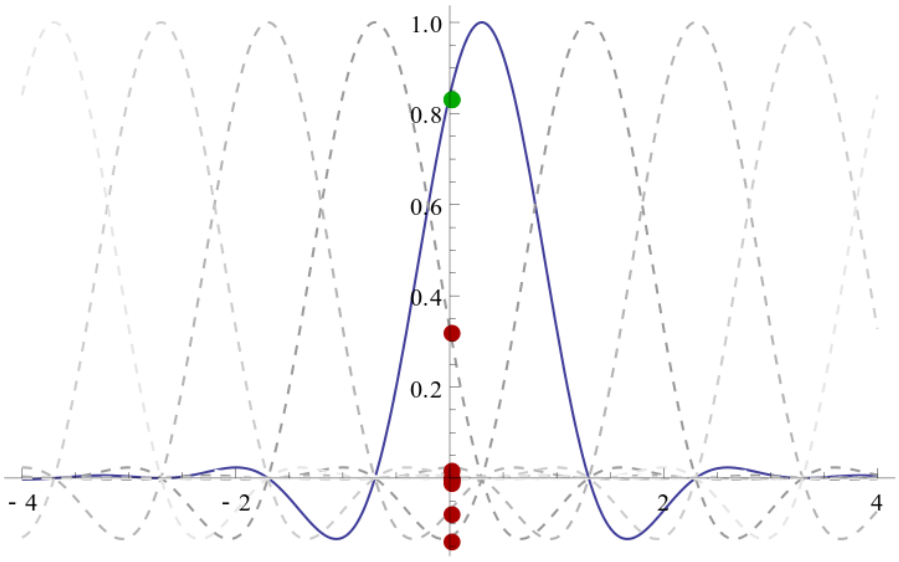


Figure 4: If a timing offset is added to the root raised cosine it no longer evaluates to 0 or 1 at the sampling time. This results in reduced receiver performance when the receiver and transmitter are not properly synchronised.

current symbol is attenuated according to the raised cosine function of the timing error. This last point is crucial to this project:

Proposition 1. *An amplitude-modulated system with a given channel function* $g[t]$ *and a timing error* δt *will experience an attenuation due to the timing error of the transmitted symbol equal to* $g[\delta t]$.

In practice, the adjacent symbols can be modelled as Gaussian noise, and we can therefore express the effects of timing error as a gain dependent on the instantaneous timing error.

2.3 PREVIOUS WORK

3

METHODOLOGY

This project models channel noise as additive and Gaussian. Fading is assumed to be Rayleigh-distributed and Maximal-Ratio Combining is considered as a counter-measure (Equal-Gain Combining was examined and found to perform too poorly to be of interest). Timing error is assumed to be Tikhonov-distributed, and knowledge of both channel gain and approximate timing error variance is assumed to be available to the decoder.

A simplified model of a communications system ignoring fading was developed in order to allow the author to familiarise himself with the problem, its associated theory and the *Mathematica* environment. This model was used to verify Proposition 1 above.

the ultimate goal was to prove the existence of sub-optimal performance in the system due to receiver timing errors, and demonstrate an optimum decoder with better performance than traditional decoders assuming perfect synchronisation. The effects of fading were incorporated into the model and the performance of the system in the presence of timing error was examined thoroughly through simulation, to determine the qualitative effects of timing error. The optimum decision region boundaries were then determined to inform the optimum decoder description. Finally, the resulting SER for both receivers was determined from the simulation results.

An analytical verification of these results was attempted by approximating the received symbol PDF conditioned on know timing offsets and channel gains by a Gram-Charlier series and averaging over the respective distributions.

4

RESULTS

4.1 EFFECTS OF TIMING ERROR ON THE NON-FADING SYSTEM

The general effects of timing error were examined through simulation. A Mathematica simulation of the AWGN communications system with a variable timing error parameter was constructed, and run for a range of timing error parameters.

Timing error was found to result in attenuation of the received signal The level of attenuation depends on the severity of the timing error and the response of the channel, as given be Proposition 1 given earlier.

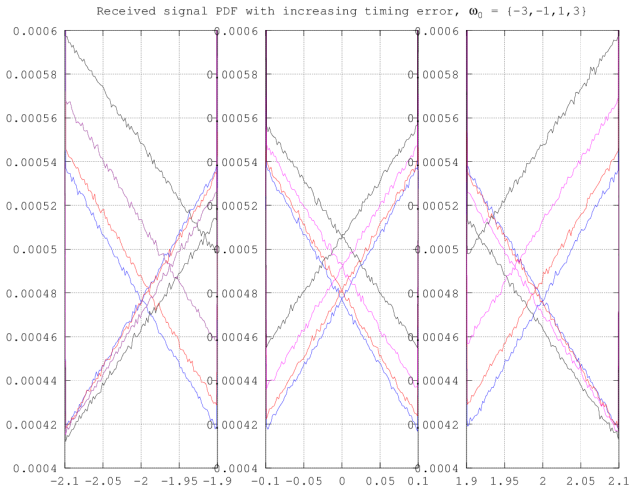


Figure 5: Recieved signal PDF for a number of timing offsets (4-PAM, AWGN channel, SNR=4dB))

The effects of timing error offset were significant enough to warrant a study using a more realistic fading model, in order to determine the effects in a typical system.

4.2 EFFECTS OF TIMING ERROR ON THE RAYLEIGH-FADING SYSTEM

The previous model was extended to include the effects of Rayleigh-distributed fading. The receiver was extended to an l-antenna system, accompanied with selectable EGC or MRC techniques. Timing error was implemented as a Tikhonov-distributed stochastic process.

Both EGC and MRC systems were similarly examined, and in both cases the effects of timing error were also present, to a lesser extent than in the non-fading case, as the antenna diversity has an averaging effect on the received signal. Nonetheless the presence of attenuation suggested that decoding performance was sub-optimal, and could be improved through decoder redesign.

4.3 OPTIMAL DECODING IN THE PRESENCE OF TIMING ERROR

The optimum decision region boundaries can be shown to be the intersection between adjacent symbol's PDFs at the input to the decoder. The received symbol histograms generated through simulation were used to determine the optimum decision region boundaries, and the Symol Error Rate defined as the probability of a symbol being incorrectly decoded by the decoder.

$$\mathrm{SER} = \int_{Y_{DRB}}^{\infty} H_1(Y)\, dY + \int_{-\infty}^{Y_{DRB}} H_3(Y)\, dY$$

If the $\omega = 1,3$ histograms H_1 and H_3 have a bin width ΔY, and a decision region boundary of Y_{DRB} is chosen, where $Y_{DRB} \simeq (n + \frac{1}{2})\Delta Y$, then the SER was approximated as

$$\mathrm{SER} \simeq \sum_{i=(n+1)\Delta Y}^{N} H_1[i] + \sum_{j=0}^{n\Delta Y} H_3[j]$$

The Symbol Error Rates were determined for a range of timing error variances. EGC was found to perform unsatisfactorily at the target SNR of 20dB, so the MRC system was focused on for the continuation of the project. Timing offset was shown to increase Symbol Error Rates by up to two orders of magnitude[1].

Note on determining the SER: In this project, symmetric 4-PAM signalling was used, so one of the three decision region boundaries is located at 0 and therefore unaffected by timing error. The overall SER presented here is therefore 2/3 that calculated.

With the fall in received signal amplitude comes a fall in optimum decision region boundary. The optimum decision region boundary for the MRC system was found to drop with increasing timing error. Therefore, the traditional system performs sub-optimally. An optimum receiver was designed taking into account the variance of the timing error, and was found to result in a decrease of up to 11% in SER, for the given number of diversity branches and SNR.

While the simulation demonstrated the effects of timing error on the described system, a complementary analytical description of the problem would provide additional support to these findings.

1 4-PAM signalling, l=2, SNR=20dB

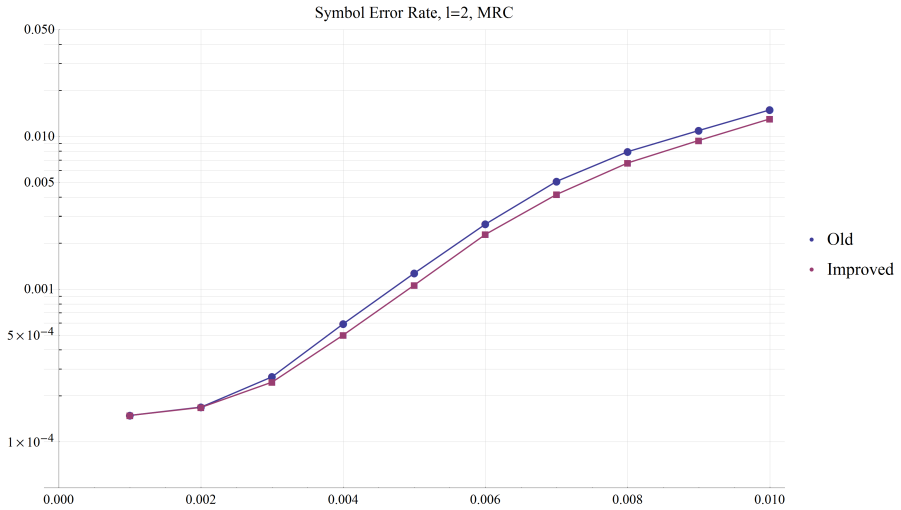


Figure 6: Symbol Error Rate versus timing error variance (4-PAM, Rayleigh-fading AWGN channel, MRC, l=2, SNR=20dB, N= 5×10^{10}))

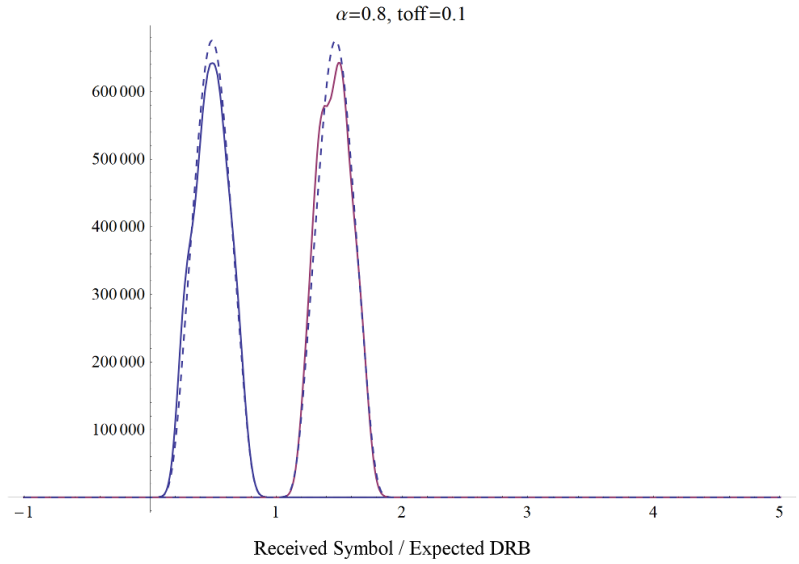
4.4 ANALYTICAL STUDY OF TIMING ERROR

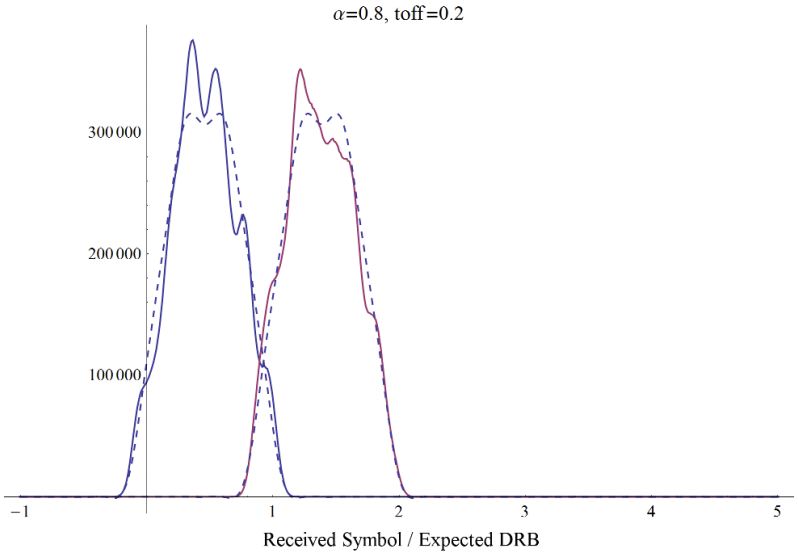
By approximating the received signal PDF, conditional on set channel gains and timing errors, by a Gram-Charlier series[2], an analytic solution to the conditional PDF was developed. Simulation and analytical plots were compared and the latter was found to be a sufficiently strong approximation to provide an estimate for the optimum decision region boundary.

