SAINT MARY'S COLLEGE OF CALIFORNIA

MATHEMATICS SENIOR SEMINAR

SENIOR ESSAY

Breaking the Enigma Code

Erica Musgrave

Faculty Advisor: Dr. Christopher Jones

May 16, 2016

1 Introduction

Until the end of World War I, almost all commmunication between military personnel was through transferring messages by hand or by telegram. These methods of commmunication were often challenging and slow, but the interception of the enemy's messages was rare. As a result, people did not see the need for a high level encryption. After World War I, the radio became the primary method of contact which meant communication became nearly instantaneous, and it was much easier to communicate with any military station whether it was on land or at sea. However, communicating through radio also had a major consequence. It was now very easy for the enemy to intercept messages, so people soon realized the codes and encipherments that were previously used would no longer suffice. As a result, many cryptographers began to develop more advanced encryption systems [2].

Arthur Scherbius, a German inventor, was one of these cryptographers who had a desire to replace the previously inadequate cryptographic systems. He developed an electrical cipher machine called the Enigma which was not only effective but also extremely efficient and easy to use. In 1923 he started to produce and advertise his Enigma machine which resulted in the German government becoming interested in his design. Then in 1926, the German military decided to buy a large number of machines and use them as their main source of encipherment, and the Enigma quickly became known as the most fearsome encryption device in history.

In the 1930s, the Germans started to rearm themselves which made the Polish extremely nervous. As a result, the Polish increased their monitoring of German communications. Since the Germans had improved their encryption system, the Polish decided they needed to focus on recruiting cryptologists. Among these cryptologists was Marian Rejewski who had received his Ph.D in mathematics from Poznań University in 1929. Rejewski used the theory of permutations to make a major breakthrough in the analysis of the Engima machine which led to the Polish being able to eventually find a method to decode German messages [3].

Figure 1: Arthur Scherbius

Figure 2: Marian Rejewski

In this paper, we will first introduce some necessary background about ciphers and permutations. Then we will discuss the structure of the Enigma machine and how it is used. Finally, we will discuss how the Polish were able to eventually break the Engima code, and we will go over one method of decoding messages enciphered with the Engima machine.

2 BACKGROUND

Cryptography is the process or skill of communicating through encoding and decoding secret messages. This field of study is highly valued because it allows for the communication of confidential information securely between two parties. Therefore, there has always been a great need for secure cryptosystems, which are methods of allowing a sender to produce a seemingly nonsensical message that the receiver can then decipher and read by using a specific key or piece of information.

At first, these encoding systems were very basic. They were mostly ciphers that involved the shifting or substituting of letters in a message by a particular set key. One of the earliest examples of a cipher was created by Julius Caesar during the Roman times, and it is called the shift or additive cipher. This consists of shifting each letter of the plaintext message over a certain number of letters to produce a ciphertext letter. For example if the key is 5, then each letter would be shifted by 5 letters. An a would become an F, and b would become G, and etc. Note that lowercase letters are usually used for plaintext and capital letters are used for ciphertext. Even though this is a very simple system to use, it is not very secure because there are only 26 different possible keys. This means an individual could easily decode a message by using a brute force attack of checking each possible key.

Another more complicated example is a substitution cipher. A substitution cipher replaces each letter of the plaintext message with another letter. The key of a substitution cipher is the correspondence between each plantext letter and ciphertext letter. For example, the following is a key of a substitution cipher:

Plaintext:	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*	*l*	*m*	*n*	*o*	*p*	*q*	*r*	*s*	*t*	*u*	*v*	*w*	*x*	*y*	*z*
Ciphertext:	*E*	*K*	*M*	*F*	*L*	*G*	*D*	*Q*	*V*	*Z*	*N*	*T*	*O*	*W*	*Y*	*H*	*X*	*U*	*S*	*P*	*A*	*I*	*B*	*R*	*C*	*J*.

The key can also be described as a permutation. A permutation of a set X is a bijection from X to itself. In other words a permutation can be thought of as a rearrangement of the items of X in a given order. In a substitution cipher, the set is all 26 letters and the key is a permutation of the 26 letters. A more useful way to represent permutations is with cyclic notation where we write the permutation as a product of what are called disjoint cycles. Note in our key we can see

$$b \to k \to n \to w \to b$$

so we have completed a cycle since we started and finished with the same letter. This cycle can be represented as $(bknw)$. We can follow the same process to find all of the other cycles created by our permutation, then we can write our permutation as the following product of cycles

$$(aeltphqxru)(bknw)(cmoy)(dfg)(iv)(jz)(s).$$

Note that all of our cycles are disjoint because a permutation is a bijection which means that two plaintext letters cannot go to the same ciphertext letter.

Substitution ciphers are more secure than Casesar ciphers because there are

$$26! = 26 \times 25 \times 24 \times \cdots \times 3 \times 2 \times 1 = 403291461126605635584000000$$

different possible keys for a simple substitution cipher. This means it is very difficult to do a brute force attack because there are too many possible keys. However, there is still a method in which these ciphers can be broken.

