A Collection of Actuarial Papers

An Honors Thesis (HONRS 499)

By

Anthony S. Enk

Kerry Jones

Ball State University

Muncie, Indiana

May 2001

May 2001

TABLE OF CONTENTS

Abstract

This collection of papers consists of three major pieces. The first paper, "Experiences of an Actuarial Student," discusses my college career and what it was like for me to be an actuarial student at Ball State University. The other two papers were opportunities for me to research mathematical topics that I would not have been able to in my regular courses. The first of these, "Mathematical Coding," discusses a couple of methods of mathematical coding, encryption and Huffman coding. The second, "Conway's Theories," discusses three theories and/or algorithms that mathematician John Horton Conway discovered. This thesis was an opportunity for me to not only look back on what I have experienced here, but also research a couple of areas of mathematics I most likely would not have had reason to otherwise.

Acknowledgments

Thank you to Kerry Jones, my thesis advisor, who gave me the chance to write the two research papers. Thanks to Jeff Ackenback for helping me with all of the proofreading and editing of my papers. I would not have been able to put this all together without his help. Thanks also to all the rest of my friends and family who have helped and supported me through this entire process. Last but not least, I would like to thank God, who has given me so much and without Him I would never have had the opportunity to get to where I am today.

Experiences of an Actuarial Student

By Tony Enk

Throughout the past four years as an actuarial science student, I have been through quite a few interesting and educational experiences. Some of these experiences had to do with various courses that I was enrolled in, some dealt with the exam process involved in becoming a Fellow in the Society of Actuaries, and many dealt with job/internship hunting. Furthermore, one of the most memorable experiences I had was my summer internship between my junior and senior years. Some of these experiences taught me quite a bit about what it is going to be like once I become an actuary, while some of them, I think, taught me how I can become a better actuary in the long run.

The first of my experiences that I will talk about will be my classroom experiences, mostly dealing with the variety of courses that I was enrolled in throughout my four years at Ball State. As an actuarial student, I have taken a majority of the mathematics courses offered here, with the exception of the math education courses. For most of my eight semesters, I had at least two mathematics courses, and a couple of semesters I was enrolled in three at a time. This is on top of other course work as well, whether that included other general studies requirements or Honors College courses. Some of these courses included subjects such as Sociology, Humanities, and Western Civilizations. On top of these, I also was studying to get a minor in Insurance, so I had a variety of insurance courses as well. Outside of my required courses, I enrolled in some other courses which I thought would benefit me in the actuarial field, such as accounting and economics, along with a couple of computer courses to better familiarize myself with various computer programs. Through all of my course work, I managed to learn quite a bit about quite a few subjects. Portions of my education I think are more important than others. For example, I think the Actuarial Mathematics course I took will help me quite a bit more than the Humanities courses I had, but all of my courses helped to shape me into the individual I have become. One particular course I had my senior year was my Math Senior Seminar, in which we discussed various mathematical topics that mostly were not touched on in the "regular" math courses. This was one of my favorite courses, since it gave me a chance to learn about places and methods that mathematics is not normally applied. I also did two research papers for this course, which can be found following this paper. These two papers were about mathematics topics that I found interesting but were not discussed in my other courses.

During the same period I was taking all these courses, I also was involved in the exam process that is required to become a Fellow in the Society of Actuaries (FSA). The exam process was recently changed so there are eight exams to pass to become an FSA, and at the present time I have passed one. I began taking the exams my freshman year, but at that time it was mainly to see what the exams were like. Needless to say, I failed the exam that time, followed by the next two times as well. Eventually, in the fall of my junior year, I passed the first exam. Since then I have only taken the second exam once, but I plan to take it again shortly after graduation. The studying process for the exams is quite difficult, as the exams tend to cover quite a bit of material. Therefore, it takes about a couple of months to prepare for them.

An extremely valuable experience I had while I was in school was trying to obtain an internship, and, later on a full time job. The process for this was really rather simple, with the hard part being good enough for the companies to want to hire me. My freshman year, I interviewed with a few of the companies that came to campus trying to get an internship for that summer. Seeing as I had very little knowledge within the actuarial field at this point, it was very difficult for me to obtain an internship. The experience taught me quite a bit though, at least to an extent, what the companies were looking for in an intern, what they were looking for from their interns, and also, most importantly, what it is like to have an interview. It was the first time I had ever been through an actual sit-down interview like that, and that first year it was very intimidating. However, the intimidation factor has dwindled off as the process has gone on, and now I don't get nearly as intimidated by interviewers. In my sophomore year, I interviewed with more companies, but I was again unable to obtain an internship for the summer. Finally, in my junior year, after interviewing yet again with more companies, I was able to obtain an internship with Conseco, an insurance company in Indianapolis.