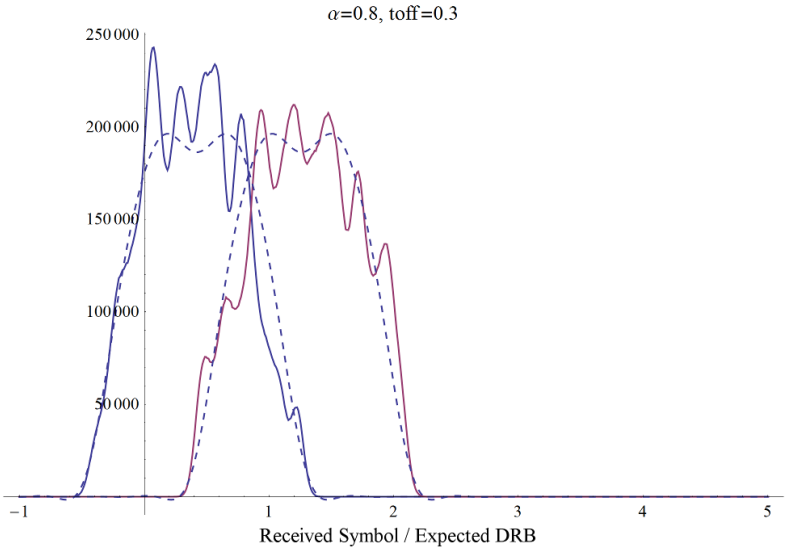
Figure 7: Overleaf, left: Comparison between simulated received signal PDF (solid) and Gram-Charlier approximation (dashed)

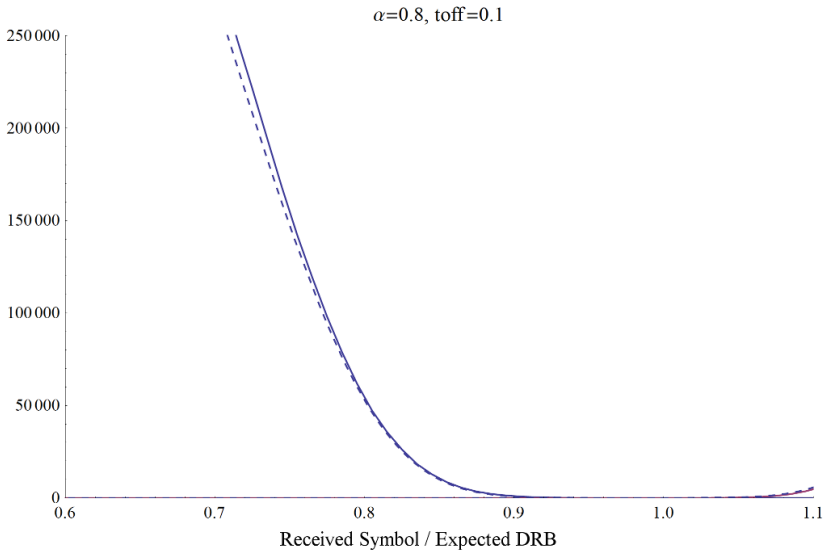
Figure 8: Overleaf, right: Comparison between simulated received signal optimum decision region boundary (solid) and Gram-Charlier approximation (dashed)

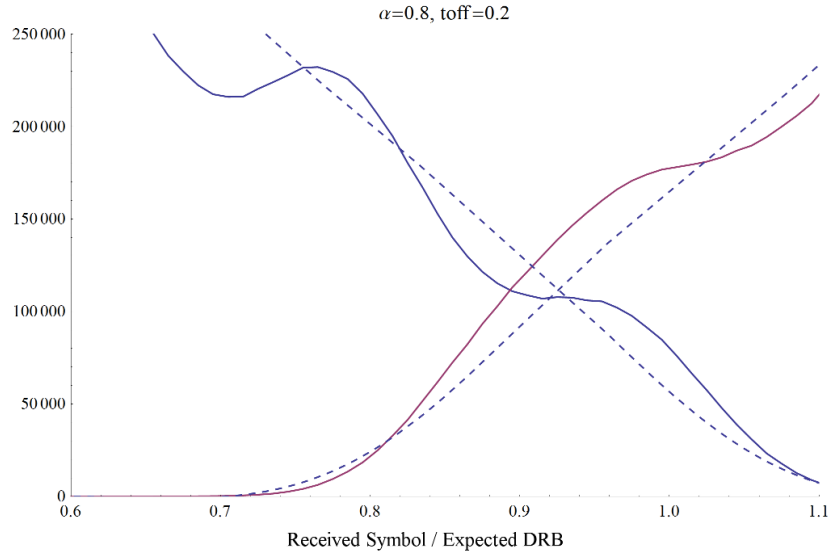
2 See Appendix TODO

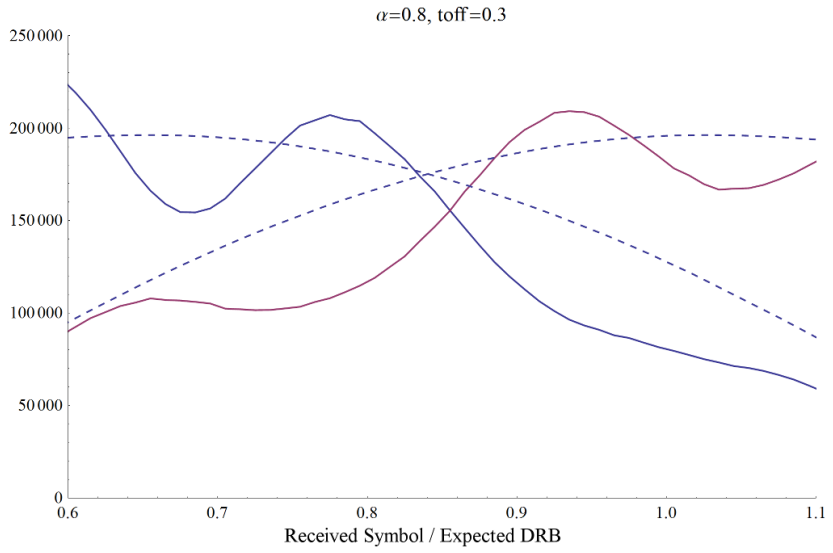

α=0.8, toff=0.1
600 000
500 000
400 000
300 000
200 000
100 000
−1
1
2
3
4
5
Received Symbol / Expected DRB


α=0.8, toff=0.2
300 000
200 000
100 000
−1
1
2
3
4
5
Received Symbol / Expected DRB


α=0.8, toff=0.3
250 000
200 000
150 000
100 000
50 000
−1
1
2
3
4
5
Received Symbol / Expected DRB


α=0.8, toff=0.1
250 000
200 000
150 000
100 000
50 000
0
0.6
0.7
0.8
0.9
1.0
1.1
Received Symbol / Expected DRB


α=0.8, toff=0.2
250 000
200 000
150 000
100 000
50 000
0
0.6
0.7
0.8
0.9
1.0
1.1
Received Symbol / Expected DRB


α=0.8, toff=0.3
250 000
200 000
150 000
100 000
50 000
0
0.6
0.7
0.8
0.9
1.0
1.1
Received Symbol / Expected DRB

5 DISCUSSION OF RESULTS

This project has demonstrated a fundamental effect present to a certain degree in a large number of communications systems. The particular system used to illustrate the problem is representative of a large number of systems; the channel response need not necessarily be Raised-Cosine, merely relatively low-bandwidth, and the effect could occur with any similar choice of fading or timing distribution.

The proposition that the conditional effects of timing error on the received symbol could be approximated by an attenuation was proven through simulation. Furthermore, it was shown that this property could be used to reduce the effects of timing error offset through redesigning the symbol detection algorithm, and the effectiveness of this route was proven.

While the improvements may not seem massive, it is key to note that the change is made *without adding any stages to the detector*. The proposed changes can be implemented with minimum complexity - the factor change in decision region boundary for a given timing error can be pre-determined and stored in a look-up table, and current timing statistics can be used to address the relevant correction factor.

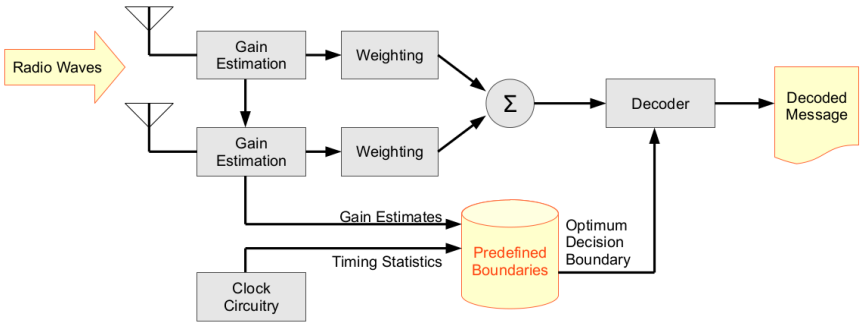


Figure 9: Improved decoder implementing the described corrections as a LUT and multiplier block.

6 FUTURE WORK

Future work will be carried out by the author on extending the analytical solution to the conditional PDF to an analytical solution for the overall PDF, with a goal to deriving the optimum decision region boundary for a given timing error variance. This would allow the correction terms for a practical system to be determined without having to perform extensive simulation work.

Given a complete analytical solution for the overall received symbol PDF, it would be possible to generate the optimum decision region boundary conditioned on factors other than timing error variance, such as channel gain and channel SNR. The performance gains from such a system would have to be offset against system cost, but these parameters are readily available to the system.

An extension of the theory presented here to other receiver architectures, channel models and conditions would be a logical continuation to this project, and would help determine the areas in which timing errors are of most concern. In particular, Nakagami-n fading and the effects of increased diversity were not studied, and could yield some interesting results.

This project has theoreticized a fundamental issue present in amplitude-modulated communications systems prone to timing errors. It would be complemented with a practical description of the problem. The described 4-PAM MRC system could be implemented and either the timing offsets artificially added in or the hardware's own timing error monitored.

7

CONCLUSIONS

Part I

APPENDIX

A

LOGBOOK

"This report, by its very length, defends itself against the risk of being read." - Winston Churchill

A.1 WEEK 1

A.1.1 *30/09/13 - Exploring simple case with PAM modulation*

I received the *PAM.pdf* file outlining the case where a signal is sent through a channel with AWGN and received with a timing error at the receiver. I read through the file several times to get an understanding of the underlying equations.

Leaving the Gram-Charlier series aside for the moment, I started getting to grips with Mathematica and implementing the transmission system model:

$$X = \omega_0 g_0 + \sum_{k=1}^{40} (\omega_{-k} g_k + \omega_k g_k) + \nu$$

where $g_k = g((\Delta + k)T)$, $g(t) = (u_T * h_l * u_R)(t) \times \cos(\theta)$ and ν is a zero-mean Gaussian random variate with $\sigma_\nu^2 = N_0 \varepsilon_R$.

I learnt the basics of the interface, and began implementing the filter and channel impulse responses (I.R.). I need to double-check the definition of the Root-Raised Cosine (RRC) Filter, as the impulse response wasn't as expected.

Later, I found the correct form for the RRC [**?**] and double-checked it using Octave. The equation used is listed below. A plot showed that this equation is invalid at $t = \left[-\frac{T_s}{4\beta}, 0, \frac{T_s}{4\beta}\right]$, so I plan to find its limit at these points using Mathematica to obtain the complete solution.

$$h_{RRC}(t) = \frac{2\beta}{\pi\sqrt{T_s}} \frac{\cos\left[(1+\beta)\frac{\pi t}{T_s}\right] + \frac{\sin\left[(1-\beta)\frac{\pi t}{T_s}\right]}{\frac{4\beta t}{T_s}}}{1 - \left(\frac{4\beta t}{T_s}\right)^2}$$

A.1.2 *01/10/13 - Implementing Raised Cosine functions*

I implemented the function above in Mathematica, and using the `Limit` function found the value of the function at the following undetermined points:

$$h_{RRC}(t) = \begin{cases} \dfrac{4\beta + \pi(1-\beta)}{2\pi\sqrt{T_s}} & t = 0 \\ \dfrac{\beta}{2\pi\sqrt{T_s}}\left(\pi \sin\left[\frac{(1+\beta)\pi}{4\beta}\right] - 2\cos\left[\frac{(1+\beta)\pi}{4\beta}\right]\right) & t = \pm\frac{T_s}{4\beta} \\ \dfrac{2\beta}{\pi\sqrt{T_s}} \dfrac{\cos\left[(1+\beta)\frac{\pi t}{T_s}\right] + \dfrac{\sin\left[(1-\beta)\frac{\pi t}{T_s}\right]}{\frac{4\beta t}{T_s}}}{1-\left(\frac{4\beta t}{T_s}\right)^2} & \text{otherwise} \end{cases}$$

I also implemented the Raised Cosine function for the channel function, using the impulse response below[1]. I was unable however to convolve the receiver and transmitter filter functions using the `Convolve` function, even when I limited the impulse response using a `UnitBox`.

$$h_{RC}(t) = \frac{\operatorname{sinc}\left(\frac{\pi t}{T}\right)\cos\left(\beta\frac{\pi t}{T}\right)}{1-\left(2\beta\frac{t}{T}\right)^2}$$

I looked into Mathematica's treatment of the Gaussian distribution, and figured out how to generate random noise vectors following a Gaussian distribution, as well as how to generate a list of random binary symbols.