My internship proved to be the most informative, exciting, and by far the most memorable of all of my experiences throughout my college career. While I was at Conseco, I finally got a chance to learn just what it would be like to actually be an actuary. I was in the financial reporting area of the actuarial department, and was placed in a workgroup of five people. For the summer, I was paired up with a member of Conseco's Actuarial Student program. This basically meant that the person I was paired up with was still going through the exam process, still working to become an FSA. Throughout the summer, I was involved with numerous projects, some of which were short-term while others were long-term. I was actually put in charge of doing all of the quarter end reports for one of the smaller companies Conseco owns, while also doing the majority of the tax reserve work for another small company. Another big project that I was a part of was coding new valuation software, and then assisting in the implementation of that software. I was able to see most of the projects I was involved in carried through to completion, with the exception of a couple of the larger projects. Through the experiences of all the projects I was involved in, I discovered quite a bit about what it actually takes to be an actuary. I learned a lot about life in the business world, from handling multiple deadlines to dealing with coworkers, among many others. I believe all of that will come in very useful throughout my upcoming career as an actuary. In fact, after graduation I am going back to Conseco to begin my career as an actuary, where I can hopefully apply some of the skills I learned in my internship.

This is only a brief look at some of my experiences throughout my past four years at Ball State as an actuarial science student. My coursework, jobhunting, the exams, and my internship are merely a few of my experiences, chosen from among uncountable others. These experiences, as well as all the other ones left undiscussed, have helped mold me into the person I am today and into the person I am still becoming. Hopefully I will be able to take all of the knowledge that I have gained throughout my coursework and internship experiences and apply it to my career.

Mathematical Coding

By Tony Enk

In today's ever growing and expanding business world, communication is by far one of the most important aspects. At times, it is important for some of this communication to be written in some sort of code, or encrypted, in case an unintended party reads the message. Granted this is not the case for many college students or your everyday businessman, but it may be for those in more delicate positions that deal with confidential information, such as government agencies. This is where encryption can come into play, allowing someone to send a message in a virtually unreadable format if the receiver does not know the code being used. There are two fairly common formats that are associated with information coding, or encryption, one of which is simply writing in code, the other is defined as writing in cipher. Each of these will be discussed to some length, focusing somewhat on the mathematics behind one form of each.

First off, it is important to understand the difference between writing in code and writing in cipher. When one writes in code, they typically use numbers, a combination of numbers, or other symbols in place of merely writing words. A common example of code used by police officers, truckers, and nearly anyone communicating over radio is "10-4," which is understood to mean "affirmative." On the other hand, when one writes in cipher, they typically use letters to represent other letters, therefore everything is still written with letters. A quick example might be the "letters" would be written as "phnnhef," where "p" represents "l," "h" represents "e," "n" represents "t," "e" represents "r," and "f" represents "s." This is a very simple example of something that can actually be very complicated.

Now let's get into further detail about a common form of coding, known as Huffman coding. D.A. Huffman, a professor at the University of Santa Cruz, created Huffman coding in the 1950's. In simple terms, the idea behind Huffman coding is to use shorter bit patterns to represent more common characters, and longer bit patterns for less common characters. The procedure used finds the optimum uniquely decodable, variable length code associated with a set of events (typically letters) based on the probability each one occurs. The codes that are used to represent the symbols are typically an assigned arrangement of 0's and 1's, all based on a developed "tree" which is the key to the entire code. This key is unique to every grouping, seeing as it depends on the probabilities of all the individual elements of the information to be coded. An example of creating the tree can be found attached in Appendix A. In Appendix A, there is an example piece of information, with the coding tree and the encoded message as well.

The mathematics behind this form of coding lies mostly in the probabilities of each event within the information. It is these probabilities that nearly every other bit of coding is dependent upon in order for this code to work out effectively. These probabilities are very simple themselves, in that it is just a count of how many occurrences of each event there are out of the total number of events. For example, if there is a string of information that is fifty events in total, and thirteen of those events are all the same, then the probability of that event is 13/50. Furthermore, if this happens to be the most common event, then it will most likely have the shortest codeword assigned to it, in an effort to save space.

There are also elements of mathematics involved in creating the tree to assign the codeword to each event. There is a basic repetitive series of steps to go through to generate the tree, the first step of which is to list the events in order of decreasing probability. Then, the events with the two lowest probabilities are grouped to form a new event, and then a new list is created, again in decreasing probability. This process is repeated until two events are left, which are then combined to one event, then this serves as basically the “root” of your tree. This process is shown in the example in Appendix A.

This form of coding is very effective, in that it results in a nearly unbreakable code. The reason this code is nearly unbreakable is again based on mathematics. A message sent encoded in this manner will come out looking like a random series of 0’s and 1’s, for example 010110100101001. Anyone who may intercept this message, and does not have the key (the tree) for decoding it, has virtually no chance of cracking the code. As can be seen in that short example, as well as in the example in Appendix A, there is nearly a fifty-fifty split between the 0’s and 1’s in the message. This results in it being practically impossible to determine where one event may end and the next begins simply by looking at the string of numbers. However, with the coding tree, it is very simple to decode. This is why this form of coding is very useful, not to mention effective.

Writing in cipher is another method for encryption. As was mentioned previously, writing in cipher involves using letters to represent other letters. This can be done in a variety of ways, obviously, since any given letter could represent any of the twenty-six letters of the alphabet. Actually, there are 26! ways of choosing which letter represents which, assuming that each letter is always associated with representing the same letter for the ciphering. A very simple example of this is given in Appendix B, which contains a brief message, along with the key for decoding, as well as the encrypted message. There is not nearly as much mathematics involved here as there were with Huffman coding, but there still are some. Also, there is a very famous historical example of ciphering, and that is the Enigma machines used by the Germans in World War II.

The Enigma machines were very complex ciphering machines, which the Germans used in WWII to send their messages back and forth. From the outside, it looked basically like a standard typewriter. However, on the inside, things were much different. The innards of the machine consisted of three rotating wheels, which were selected from five available wheels. These wheels initially started in a certain position designated by the Germans, but then each time a letter key was pressed, the wheels would rotate in a seemingly random fashion. This is what made the Enigma ciphering so difficult to crack, simply because the possibilities of what letter actually represented what letter was nearly endless. In fact, in order to press the one key and get the same output twice, for example pressing the "B" key and getting a "T," it would take 16,900 keyings, when the inner mechanism returned to the same positioning. To avoid this possible recurrence, the Germans actually limited their messages to 250 letters or less.

The mathematics behind the initial setup of the machine lent to enormous capabilities in ciphering uncrackable messages. For instance, there were sixty possible wheel orders for the three wheels and 17,576 ring-settings for each wheel order. There were also other settings that needed to be made and numerous choices for them as well. In total, the number of possible key settings each day was about 159,000,000,000,000,000,000. Obviously, this gave the Germans lots of confidence in their ciphers, leading them to believe they would be uncrackable. Unfortunately for them, they were repetitive in some of the checks they used, which assisted the Allies in eventually cracking the ciphering using very advanced statistical analyses.

These two methods of encryption are only a scratch in the surface of all the possibilities for encrypting that are out there. The mathematical theory discussed is merely a scratch in that surface as well. There is much more detail involved in both the Huffman coding theory and the ciphering theory as well. However, in order to cover all of the theory in each one, not only would it take an infinite amount of time, but also an infinite amount of pages as well. Hopefully, the information presented is enough to at least pique some interest, and provide some insight into some of the theory of encryption.

APPENDIX A

Original Word:
actuarial

Encoded Word:
001100100110010111100100

The word has gone from taking up 72 bits (8 bits per letter) to taking only 24 bits.

a ⇒ 3/9
c ⇒ 1/9
i ⇒ 1/9
l ⇒ 1/9
r ⇒ 1/9
t ⇒ 1/9 ⟩ A ⇒ 2/9
u ⇒ 1/9

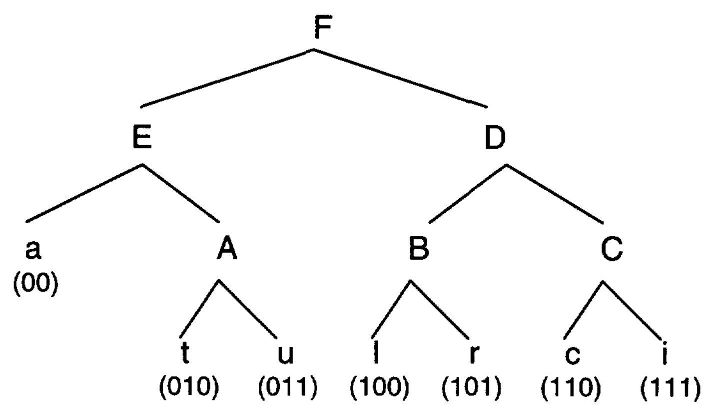
a ⇒ 3/9
A ⇒ 2/9
c ⇒ 1/9
i ⇒ 1/9
l ⇒ 1/9 ⟩ B ⇒ 2/9
r ⇒ 1/9