After discussing the convolution issue with David Murphy, he suggested that the channel should be initially modelled as ideal and therefore the overall channel and filter I.R. $g(t)$ can be defined as a Raised Cosine function, as defined above. I should therefore be ready to implement the simple ISI model tomorrow.

A.1.3 *02/10/13 - Wrapping Up the Initial PAM Model*

I pulled together the Raised Cosine function and random number generator to impiment the given simplified function for the PAM receiver output, given below. Playing around with the settings, I was able to show how the g_k function increases with the timing error. I decided to study the Mathematica environment a little more before carrying on with any programming.

$$X = \omega_0 g_0 + \sum_{k=1}^{40} (\omega_{-k} g_{-k} + \omega_k g_k) + \nu$$

A.1.4 *03/10/13 - Delving deeper into Mathematica*

I devoted some time into looking through Michael Quinlan's notebooks and better understanding the workings of the `Table` functions

1 Proakis, "Digital Communications"

and the various plotting options. Fortunately my notebook was corrupted so I was able to rewrite it and understand the model a bit more. I need to figure out what variance value the noise PDF should take on, as the noise appears to be overwhelming the timing error effects. Translating the resulting PDF's into patterns is another question that needs some thought.

A.1.5 *Week 1 Summary*

Week 1 was mostly spent becoming acquainted with Mathematica and getting a feel for the equations underlying PAM transmissions. A simple model of a PAM receiver was constructed.

A.1.6 *Goals for Week 2*

- The PAM model will need to be extended to calculate the optimum decision region boundary from the estimated PDF.
- A better setup will be required to perform large-scale simulations within an acceptable time period. We will look into applying for an account on the Boole cluster.

A.2 WEEK 2

A.2.1 *07/10/13 - Matrix manipulations*

I decided to spend another day learning about the Mathematica environment, in particular matrix manipulation and generation. I looked into the `Apply`, `Map` and `Partition` functions and wrote some examples to figure out how to convert mathematical problems to Mathematica notation using matrices. I hope to convert the code to use matrices tomorrow to hopefully simplify and speed things up.

I also implemented PhD student David Murphy's equation for properly calculating the AWGN function variance from SNR[2], from last Friday's meeting.

2 $\sigma_\tau^2 = \frac{L^2-1}{6\log_2(L)\left(\frac{E_b}{N_0}\right)}$

A.2.2 *08/10/13 - Fixed I.R. and Kernel Density Estimation*

The first job was to rewrite the code to make use of the simple dot operator to calculate all the ISI components[3]. With the new code I was able to carry out many more runs and get much more detailed output. In addition, when I was rewriting the code I noticed a typo in the Raised Cosine I.R. that was degrading performance in the perfectly synchronised case. With both of these changes made, I decided to use Kernel Density Estimation to see what effects the timing offset has.

Using offsets of 10^{-15}, 0.05, 0.1 & 0.15, the following values of g_k, $k \in \{-40 \cdots -1, 1 \ldots 40\}$ were calculated.

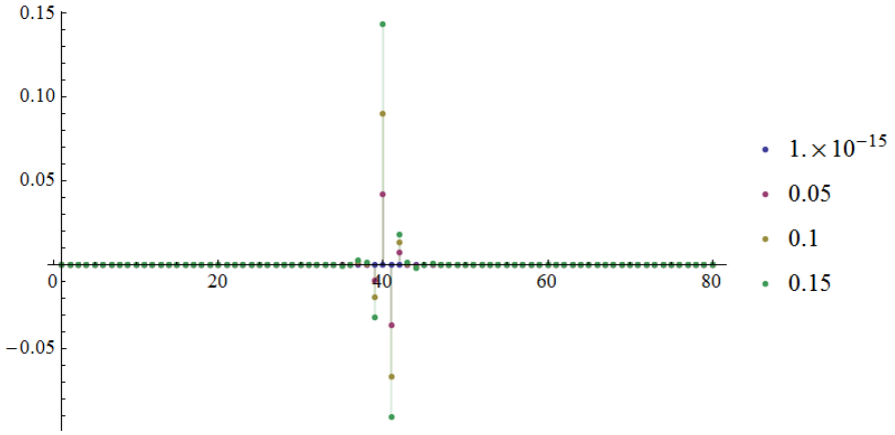


Figure 10: g_k linear plot

Using `SmoothKernelDistribution` to perform Kernel Density Estimation with 1 million points produced the following estimated PDFs for both possible transmitted values. As the timing error increases, we note that the PDF spreads out, but the mean remains steady.

A.2.3 *09/10/13 - Setting up Digital Comms Lab PC*

With Ger Houten's help, I set up an account on `Digital Comms Lab 1` & `Digital Comms Lab 2` and got the internet working. Mathematica 8 is installed and working on both machines, we will have to consider

3 The ISI components are now calculated using:

$$\left[\sum_{k=0}^{k=40}\left(g_k\omega_k^j + g_{-k}\omega_{-k}^j\right)\cdots\right]^{j=\{1..m\}} = [g_{-40}\cdots g_{-1}g_1\cdots g_{40}]\bullet\begin{bmatrix}\omega_{-40}^j\\ \vdots\\ \omega_{-1}^j\\ \omega_1^j\\ \vdots\\ \omega_{40}^j\end{bmatrix}^{j=\{1..m\}}$$

where ω_k^j is the k'th ISI with the j'th timing offset.

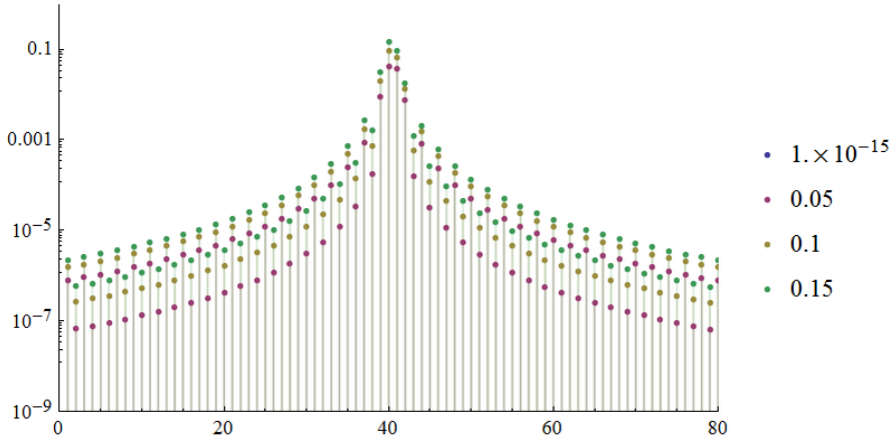


Figure 11: g_k log plot

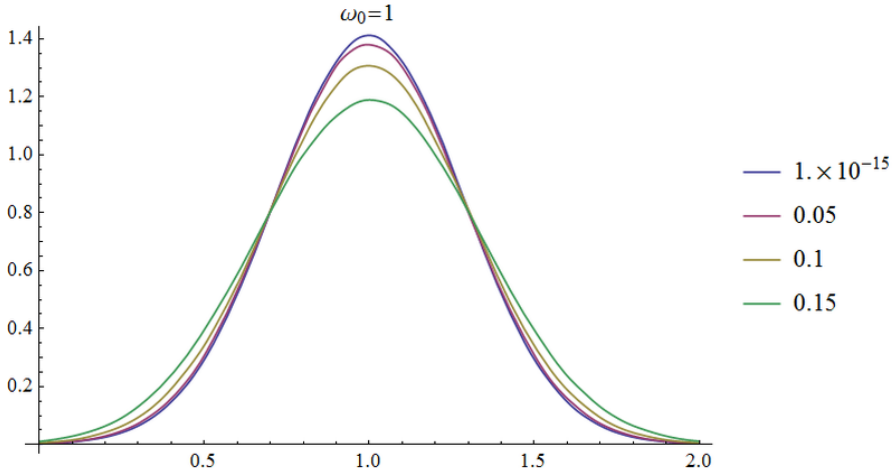


Figure 12: Kernel density estimation $\omega_0 = 1$

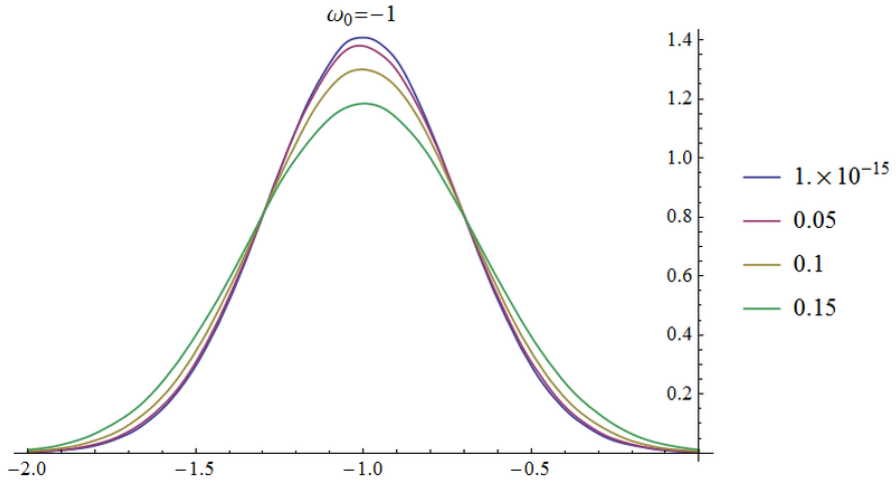


Figure 13: Kernel density estimation $\omega_0 = -1$

whether an upgrade to Mathematica 9 would be useful or not. Git and VNC or similar have to be installed next. A request was made to the Boole cluster for access for this machine, however the email given (`bcrisupport@bcri.ucc.ie`) was invalid.

A.2.4 *10/10/13 - Probing the Elec Eng network*

After finding out the Boole cluster was no more, I used today to examine what hardware I had available to me. I got access from Ger to the public `UEPC004` server, and from there I am able to access machines on the Engineering network. I set up a *Remote Desktop Protocol* link to `Digital Comms Lab 1` through this server, allowing me to control the machine from any location. I am also able to log remotely into EDA lab machines, and run Mathematica 6 on those machines. Ger Houten has been known to tweak machines in response to personal requests, so if asked nicely he may let me use two or three of these machines concurrently.

The GUI does not work when using `ssh` to access the EDA Lab machines, but using the command `math` to start and operate Mathematica kernels does.

Given these resources, I feel there are three ways I could continue:

- I could upgrade to the latest version of Mathematica on all machines, and set up a Mathematica cluster with `Digital Comms Lab 1` as the front end and the EDA Lab PCs as remote nodes. With this setup, all machines would act as one (as in a traditional cluster). This would be the easiest to use, but would require considerate work to set up.
- I could use the `MathLink` interface to achieve a similar, lower-level version of the former, with the EDA Lab machines as independent, remote slaves and `Digital Comms Lab 1` sending commands to these slaves and collating the replies. This setup is distributed computing with a star topology, and would be easier to setup. The downside is that the code needs to manually divide the task between each of the nodes, and needs to be well designed to minimise network delays.
- I could simply run the code in parallel on each of the machines available to me, dumping the results to text files, and collate the data at the end. This would require no setup, and code written on any machine would only require porting to another version of Mathematica. Additionally this seems like it would deal best with hiccups such as machines going down and it does not depend on a connection between the machines. The downside is there would be some overhead with collecting the results afterwards.

A.2.5 *Week 2 Summary*

I fixed the code written last week and began setting up my simulation environment.

A.2.6 *Goals for Week 3*

- Work out a setup that will allow me to carry out largeer-scale simulations.
- Adapt the previous code to run in parallel and produce useful machine-readable output.

A.3 WEEK 3

A.3.1 *14/10/13 - Running longer scripts on the EDA machines*

Today I spent some time figuring out how to build and run scripts on the EDA machines. I found that defining a module in a text file and copying the Mathematica code into the module allows the code to be called with input arguments, and writing the output to a text file and placing the module in a loop allows each pass to be recorded for later parsing. After running the code overnight, this system appears to work, and is scalable over multiple machines. The main disadvantage is the size of these files (7.7MB per 400,000 values), so I must either figure out how to transfer them over the network or see whether reducing the precision of the output values will reduce the file sizes.

Using the `Get` and `Put` methods. The `DumpSave` method is supposed to be more efficient, but was added after Mathematica 6.