a ⇒ 3/9
A ⇒ 2/9
B ⇒ 2/9
c ⇒ 1/9 ⟩ C ⇒ 2/9
i ⇒ 1/9

a ⇒ 3/9
A ⇒ 2/9
B ⇒ 2/9 ⟩ D ⇒ 4/9
C ⇒ 2/9

D ⇒ 4/9
a ⇒ 3/9 ⟩ E ⇒ 5/9
A ⇒ 2/9

E ⇒ 5/9 ⟩ F ⇒ 9/9
D ⇒ 4/9



APPENDIX B

KEY	
A=Z	N=M
B=Y	O=L
C=X	P=K
D=W	Q=J
E=V	R=I
F=U	S=H
G=T	T=G
H=S	U=F
I=R	V=E
J=Q	W=D
K=P	X=C
L=O	Y=B
M=N	Z=A

Original Message:
Four score and seven years ago our fathers brought forth on this continent a new nation, conceived in liberty and dedicated to the proposition that all men are created equal.

Encoded Message:
Ulfi hxliv zmw hvevm bvzih ztl lfi uzgsvih yilftsg uligs lm gsrh xlmgrmvmg z mvd mzgrlm, xlmxvrevw rm oryvigb zmw wvwrxzgvw gl gsv kilklhrgrlm gszg zoo nvm ziv xivzgvw vjfzo.

References

"7.3.1 Huffman Coding." February 17, 1998.
http://www.ece.wpi.edu/infoeng/textbook/node94.html

Brigger, Patrick. "Huffman coding." December 14, 1995.
http://ltswww.epfl.ch/pub_files/brigger/thesis_html/node93.html

"Huffman Coding." 1997-2000.
http://www.rasip.fer.hr/research/compress/algorithms/fund/huffman/

"Huffman Compression." http://www.eee.bham.ac.uk/WoolleySI/All7/huff_1.htm

Schindler, Michael. "Practical Huffman Coding." October, 1998.
http://www.compressconsult.com/huffman/

Stripp, Alan. "How the Enigma Works." November 2000.
http://www.pbs.org/wgbh/nova/decoding/enigma.html

Conway's Theories

By Tony Enk

There are many interesting mathematical tricks and oddities that have been discovered throughout history. A few of these tricks or oddities have been discovered and calculated by the same man, that man being John Horton Conway. He has published some of his theories and tricks in the past, a few of which will be discussed in some length in the following pages. The theories (or in some cases formulas) which will be focused on are as follows: how to calculate the day of the week given the date, how to calculate the date of Easter, and how to calculate the phase of the moon.

The first of these theories is what has come to be known as the Doomsday Algorithm. This algorithm allows you to calculate, given a specific date, what day of the week that date was or will be. First of all, we will start of with a chunk of information that is to be used as a reference for this calculation. Within any given year, the following days, which are called doomsdays, all fall on the same day of the week: 4/4, 6/6, 8/8, 10/10, 12/12, 5/9, 9/5, 7/11, 11/7 and the last day of February. Another piece of information that it is important to know is that the doomsday for any century follows a certain pattern as well, which goes as follows:

In years:	Doomsday is:
15xx, 19xx, 23xx, etc.	Wednesday
16xx, 20xx, 24xx, etc.	Tuesday
17xx, 21xx, 25xx, etc.	Sunday
18xx, 22xx, 26xx, etc.	Friday

Knowing these two pieces of information, you can now begin to calculate doomsday for a desired year. This is the most difficult part of the algorithm, due to the fact that it requires a slight bit of actual mathematical calculation. If the year in question is 19xx, divide the xx part by 12 to get a quotient, which we will label as q, and a remainder, which we will label as r. Next, divide the remainder r by 4, getting another quotient s. Any remainder left over at this point can be disregarded, as it is not used. Add q, r, and s together, then count that many days starting from that century's particular doomsday, which is Wednesday for the 1900's, as stated in the table above. From this point, it is very simple to find what day of the week any given day may be within a particular year. It is merely adding or subtracting days from any of the previously given doomsdays within the year. The only hassle that comes in is if the year in question is a leap year and you are trying to figure out a day of the week that occurred before the end of February. The only other word of warning would be that this only works for the Gregorian calendar, and is said to only work up through 3999.