A.3.2 *15/10/13 - Reducing output size*

Given the 15GB of samples produced the night before was far too much to pull off the machine, I copied 20 million of the samples and plotted them to make sure the script had worked in practice. I then looked into how I could reduce the size of the output produced, and decided to replace the `SmoothKernelDistribution` function (which came in after Mathematica 6.0 and therefore couldn't be used on the EDA machines) with a fine-grained histogram function. This allowed me to add the probabilities generated in each sweep to those generated before and keep the output to a handful of 1kB files. I ran the simulation overnight to check it.

For the record, I could only use a fraction of them, as loading all 20 million samples crashed the machine for over an hour.

I am assuming that both the smooth kernel distribution and the histogram approach the true PDF as $N \to \infty$

A.3.3 *16/10/13 - Moving onto 4-PAM*

Checking the output from the night before, I get a similar PDF plot as with the `SmoothKernelDistribution` function. I therefore modified the code to examine all 3 decision region boundaries in a 4-PAM sys-

tem and ran the simulation for 100 million samples per condition. The resulting distributions shown below show increased probability of error with timing error, as expected, but decision region boundaries in this case remain the same.

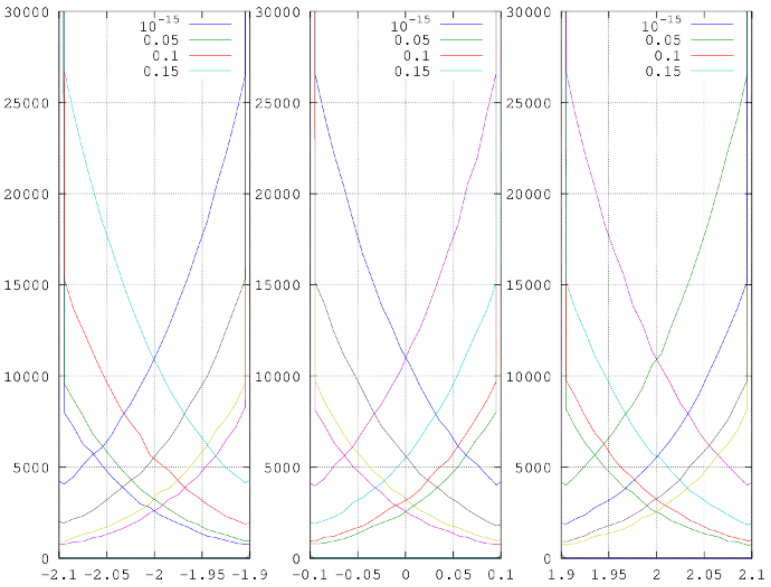


Figure 14: PDF for 4-PAM, $\omega_0 \in -3, -1, 1, 2$, 10^8 samples

I could imagine finding a value for the probability of error and moving onto PSK systems as the next steps in the process.

A.3.4 *Week 3 Summary*

Code was written that could be executed in parallel on multiple machines, and this was demonstrated in practice. The code was extended to the 4-PAM case, and showed no change in optimum decision region boundaries. Upon later consultation with David Murphy, it seems this is because the decision region boundaries shift due to a change in the g_0 term, and not the appearance of ISI components due to the g_k terms; the latter was believed to be the expected cause, and so the g_0 term was assumed to be 1 in the code.

A.3.5 *Goals for Week 4*

- Re-run the simulations to see if implementing the change in g_0 with timing error changes the location of the optimum decision region boundaries.
- If so, it would be interesting to see if the Gram-Charlier approximation produces the same boundary locations.

A.4 WEEK 4

A.4.1 *21/10/13 - Implementing the Gram-Charlier series*

Over the weekend, I implemented the g_0 term fix discussed in our Friday weekly meeting and re-ran the simulation, this time across two machines. Results showed that the Decision Region Boundary is displaced towards the origin as the timing offset increases.

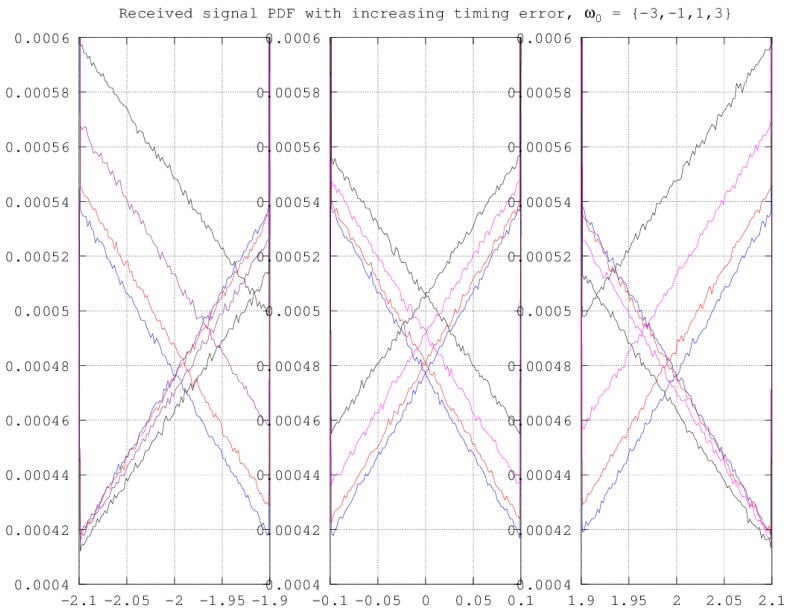


Figure 15: PDF for 4-PAM, $\omega_0 \in -3, -1, 1, 2$

I spent Monday carrying out two tasks:

1. I re-wrote David Murphy's Gram-Charlier equations for Mathematica, and should be ready to try them out tomorrow.
2. I modified the PAM simulation with a coarser-grained histogram, but more timing offset values, in order to see how the decision variate varies with timing offset. The results should be available in the morning.

A.4.2 *22/10/13 - More Gram-Charlier series*

The simulation results showed that the decision region boundaries did decrease with timing error, however the histogram was not fine-grained enough to accurately determine the exact boundary locations, so the simulation was re-run with more bins.

I fixed some bugs in my implementation of the Gram-Charlier series and was able to generate a few plots, which were very similar to those generated by the simulator, albeit with half the amplitude.

A goal for tomorrow is to generate the plots with identical timing offsets to the simulation and compare both plots.

A.4.3 *23/10/13 - Proper Gram-Charlier plots*

The simulation results had been appended to the previous set of results by accident, so the whole thing had to be run again for tomorrow. On a more positive note, I noticed a missing power in my implementation of the Gram-Charlier series, and the plots are now a lot closer to those generated previously.

A.4.4 *Week 4 Summary*

I implemented the Gram-Charlier series and was able to compare the results from the simulations to the Gram-Charlier series. These are close, but not exact, so we will have to look closely at where the differences may be coming from.

A.4.5 *Goals for Week 5*

- I will make use of the long weekend to run some extra-long simulations and compare these to the Gram-Charlier series.

A.5 WEEK 5

A.5.1 *29/10/13 - Comparing Gram-Charlier to Simulation*

The simulations ended, and I was able to compare simulated and Gram-Charlier PDF plots. I extracted a rough estimate of the decision region boundaries given by both methods and compared them to the corresponding values of $2g(\Delta)$, and found very close correlation.

A.5.2 *30/10/13 - Applying the Tikhonov Distribution*

I was able to implement the Tikhonov Distribution using the equation provided in *PAMTikhonov.pdf*:

$$F_\Delta(y) = \frac{\mathrm{Exp}\left[\frac{\cos(2\pi y)}{(2\pi\sigma_\Delta)^2}\right]}{I_0\left(\frac{1}{(2\pi\sigma_\Delta)^2}\right)} \text{ where } -\frac{1}{2} \leqslant y \leqslant \frac{1}{2}$$

Given these timing error probabilities and the optimum decision region boundaries for each timing error, I calculated the overall optimum decision region boundary for each timing error probability distribution using

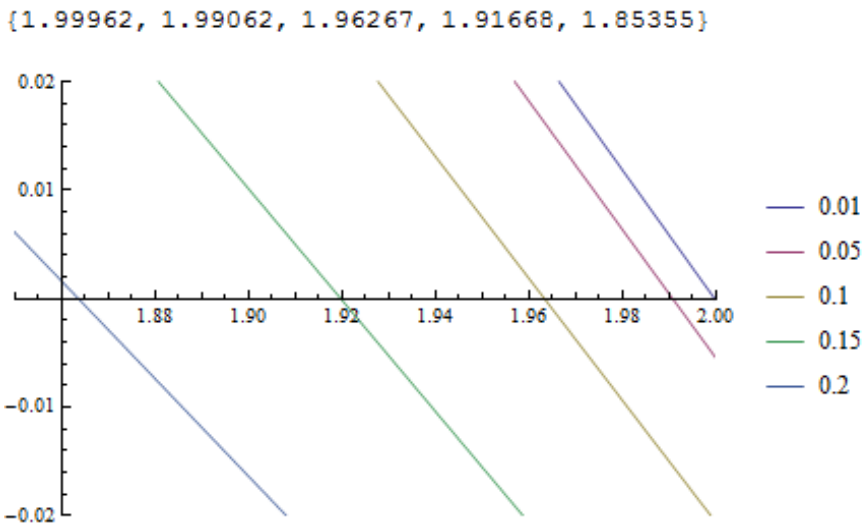


Figure 16: Gram Charlier approximation of $P(\omega_0 = 1, R) - P(\omega_0 = 3, R)$

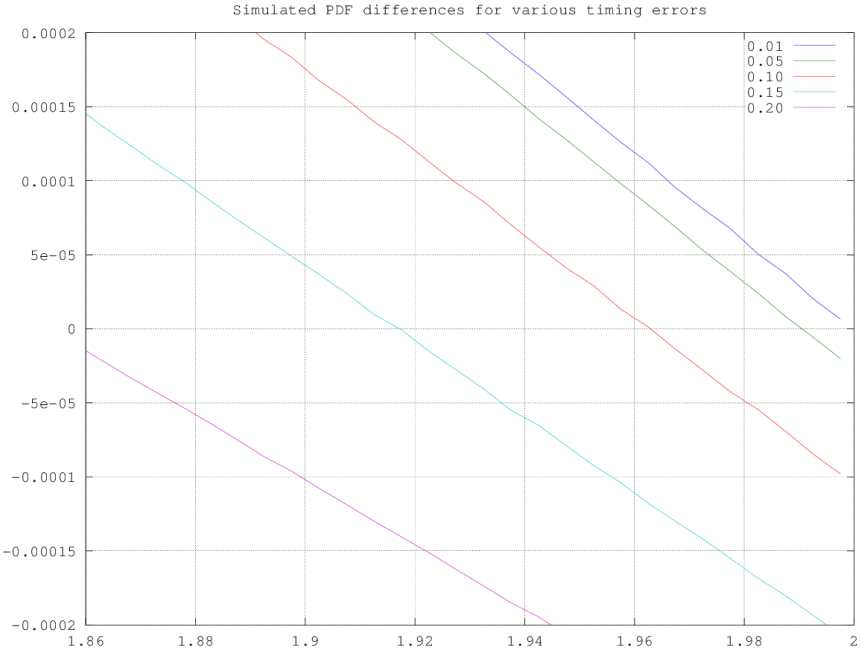


Figure 17: Simulation of $P(\omega_0 = 1, R) - P(\omega_0 = 3, R)$, N=$3 \times 10^6$

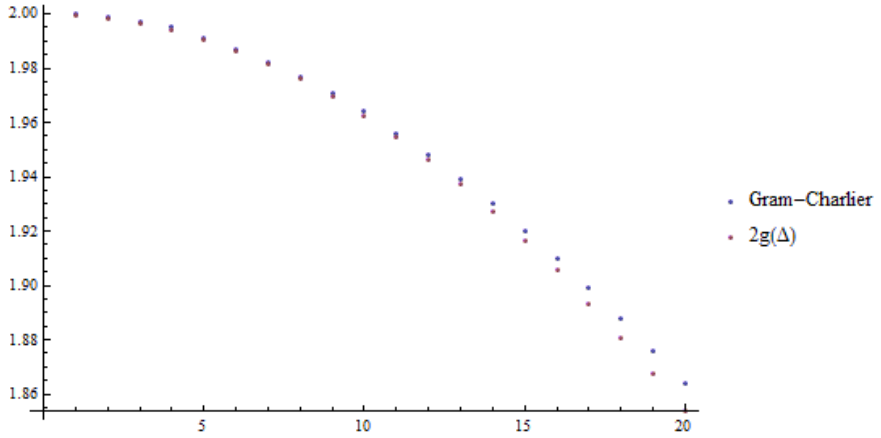


Figure 18: Comparison of Gram-Charlier Decision Region Boundaries and $2g(\Delta)$ estimation ($0.01 \geqslant \Delta \geqslant 0.2$)

$$B_{OPT} \sim \sum_{\Delta} P(\Delta) B_{OPT,\Delta}$$

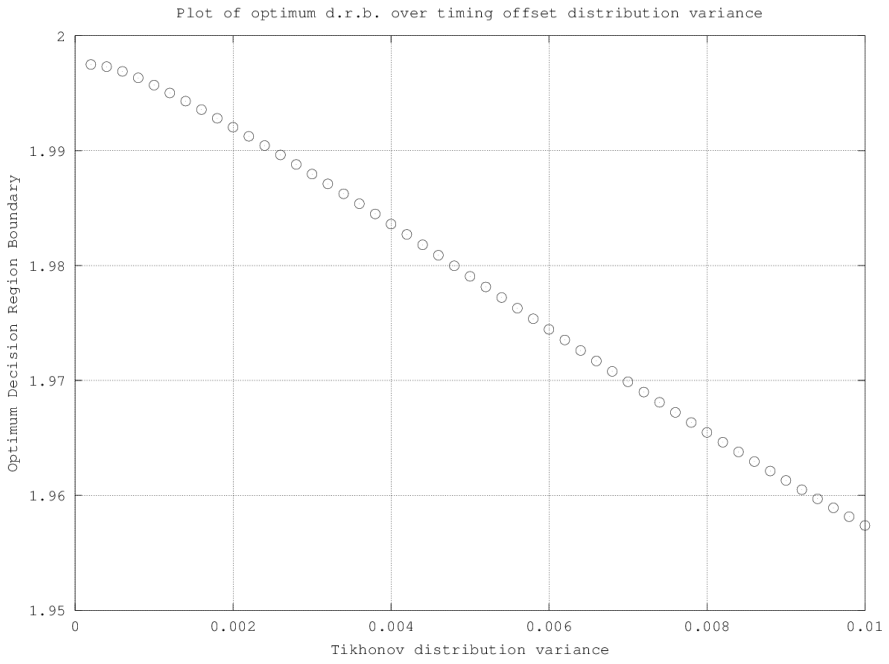


Figure 19: Optimum Decision Region Boundary for various timing error probability distributions

It is important to note that with increasing variance, the probability density function places more weight on larger timing errors outside the range simulated, so these results are less accurate for higher variances.

A.5.3 *Week 5 Summary*

In week 5, I calculated the optimum decision region boundary for a range of timing offsets, through simulation and the Gram-Charlier approximation. I demonstrated a close correlation between these boundaries and the $2g_k$ term. A slight difference between the Gram-Charlier approximation was found to be due to a typo in its implementation. I applied the Tikhonov distribution to the calculated optimum decision region boundaries for each timing offset, in order to calculate an optimum decision region boundary for a given Tikhonov distribution of timing offsets

A.5.4 *Goals for Week 6*

- On the simulation side, a key goal for week 6 is to randomly generate timing offsets according to the Tikhonov distribution and apply these to the simulation as timing offsets, in order to verify correlation with the Gram-Charlier and $2g_k$ approximations.
- A typo in the Gram-Charlier implementation has been found and corrected, and it would be interesting to see if this approximation matches $2g_k$.

A.6 WEEK 6

A.6.1 *04/11/13 - Fixing errors*

David Murphy took a look at my code and spotted errors which I fixed. The fixed Gram-Charlier implementation was found to match $2g_k$ very closely. The fixed simulation was left to run overnight; unfortunately Mathematica 6.0 running on the Unix machines was unable to run it, so the number of points had to be reduced.

A.6.2 *05/11/13 - Corrected simulation results*

The produced PDFs were too inaccurate to properly calculate the zero crossing points, so the simulation will have to be run over several days.

A.6.3 *Week 6 Summary*

A simulation was constructed that generated timing error offsets according to a Tikhonov distribution of predetermined variance, and used to produce received symbol PDFs. The simulation was found to run very slowly, and could only be run on Mathematica 9. Ger

Houten has been asked whether it would be possible to upgrade the Unix machines to this version and he will look into it.

A.6.4 *Goals for Week 7*

- Continue running the simulation, trying to speed it up if at all possible.

A.7 WEEK 7

A.7.1 *11/11/13 - Returning to the UNIX machines*

The UNIX machines were upgraded to Mathematica 9 over the weekend, so I was able to port the code in order to run off these. In addition, David Murphy suggested that I look into parallelizing the code. Since these were dual-core machines I was able to make use of Mathematica's `ParallelTable` function to reduce run times a little. The simulation will have to run over several days, however, as the expected deviation in optimum decision region boundary is very small.

A.7.2 *19/11/13 - Day 9 of Week 7*

After several days of running the simulations, we found the optimum decision region boundaries given by the simulations, in red, converged to roughly those predicted by averaging the optimum decision region boundary of a timing offset over the Tikhonov distribution of timing offsets, in blue, given by the equation:

$$B_{OPT} \sim \sum_{\Delta} P(\Delta) B_{OPT,\Delta}$$

A.7.3 *Week 7 Summary*

Simulations supported the theory that the optimum decision region boundary in the presence of statistically distributed receiver timing errors will decrease from the expected value. Additionally, it was shown through simulation that the new optimum decision region boundary can be approximated, assuming a known distribution of these timing errors, by averaging the optimum decision region boundary given each timing offset over the distribution of timing offsets.

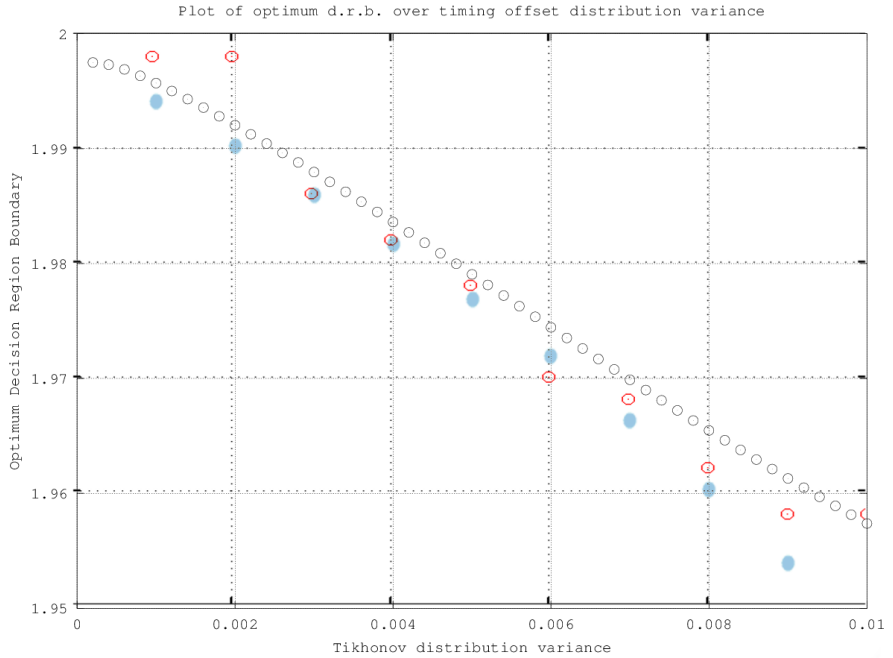


Figure 20: Optimum Decision Region Boundary for various timing error probability distributions. Black: estimate found by averaging fixed timing errors over Tikhonov distribution; Blue: analytical result determined by David Murphy; Red: simulation results.

A.7.4 *Goals for Week 8 onwards*

- Verify that the Gram-Charlier series provides an adequate approximation to the received symbol PDF in the presence of timing errors.
- Provide numerical values for the probability of error P_e in the presence of a distribution of timing errors.

A.8 WEEK 8

A.8.1 *Week 8 Summary*

After spending the Christmas break reviewing the literature on Rayleigh fading, mainly *Proakis*, I implemented the Rayleigh distribution as a modified Nagakami-n distribution I implemented this as well as a multi-receiver combining system and added it to the existing simulation. Following a meeting with David Murphy and Colin Murphy, we decided to proceed as follows:

The Rayleigh distribution is a special case of the Nagakami-n distribution, with n=1

- Run simulations similar to those run before Christmas to determine numerically the optimum decision region boundaries and corresponding error probabilities in the presence of Rayleigh fading with Equal Gain Combining (EGC).

- Alongside this, attempt to provide an analytical derivation for the optimum decision region boundaries in the presence of timing errors in a non-fading environment. Should the above simulations show merit in modifying the decision region boundaries, this could then be adapted to the fading case later on.

A.8.2 *Goals for Week 9*

- Implement the simulation. This is expected to run for the duration of Week 8, and demonstrate the optimum decision region boundaries for an EGC receiver with Rayleigh fading. The main goal is to determine if different decision region boundaries would reduce the probability of error.
- Attempt to describe the system analytically, ignoring fading. This will be based on a Gram-Charlier approximation.

A.9 WEEK 9

A.9.1 *13/01/14 - Examining combining and fading*

Over the weekend, three separate approximate simulations were run to examine receiver performance in three cases:

1. Similar to before Christmas, a single-receiver system in a non-fading environment
2. A multi-antenna system using EGC in a non-fading environment
3. An EGC system in the presence of Rayleigh fading

Initial simulations for case 3. showed reduced optimum decision region boundaries in the presence of Rayleigh fading, which appeared to remain constant with changing timing offset variance. A simulation of case 2. showed changing optimum decision region boundaries for different timing offset variances. A more detailed simulation of case 3. examining more variances was commenced, and is expected to finish mid-week.

A.9.2 *14/01/14 - Planning an analytical analysis of the non-fading case*

I found an implementation of the Gram-Charlier series and Tikhonov distribution I had developed previously that could be used as a basis for an analytical exploration of the project topic. The code produces a Gram-Charlier PDF for a range of timing offsets, calculates the optimum decision region boundary for each, and averages these boundaries over the Tikhonov distribution. David Murphy suggested that while this approach wasn't mathematically correct, if the Gram-

Charlier PDF's were averaged over the Tikhonov distribution to provide an overall PDF, the optimum decision region boundary could be calculated from this.

A.9.3 *15/01/14 - Evaluating the Gram-Charlier distribution over the Tikhonov distribution*

I carried out the changes detailed above, and added a loop to numerically estimate the location of the PDF crossings, thereby estimating the optimum decision region boundaries. Unfortunately there was not enough time to run the code over all possible conditions.

One of the simulations begun at the start of the week quit unexpectedly, and had to be restarted, moving back the expected end-date for the simulations to Friday.

A.9.4 *16/01/14 - Results for the above*

I was able to run the Tikhonov-Gram-Charlier code described above and thus plot the optimum decision region boundaries for Tikhonov-distributed timing offsets, assuming a Gram-Charlier approximation for the received symbol. These correlated strongly with the optimum boundaries found through simulation.

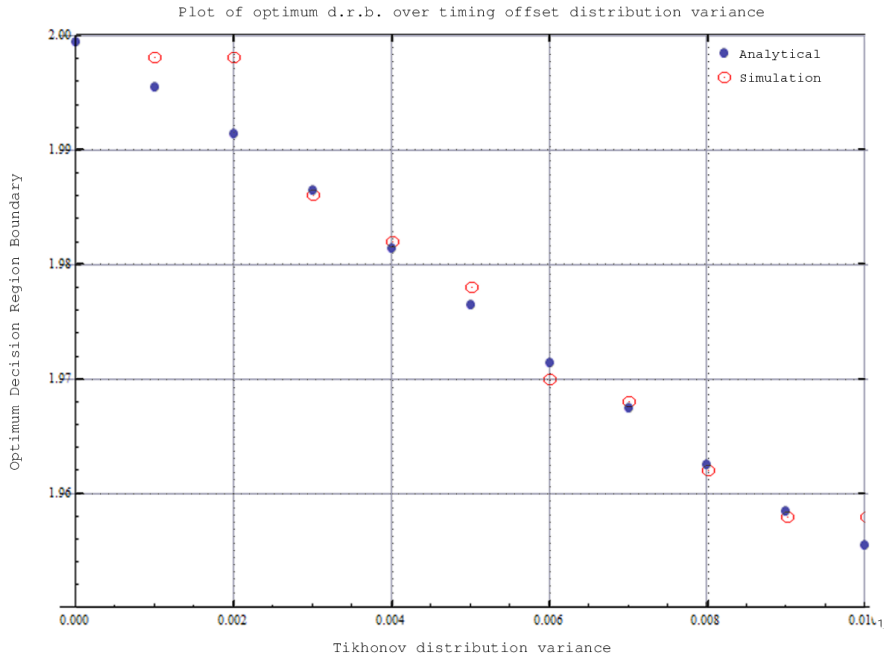


Figure 21: Plot of optimum decision region boundaries versus Tikhonov-distributed timing error variance, from Gram-Charlier approximation and simulation respectively

A.9.5 *Week 9 Summary*

A simulation covering the 4-PAM case studied before Christmas was extended with a 4-channel EGC system and simulated Rayleigh channel fading. This simulation was run over the course of the week and (*proved xyz...*)

A rough analytical study of the non-fading timing error model examined before Christmas was carried out. The PDF's of various cases with a fixed timing offset were generated, and averaged over a discretized Tikhonov distribution to estimate the PDF of the system in the presence of Tikhonov distributed timing offsets. This was carried out for a range of Tikhonov distribution variances to examine different conditions. The optimum decision region boundary for each condition was found numerically and found to correlate strongly with the locations determined through simulation previously, with the exception of some low-variance points.