As an example, if we wanted to calculate what day of the week June 30, 1979 (a Saturday, according to the calendar) was, we would take the following steps. First of all, doomsday for the 1900's is Wednesday, as was said before. For 1979, we divide 79 by 12, getting q=6 and r=7. Then, we divide r=7 by 4, giving us s=1. Finally, when we add the three together, we get 6+7+1=14, telling us that doomsday for 1979 is Wednesday. Therefore, we go back to 6/6, which was Wednesday, and count 24 days, which since we are dealing in days of the week is the same as counting 3 days, therefore June 30, 1979 was a Saturday. As stated before, June 30, 1979 was in fact a Saturday.

The second theory or formula that Conway came up with was how to calculate the date of Easter. Commonly, Easter's date is fairly difficult to calculate, as it is said to fall on the first Sunday after the full moon that occurs next after the vernal equinox. However, with Conway's method, it is merely a calculation of a few numbers. The first number, often referred to as the golden number, is g = Year(mod 19) + 1. The second number that must be calculated is the century term, or c. c is calculated (again, in Gregorian years), using the year Hxx (where H is the century) using the formula c = -H + [H/4] + [8&(H+11)/25] (use the integer part for the brackets [ ]). Next, you must find the date of the Paschal Full Moon, the first full moon after the vernal equinox. The formula for this is (March 50=April 19) – (11&g+c)(mod 30). The first part of that formula, (March 50=April 19), requires a bit of explanation. This means if the number in the second part of the formula is smaller than 19, use April 19 as the start date, but otherwise use March 50 as the start date. The Paschal Full Moon then falls on the calculated date, and Easter is the following Sunday. At this point in the process, the Doomsday Algorithm must be used to determine the actual day the Paschal Full Moon takes place, and then added correspondingly to reach the following Sunday. There are two exceptions to the Paschal Full Moon calculation: if the formula returns April 19, then the Paschal Full Moon actually falls on April 18, and if the formula returns April 18 and g (the golden number) is 12 or greater, the Paschal Full Moon is April 17.

As an example, let us verify that Easter actually was supposed to be April 15 in 2001. First of all, the golden number is calculated to be g=7. Secondly, the century term can be calculated to be c=–6. Now for the Paschal Full Moon, which with a quick calculation is shown to be

April 19 – (71-6)(mod 30) = April 19 – 11 = April 8. Using the Doomsday Algorithm, it can be shown that April 8 was on a Sunday, which means that Easter was in fact on the following Sunday, April 15.

The third, and most likely the simplest, of Conways' theories discussed here is the one that deals with calculating the phase of the moon. In the 20^{th} century, calculate the remainder of the last two digits of the year divided by 19. If this number is greater than 9, subtract 19 so that you get a number between –9 and 9. Take the remainder that you got and multiply it by 11, then reduce it (mod 30). This gives you a number between –29 and 29. At this point, add the number of the day and the month (use 3 and 4 respectively for January and February instead of 1 and 2). Now subtract 4 and reduce (mod 30) to get a number between 0 and 29, which represents the age of the moon. As a modification, in the 21^{st} century use –8.3 days instead of –4 for the last part of the calculation. Now let's see if this calculation works by checking to see if there really was a full moon on April 8, 2001 (which we just calculated as the Paschal Full Moon above). The remainder after the first calculation of 01/19 would be 1, and when multiplying that by 11 we would end up with 11. There is no need to reduce this (mod 30) as it would also be 11 (mod 30). Now we add the day and month, so we add 4 and 8, giving us 23. In the final calculation we subtract 8.3, leaving us with 14.7, which is basically halfway through the life of the moon, or full. Therefore, the calculations worked, since on April 8, 2001 there actually was a full moon.

These three theories, more calendar tricks really, were very intriguing to me, and hopefully they were to you as well. It is pretty interesting how one can calculate the day of the week that pretty much any given date falls on, or to be able to calculate what date Easter is going to be, and finally approximately what phase the moon will be in on a specific date. There are many other interesting theories out there that can be discovered if you look hard enough. These just happened to be a few that I found the most interesting. Granted they are not entirely the most useful bits of mathematics, but they are intriguing nonetheless.

References

Berlekamp, Elwyn R., et al. Winning Ways, For Your Mathematical Plays. New York: Academic Press, 1982.

"How do I calculate the phase of the moon?" April 21, 2001. http://www.faqs.org/faqs/astronomy/faq/Part3/section-15.html

Jefferys, William H. "What is the day of the week?" 1998. http://quasar.as.utexas.edu/BillInfo/doomsday.html

Jefferys, William H. "Date of Easter, Rosh Hashanah, and Passover." 2000. http://quasar.as.utexas.edu/BillInfo/ReligiousCalendars.html

"[sci.astro] Time (Astronomy Frequently Asked Questions) (3/8)." http://www.uni-konstanz.de/misc/faqs/sci/msg00014.html