A.9.6 *Goals for Week 10*

- Examine the performance of the system in the presence of Rayleigh distributed channel fading.
- Determine the probability of error of the system in the non-fading case, and evaluate any performance gains from using the calculated optimum decision region boundaries.

A.10 WEEK 10

A.10.1 *20/01/14 - Rayleigh Fading Results*

Simulation ended over the weekend and showed an optimum decision region boundary around 1.72, for Tikhonov variances of 0.001 to 0.010. This is an interesting result, as it suggests that in this particular case the optimum decision region boundary is only loosely related to the Tikhonov variance, which would imply that knowledge of the timing statistics is less important in implementation than previously thought.

David Murphy produced an analytical solution for the optimum decision region boundary in the non-fading case, which demonstrated clear BER gains (20-33%).

A.10.2 *21/01/14 - Implementing David's suggestions*

Following some consideration, Dave suggested that the SNR of 8dB used in the simulation was too low for 4-PAM, and it's possible that the ODRB of 1.72 seen was the lower bound for the ODRB. We settled on a more reasonable SNR of 20dB, and decided to apply each set of

random conditions to both possible sent symbols $\omega_0 = 1{,}3$ so that we would only have to run the simulations once. Another suggestion made by David Murphy was to apply a different timing offset to each channel, as each channel has a separate receiver, and therefore independent timing. I also implemented Maximal-Ratio Combining as an option in order to compare the performance of both systems.

A.10.3 *Week 10 Summary*

This week we were able to show reduced optimum decision region boundaries in the presence of Rayleigh fading with Equal-Gain Combining. These results showed that the performance of the EGC receiver could be considerably improved when facing low SNR. It was decided that the short-term goal would be to evaluate the receiver's performance with a more reasonable SNR, and compare it to the performance of a Maximal-Ratio Combining system under similar conditions.

Work stopped mid-week to facilitate the author's funding application, and is expected to resume in full next week.

A.10.4 *Goals for Week 11*

- Evaluate the error rate of an EGC receiver with optimised decision region boundaries in Rayleigh fading with an SNR of 20dB.
- Similarly, evaluate the error rate of a MRC receiver in similar conditions.
- Compare and contrast the performance of each, taking into account the higher area and power requirements of the latter.

A.11 WEEK 11

A.11.1 *27/01/14 - Simulating EGC and MRC*

The changes discussed in the last entry were implemented, and both simulations started.

A.11.2 *30/01/14 - EGC vs MRC results*

These simulations took a little longer than expected due to the extra processing required for MRC. Error rates were found to decrease after redefining the decision region boundary of the EGC receiver. The MRC system suffered more from the effects of the timing error, but the reduced EGC error rates were still a good bit higher than the equivalent error rates using MRC. Nonetheless, a 20-33% improvement was found over the un-tweaked decision region boundaries.

Redefining the decision region boundary for the MRC also showed improvements, although these were a more modest 7-20%.

Tikh var	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009	0.01
Prev ER	0.111	0.115	0.119	0.124	0.129	0.133	0.139	0.148	0.154	0.159
New ER	0.0751	0.0801	0.0844	0.0900	0.0951	0.0994	0.106	0.115	0.122	0.126
% drop	33.	30.	29.	27.	26.	25.	24.	22.	21.	20.
MRC ER	6.00×10^{-8}	0.0000298	0.000354	0.00121	0.00320	0.00565	0.00943	0.0146	0.0193	0.0265
MRC ER	8.00×10^{-8}	0.0000159	0.000218	0.000970	0.00246	0.00470	0.00818	0.0129	0.0179	0.0242
% drop	−33.	47.	38.	20.	23.	17.	13.	12.	7.	9.

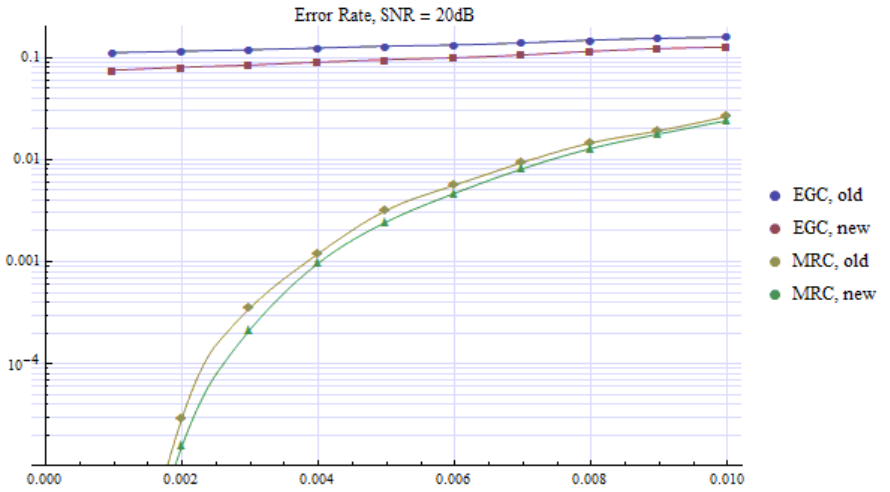


Figure 22: EGC and MRC error rates, SNR=20dB

A.11.3 *01/02/14 - SNR increase to 28dB*

In order to determine how much of an effect noise has on the system in the presence of fading, the simulation was restarted with an SNR of 28dB.

A.11.4 *Week 11 Summary*

The simulations ran this week proved that an error rate reduction could be obtained by merely redefining the location of the decision region boundaries, in the Rayleigh fading case. Although the EGC system could not be made to perform with comparable performance to the MRC system, we demonstrated rapidly deteriorating MRC performance with more severe timing offsets, vindicating our study of the issue.

A.11.5 *Goals for Week 12*

- Examine other conditions to gain a more general understanding of the effects of timing offsets in Rayleigh fading channels.
- Start on the analytical examination of timing offset.

- Compile a presentation for the seminars.

A.12 WEEK 12

A.12.1 *03/02/14 - Presentation work*

The results from the 28dB run showed very similar error rates, confirming my suspicion that the effects of fading are dominating over the effects due to AWGN seen in the non-fading case. A simulation at 12dB was started to see if this remained true at lower SNR's.

Tikh var	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009	0.01
Prev ER	0.111	0.115	0.119	0.123	0.129	0.134	0.140	0.148	0.153	0.159
New ER	0.0751	0.0797	0.0847	0.0883	0.0954	0.100	0.108	0.116	0.119	0.126
% drop	33.	31.	29.	28.	26.	25.	23.	21.	22.	20.
MRC ER	4.00×10^{-8}	0.0000185	0.000286	0.000937	0.00287	0.00534	0.0101	0.0131	0.0188	0.0282
MRC ER	8.00×10^{-8}	0.0000109	0.000168	0.000776	0.00219	0.00446	0.00886	0.0121	0.0172	0.0245
% drop	−100.	41.	41.	17.	23.	16.	12.	8.	8.	13.

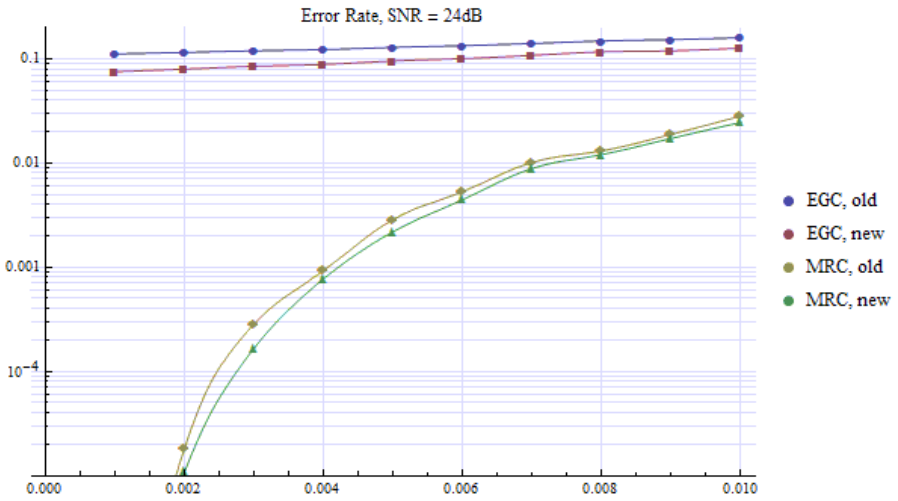


Figure 23: EGC and MRC error rates, SNR=28dB

The seminar presentation was also compiled.

A.12.2 *06/02/14 - Presentation test run and rework*

The presentation was given a test-run, leading to improvements in the clarity of the descriptions.

The simulation ended, but unfortunately it turned out that the EGC simulation had been ran with an SNR of 8dB, so had to be run again.

David Murphy announced he has been able to start the analytical proof of the timing offset, and we should be able to hear from him at the start of next week. I am planning on using his code as a base to build upon for my project.

A.12.3 *Week 12 Summary*

A relatively quiet week, with a focus on the seminar presentation.

A.13 WEEK 13

A.13.1 *11/02/14 - Analytical description of non-fading case*

David Murphy provided a proof for the Gram-Charlier approximation of the non-fading case, along with some accompanying code. Used the Gauss-Legendre method to combine the Gram-Charlier solution for a fixed timing offset with the Tikhonov timing offset distribution to provide a solution for the PDF at the input to the receiver, which is too long to reproduce here in full, but is approximately given by:

$$\frac{b-a}{2}\sum_{i=1}^{N}\omega_i\, T\left(\frac{b-a}{2}z_i-\frac{b+1}{2}\right)\, GC\left(\frac{b-a}{2}z_i-\frac{b+1}{2}\right)$$

where $\omega_i = \dfrac{2}{(1-z_i)(P(z_i))^2}$ and z_i are the roots of the Legendre polynomial $P_n(z)$.

I spent some time examining the Mathematica code he used to examine the above, and decided as an introduction to try to extend it to the Equal-Gain comining case. I used the property that for the PDF of the sum of two independent variants is the convolution of each variant's PDF (ie. for two variants X_1 and X_2 with PDF $f_1(x)$ and $f_2(x)$, the PDF of $Y = X_1 + X_2$ is $f_Y(x) = f_1(x) * f_2(x)$) to express the PDF of an m+1 antenna EGC system as

$$f_{EGC}(x) = f^{m*}(x)$$

where $f(x)$ is the PDF at each antenna and $f^{2*}(x)$, for example, is the double convolution $f(x) * f(x) * f(x)$).

I was then able to use the Gauss-Legendre method to describe this as an m-fold sum:

$$f_{EGC}(y) = f^{m*}(y) \simeq \sum_{a_1=1}^{n_f} \cdots \sum_{a_m=1}^{n_f} 3^m \omega_{a_1} \cdots \omega_{a_m} \times$$
$$f(3z_{a_m} - 3z_{a_{m-1}}) \cdots f(3z_{a_2} - 3z_{a_1})\, f(3z_{a_1} + 2)\, f(y - 3z_{a_m} - 2)$$

I was unfortunately unable to test this code before end of day.

A.13.2 *12/02/14 - Viability of independent variate route*

During an email conversation with David Murphy, he suggested that the convolution route might be difficult to implement in terms of processing requirements, and suggested instead that in the case of the MRC system, the independence of the timing error and fading statistics of each receive channel means that a "joint" PDF describing the probability of all timing and fading variables can be constructed by taking the product of each timing and fading PDF. Therefore we can derive a Gram-Charlier distribution conditioned on a particular set of timing offsets and fading factors, calculate the BER of this theoretical system, and average this over the distribution of timing offsets and fading factors described by the joint PDF.

A.13.3 *13/04/14 - MRC adaptation of Gram-Charlier distribution*

I went over the process described yesterday in more detail with David Murphy to ensure I fully understood it, and began to implement it by replacing all the $g_k(\Delta)$ terms in his implementation of the non-fading Gram-Charlier series with $\alpha_1^2 g_k(\Delta_1) + \alpha_2^2 g_k(\Delta_2)$ to reflect the effects of fading with a 2 antenna system. In this case, the expected (synchronized) DRB which was previously simply $2g_0(0) = 2$ is now given by $2(\alpha_1^2 g_0(0) + \alpha_2^2 g_0(0)) = 2(\alpha_1^2 + \alpha_2^2)$.

Testing this model, I noticed oscillations at the tails of the distributions which made it impossible to calculate the optimum DRBs, as there were multiple PDF crossings.

A.13.4 *14/04/14 - Explaining the oscillations*

I raised the issue with David Murphy, and it turned out that I had forgotten the effects of fading on the channel Gaussian noise. The variance of this noise is scaled by the effects of noise, in this case by $(\alpha_1 + \alpha_2)$. Making this correction I was able to determine the optimum DRB to be equal to $2(\alpha_1^2 g_0(\Delta_1) + \alpha_2^2 g_0(\Delta_2))$, as expected.

A.13.5 *Week 13 Summary*

In this week I examined David Murphy's derivation and modelling of the non-fading case using the Gram-Charlier series, and make a start on my own piece of the analytical work. I successfully extended his derivation to describe the decoder input PDF for a 2 antenna MRC system given known fading statistics and timing offsets, which puts me in a good position to extend this to a more thorough analytical exploration of receiver performance with diversity.

A.13.6 *Goals for Week 14*

- Evaluate the BER for given fading and timing statistics.
- Average the BER over fading and timing offset distributions to determine the average BER for a given system.

A.14 WEEK 14

A.14.1 *16/02/14 - Bit Error Rate Calculation*

I implemented the BER calculation for fixed fading and timing statistics as two integrals over the Gram-Charlier PDF approximations, using a given decision region boundary to bound the integrals. Using this I was able to determine the BERs using both the sub-optimum decision region boundaries $2(\alpha_1^2 + \alpha_2^2)$ and the optimum decision region boundaries calculated using last week's code.

A.14.2 *18/02/14 - Averaging BER over Tikhonov and Rayleigh distributions*

The code was extended to average the BER for each timing error and channel gain over Tikhonov and Rayleigh distributions, respectively, using Gauss-Legendre Quadrature. In this way the average BER for given timing offset and fading statistics can be calculated. Progress was slow as some of the previous code has to be rewritten to fit with its new application. Gauss-Legendre Quadrature was found to be unsuited to the Rayleigh distribution, for reasons I am not yet sure of, so a discrete sampling of the distribution was used instead. At the end of the day the *FindRoot* method used to determine the optimum decision region boundaries refused to converge for certain ranges of timing error and channel gain, so more work must be done to finish it off.

A.14.3 *19/02/14 - Root-finding tweaks*

I noted that the *FindRoot* method had difficulty converging for both very high and very low channel gains, as the optimum decision region boundary in these cases are much higher or lower than usual. Since the nature of the Rayleigh distribution dictates that these extreme gains are still likely, I used the observation that the gain can be roughly given by $2(\alpha_1^2 + \alpha_2^2)$ for low timing error offsets, and instructed the *FindRoot* method to start the search at this point. This improved both the speed and possibility of converging.

The algorithm was still having difficulty converging for any significant timing error offset, so I greatly reduced the accuracy requirements of the root finding algorithm, under the understanding that

we are dealing with average error rates and the goal is to show the rough gain in performance. This brought the algorithm under control, but nonetheless some issues remained:

- For very large timing error offsets, the algorithm converged to an unlikely value. This is unimportant, as in these cases the system has failed to perform any sort of detection.
- The response is somewhat oscillatory. This could be due to the accuracy of the Gram-Charlier series, or less likely the *FindRoot* method.
- Even more complexing, the algorithm seems to fail to converge for certain combinations of timing error offset. At first glance it looks like there is a pole or zero at these points. Again, perhaps Dave could suggest a fix for these points.

Insert timing error 2D density plot

A.14.4 *20/02/14 - Discussion of Gram-Charlier accuracy*

I discussed the issue described above with David Murphy and Dr. Colin Murphy, and David Murphy suggested increasing that it could be a machine precision problem, and that increasing the precision of the variables used in the Gram-Charlier series calculation could solve the problem. Alternatively, Dr. Colin Murphy suggested that an acceptable answer could be obtained by truncating the sweep to remove the areas where the *FindRoot* algorithm does not converge.

A.14.5 *21/02/14 - Overcoming the timing error problems*

Through some exploration, the random non-convergent regions were found to disappear at higher AWGN variances. I therefore rose the SNR to 20dB, and found that the oscillations in the root estimates also disappeared.

A.14.6 *22/02/14 - Further problems for high timing offsets*

I observed that the root finding algorithm failed to converge for large timing error offsets. Noting that it definitely converged for timing offsets of 0.3 and below, and that timing offsets greater than this were extremely unlikely, I decided to truncate the timing offset averaging to $-0.3 \leqslant \Delta \leqslant 0.3$.

Plot of roots verses timing error

After implementing this I discovered that the root-finding algorithm results still became increasingly inaccurate for higher timing offsets. This will require further investigation to determine its cause, as left untouched it could significantly reduce the accuracy of the model.

A.14.7 *Week 14 Summary*

This week saw the implementation of Bit Error Rate calculations for given fading and timing variates, and the averaging of these over the timing and fading probability distributions was the final step in the implementation of a mathematical model for the described MRC system and allowed a figure for the average BER of the system to be determined. While testing the corners of the implementation, it was found that the analytical approximation failed at certain timing offsets at low SNR values, but this was ignored as larger SNR values are of interest to us. Additionally the optimum DRB finding algorithm does not converge for higher timing error offsets, leading to the decision to ignore the statistically unlikely larger timing offsets. More work will have to be done on reducing the error of the algorithm at middle timing offsets, however.

A.14.8 *Goals for Week 15*

- Improve the accuracy of the model for large timing error offsets.

A.15 WEEK 15

A.15.1 *24/02/14 - Root-finding-defying wiggles*

I found that the source of the root-finding errors is in "wiggles" near the roots of the $f_1(y) = f_3(y)$ function used to find the optimum DRB. These seem to be caused by inaccuracies in the Gram-Charlier approximation. Increasing the numerical accuracy did not seem to have any difference, and increasing the length of the Gram-Charlier series only made the oscillations much worse.

A.15.2 *25/02/14 - Correct optimum DRB measurements*

A brief email exchange with David Murphy helped me understand the source of the problem. In the model I had built, it was assumed that the instantaneous channel gains and timing offsets were known, and used to make an estimate of the optimum DRB, and the resulting BER was averaged over the fading and timing error PDFs to obtain the average BER. However as mentioned earlier, while the instantaneous channel gains are known in a MRC system (but not an EGC system!), the timing offsets aren't. Hence, the correct approach is to obtain an average PDF for each pair of channel gains, deduce the optimum DRB and BER for each of those channel gains, and then average over the fading and timing error PDFs. Therefore the "wiggles" seen before will be reduced when averaging over the Tikhonov, and shouldn't be present when the optimum DRB is calculated.

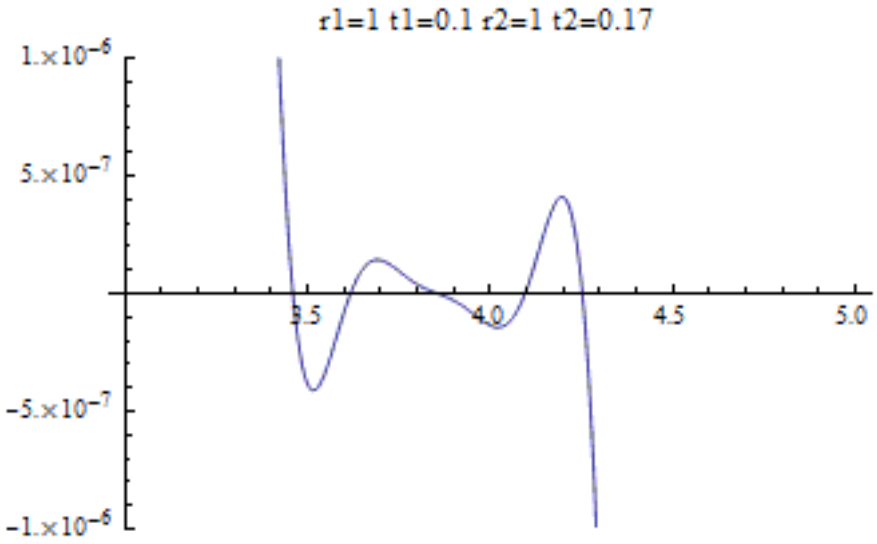


Figure 24: "Wiggles" near the roots

I implemented the averaging of the GC PDFs over the Tikhonov timing offset distributions to provide an overall PDF for given channel gains and Tikhonov variances, and from these determined the optimum decision region boundaries. These are now used to determine the BER for different channel gains and timing offsets, which are averaged over the Rayleigh and Tikhonov distributions to determine the average BER.

A.15.3 *26/02/14 - Integration*

After a suggestion from David Murphy, I tried implementing a closed-form solution for the integration of the Gram-Charlier PDF approximation, given solutions for the integration of other identities. In this manner I hoped to significantly reduce the time to determine the average BER.

Starting with the Gram-Charlier definition described earlier,

$$f_{X|\Delta}(y) = \frac{1}{\sigma_X}\,\phi\left(\frac{y-\mu_X}{\sigma}\right) + \sum_{m=2}^{M} \frac{\alpha_{2m}}{(2m)!\sigma_X^{2m}} \left[\frac{1}{\sigma_X}\,\phi\left(\frac{y-\mu_X}{\sigma}\right) H_{2m}\left(\frac{y-\mu_X}{\sigma_X}\right)\right]$$

and incorporating the given identites,

$$\int_{-\infty}^{x} \frac{1}{\sigma}\,\phi\left(\frac{y-\mu}{\sigma}\right) dy = \frac{1}{2}\left(1+\operatorname{erf}\left(\frac{x-\mu}{\sqrt{2}\sigma}\right)\right)$$

$$\int_{-\infty}^{x} \frac{1}{\sigma}\,\phi\left(\frac{y-\mu}{\sigma}\right) H_m\left(\frac{y-\mu}{\sigma}\right) dy = -\phi\left(\frac{x-\mu}{\sigma}\right) H_{m-1}\left(\frac{x-\mu}{\sigma}\right)$$

the solution for the integral of the Gram-Charlier series was found to be:

$$\int_{-\infty}^{x} f_{X|\Delta}(y) = \frac{1}{2}\left(1+\operatorname{erf}\left(\frac{x-\mu_X}{\sqrt{2}\sigma_X}\right)\right) - \sum_{m=2}^{M} \frac{\alpha_{2m}}{(2m)!\sigma_x^{2m}}\ \phi\left(\frac{x-\mu_X}{\sigma_X}\right) H_{2m-1}\left(\frac{x-\mu_X}{\sigma_X}\right)$$

I found that while the resulting equation approximated the result found using numerical integration (*NIntegrate*), the two did not match exactly. An interesting question is which correct. My own gut feeling is that the value found using *NIntegrate* is correct, as the algorithm converges to a very precise result and I have very little confidence in my own mathematical abilities, so for the moment I will stick with it. Then again, it is possible that numerical integration is reflecting some unforeseen inaccuracies in my implementation of the Gram-Charlier series, and the closed-form solutions, being mathematically-based, are accurate.

A.15.4 *Week 15 Summary*

In week 15 I fixed a significant inaccuracy in the model I had built to determine the average BER, stemming from a misconception I had acquired from getting caught up with implementation and forgetting the top-level picture. The numerical implementation of the analytical model seems to be close to finished, and a few tentative tries showed poorer results than those found through simulation. Hopefully work next week will produce the definitive model alongside some quantitative results.

A.15.5 *Goals for Week 16*

- Complete the implementation of the analytical model.
- Try a few test runs to determine the performance of the standard and modified MRC system.

A.16 WEEK 16

A.16.1 *03/03/14 - Rewriting analytical implementation*

My machine turned off on Friday afternoon, and I returned from the weekend to find that my *Mathematica* notebook had been corrupted. After some time trying to manually fix the corrupted file I decided the task was too large and reverted to a week-old backup of the file. As much work had been done in the time between I took the opportunity to rewrite much of the implementation to make it clearer.

A.16.2 *04/03/14 - Strengthening the implementation*

Some time was spent improving the speed of execution of the implementation and testing it at intermediate points to ensure the results given made intuitive sense. Currently the implementation takes roughly 25mins to run for 5 timing error and channel gain points per channel.

A.16.3 *05/03/14 - Assessing the accuracy of the implementation*

I spent some more time tweaking the accuracy and precision parameters of the *FindRoot* and *NIntegrate* function to reduce the speed of execution without compromising the accuracy of the system. By looking through the intermediate results of the implementation I found that inaccuracies in the Gram-Charlier approximation for high timing offsets are creating such large error rate estimations for the traditional decision region boundaries that their reduced weighting is insufficient to prevent their effects appearing in the averaged BER value. Unfortunately I cannot imagine how this could be mitigated without ignoring more than the most trivial timing offsets, which wouldn't allow us to show off the full utility of out method.

I added memory to the Gram-Charlier distribution definition to try to speed up the implementation a little more, and started a sweep of timing error variances.

A.17 WEEK 17

A.17.1 *14/03/14 - Pre-open day progress*

The past weekend was spent trying to account for differences between the analytical and simulation models. Both successfully demonstrated reduced optimum decision region boundaries in the presence of timing errors; however the reduction is much more severe in the simulation than using the analytically derived results. Since I was hoping to

be able to derive the optimum DRB values analytically, this was and still remains a major stumbling block.

David Murphy suggested a modification to the *FindRoot* method which mixes Newton's Method and the Bisection Method in order to prevent divergence, however since this isn't currently an issue I doubt I'll have time to implement it.

A.18 WEEK 18

A.18.1 *17/03/14 - Comparing Analytic and Simulation results side-by-side*

I spent the weekend combining the conditional analytical and simulation code to provide comparisons between both methods. I found that the analytic solution grossly underestimated the spread of the received symbol, suggesting an issue with the SNR calculations. I was able to confirm that while the analytic method broke down somewhat at higher timing offsets, the response remained somewhat approximate to the pdf generated through simulation. Overall, it did demonstrate similar characteristics to the simulation, just not to the same extent.

Suspecting that this meant that there was a simple mistake in calculating the parameters of the Gram-Charlier approximation, I made a simplified version and asked David Murphy to look over it.

A.18.2 *18/03/14 - Fixing the Gram-Charlier code*

David Murphy got back to me with two mistakes in the code:

- The variable L represents the number of symbols (4 in the case of 4-PAM), not the number of diversity branches as I had mistakenly believed.
- The SNR in the case of channel fading is gained by the RMS sum of the channel gains ($\sqrt{\sum \alpha_i^2}$) and not the sum of the gains ($\sum \alpha_i$).

Implementing these changes, I saw instant improvements in the correlation between the analytical and simulation PDFs. I also, after some consultation with Dave, rewrote the simulation to make the timing error on each branch independent, as he noted that while both branches run off the same frequency generator, the clock regeneration circuits on each branch will lead to different timing offsets, I restarted both the standalone realistic simulation and the conditional simulation/analytical for 2 diversity branches with the changes made to see how closely both methods match.

A.18.3 *20/03/14 - A much closer match*

The comparison finished, I was able to see that the Gram-Charlier approximation and simulation results now match very closely, even at higher timing offset variances. To double-check that this confirms the Gram-Charlier approximation's suitability for our purposes, I restarted the comparison zoomed into the boundary region, as can be seen below.

A.18.4 *21/03/14 - A slight mistake*

The simulation ended, and I realised that I had accidentally build on the code used to determine the results for the EGC system. Substituting the description of the MRC system, I restarted the simulation.

A.18.5 *22/03/14 - Revised MRC simulation results*

The MRC simulation finished, and I was able to observe that our revised system still gave us gains of up to 14% in the presence of strong timing offsets, for a 2-channel diversity system.

For added precision, I ran the simulation again. I also returned to the analytical description to see if the optimum decision region boundaries could be determined from the latter.

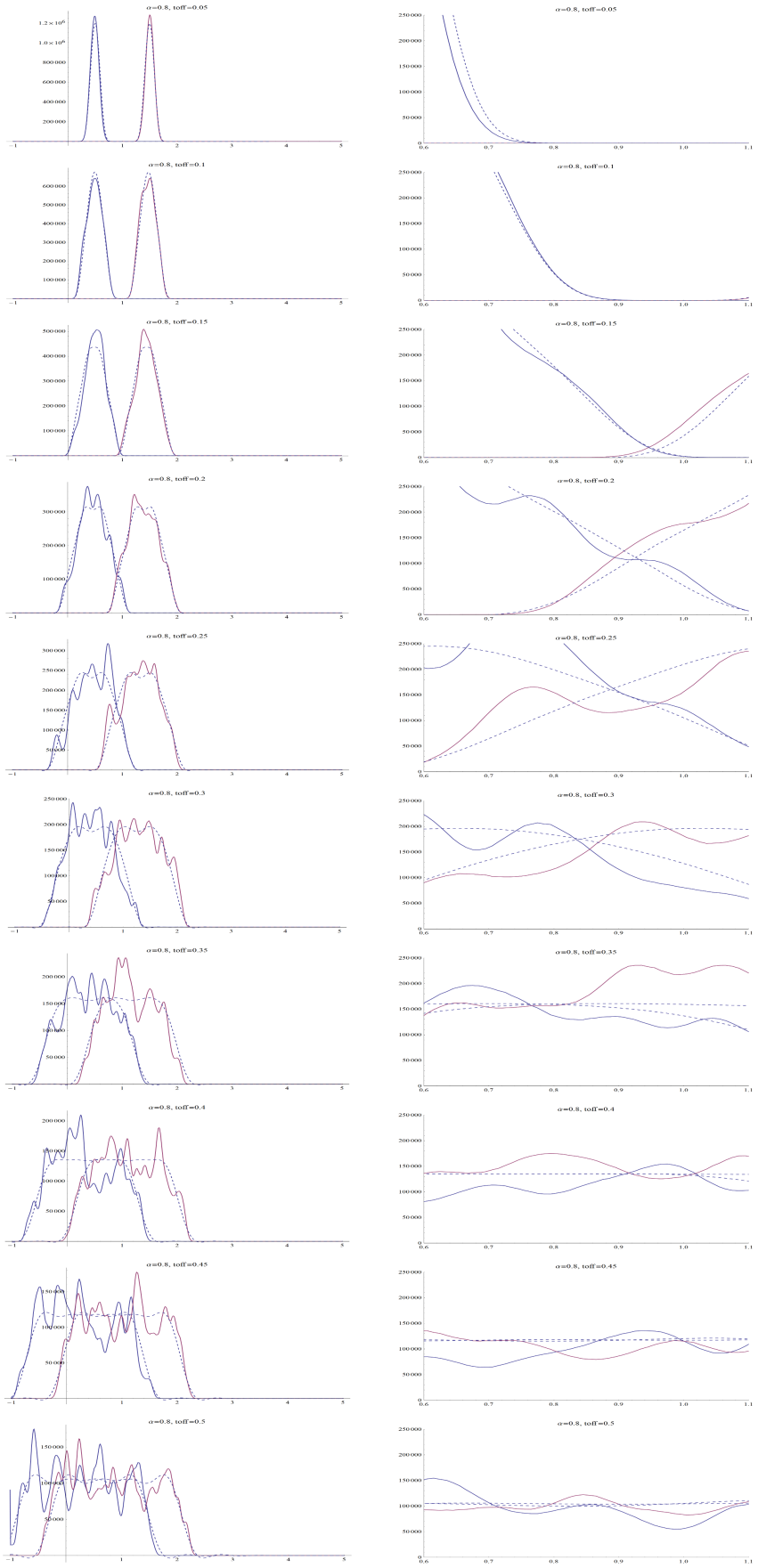


Figure 25: Sample comparison between simulation (solid) and analytic (dashed) received symbol PDF's

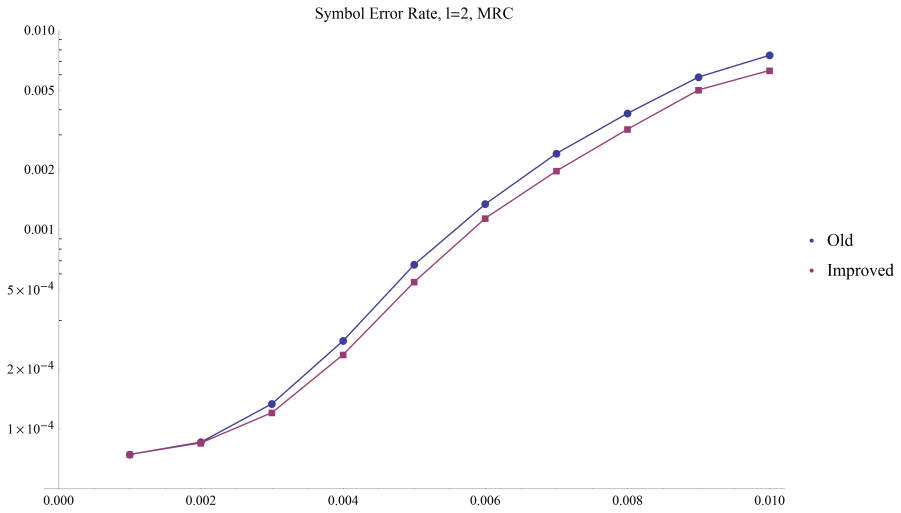


Figure 26: Symbol Error Rate for 2-channel MRC system with Tikhonov-distributed timing error

B

EXAMPLE SOURCE CODE LISTING

TODO

```
l = 4;
be = 0.5;
th = 0;
de = 0.1;
nISI = 20;
snrDB = 8;
ts = 1;
npts = 100000;

(* functions *)
h[t_] := (Sinc[(Pi*t)/ts]*Cos[(Pi*be*t)/ts])/(1 - ((2*be*t)/ts)
    ^2);
g[kk_, de_] := h[ts*(de + kk)];
thikfactor[var_] := thikfactor[var] = 1/((2 \[Pi])^2 var);
thik[var_, y_] := thik[var, y] = Exp[thikfactor[var] Cos[2 \[Pi]
    y]]/BesselI[0, thikfactor[var]];

(* Algorithm (runs 'count' times) *)
f[count_, omega0_, tvar_] := Module[{c = count, w0 = omega0, tv =
     tvar},

   Print[{w0,tv,c}];
   thikdist = ProbabilityDistribution[thik[tv, y], {y, -0.5,
       0.5}];

   (* received symbols *)
   w = Join[RandomChoice[{-3, -1, 1, 3}, nISI/2], {w0},
       RandomChoice[{-3, -1, 1, 3}, nISI/2]];
   k = Range[-nISI/2, nISI/2];
   dtiming = N[RandomVariate[thikdist, {npts}], 8];
   ndtiming = Length[dtiming];
   gk = N[ParallelTable[g[a, b], {b, dtiming}, {a, k}]];
   rk = gk.w;
   snr = 10^(snrDB/20);
   var = N[Sqrt[(l^2 - 1)/(6*Log[2, l]*10^(snrDB/10))]];
   nu = RandomReal[NormalDistribution[0, var], npts];
   r = rk + nu;

   (* histogram *)
   histpts = Range[1.5, 2, 0.002];
   newprob = BinCounts[r, {histpts}];
   filename = "1/output_" <> ToString[tv] <> "_" <> ToString[w0]
       <> ".txt";
```

```
   If[c == 0, oldprob = Array[0 &, Dimensions[newprob][[1]]],
       oldprob = Get[filename]];
   prob = oldprob + newprob;
   Put[prob, filename];];

(* Loop (runs algorithm 100 times with various tikhonov variances
     (tv)) *)
Do[Do[f[ic, 3, tv], {ic, 0, 100}], {tv, 0.001, 0.01, 0.001}]
Do[Do[f[ic, 1, tv], {ic, 0, 100}], {tv, 0.001, 0.01, 0.001}]
```

C

BIBLIOGRAPHY

[1] D. D. Murphy and C. C. Murphy, "A Gram-Charlier series method for calculating general signal constellation error probabilities", *IEEE Transactions on Communications*, Vol. 60, Number 2, p. 300 - 305, 2012.

[2] Y. Yin, J. P. Fonseka and I. Korn, "Sensitivity to timing errors in EGC and MRC techniques", *IEEE Transactions on Communications*, Vol. 51, Issue 4, p. 530 - 534, 2003.

[3] M. K. Simon and M.-S. Alouini, "Simplified noisy reference loss evaluation for digital communication in the presence of slow fading and carrier phase error", *IEEE Transactions on Vehicular Technology*, Vol. 50, Issue 2, p. 480 - 486, 2001.

[4] N. A. Najib and V. K. Prabhu, "Analysis of equal-gain diversity with partially coherent fading signals", *IEEE Transactions on Vehicular Technology*, Vol. 49, Issue 3, p. 783 - 791, 2000.

[5] X Tang, M.-S. Alouini and A. Goldsmith, "Effect of channel estimation error on M-QAM BER performance in Rayleigh fading", *IEEE 49th Vehicular Technology Conference*, Vol. 2, p. 1111 - 1115, 1999.

[6] M. O. Sunay and P. J. McLane, "Probability of error for diversity combining in DS CDMA systems with synchronization errors", *European Transactions on Telecommunications*, Vol. 9, Issue 5, p. 449 - 463, 1998.

[7] M. K. Simon, "A simple evaluation of DPSK error probability performance in the presence of bit timing error", *IEEE Transactions on Communications*, Vol. 42, Issue 234, p. 263 - 267, 1994.

[8] J. Proakis and M. Salehi, *Digital Communications*, 5th ed., New York: McGraw-Hill, 2008.

[9] D. Graffox. (2009, Sep). *IEEE Citation Reference* [Online]. Available: `http://www.ieee.org/documents/ieeecitationref.pdf`