Cryptanalysts break substitution ciphers by taking advantage of the fact that all languages have rules which create patterns. For example in the English language, we know the letter e is the most common letter. Therefore, we would expect the ciphertext letter that appears the most frequently to correspond to the letter e. It is possible we are incorrect, but it is more probable the ciphertext letters that occur the most frequently correspond to the letters $e, t, a, o, i, n,$ or, s. Cryptanalysts can also use other patterns, like in the English language we know it is very common for the letter t and h to appear next to each other. By taking into account the different word patterns and the letter frequencies corresponding to the language being used, substitution ciphers are generally very simple to break. This type of attack is called frequency analysis.

A way to make this type of cryptosystem more secure is to use a collection of substitution ciphers or permutations to encode one message. For example, one permutation would encode the first letter, then another permutation would encode the second letter, and so on. Since there are 26! different possible permutations, there are more than enough permutations to encode every letter of a message with a different permutation. This means a cryptanalyst would no longer be able to use frequency analysis, so it would be very difficult to break this type of cryptosystem.

However, this type of system is also very difficult to implement by hand. The sender has to keep track of the order of permutations they use while encoding and then communicate that order to the receiver so they may decode the message. Therefore, cryptographers developed machines to generate and use permutations to encode a message. The Enigma machine was one of the first machines to implement this cryptosystem in an effective and efficent way [1].

3 Enigma Machine

The Enigma machine is composed of three basic parts. There is a keyboard in order for the operator to input letters, a scrambler which encrypts the input letter, and a lampboard where the output letter lights up. In order to encrypt a letter with the machine, the key corresponding to the appropriate letter is pressed on the keyboard which then sends an electric pulse, generated by a battery, through the machine. This pulse is sent through the scrambler, and it ends at the lampboard where the corresponding encrypted letter lights up.

Figure 3: Example of an Enigma machine

The scrambler is the most complicated and important part of the machine, and its main components are the rotors. The rotors are thick discs with wires that run from a point on the left side to a point on the right side. Each entry point and exit point of the rotors represent a single letter, so there are 26 entry points and 26 exit points. As a result, each rotor implements a basic substitution cipher.

For example, suppose we wanted to encrypt the message *bed* with the following simplified Enigma machine that is limited to a six letter alphabet.

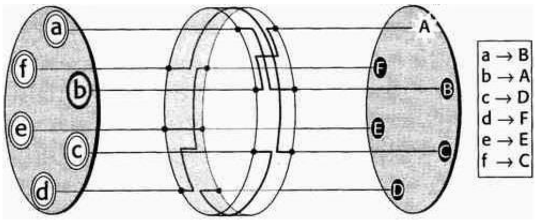

Figure 4: Simplified version of a Enigma machine with one rotor

The disk on the left is the keyboard, the middle ring is the rotor, and the disk on the right is the lampboard. The wiring of the rotor determines how the plaintext letters will be encrypted. As a result, we can see that when the letter a is typed into the keyboard, then the letter B is illuminated on the lampboard. This means that the letter a is encrypted to the letter B. We can follow the same process with each letter, and what results is the substitution cipher described in the chart on the right of Figure 4. Therefore, our message *bed* would be encrypted as AEF.

Since there are known cryptanlaysis methods that can break a basic substitution cipher as mentioned earlier, Arthur Scherbius wanted to find a way to implement a multiple subsitiution cipher system. He achieved this by designing the rotors to rotate each time an electric pulse is sent through the machine. As a result, the way the letters are scrambled changes after each letter. In order to demonstrate how to encrypt with a rotating rotor let us consider the following example.

Suppose we have one rotor with an alphabet of six letters. Then we could represent our machine with the following diagram which is a two-dimensional version of Figure 4.

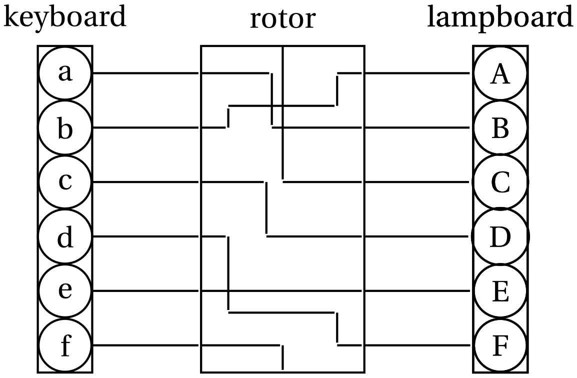

Figure 5: Simplified version of Enigma machine with only one rotor

Now suppose we still wanted to encipher the message *bed*. First we would press the letter *b* on the keyboard. Then we can see from Figure 5 that the letter *A* would light up on the lampboard. Therefore, *b* is enciphered to *A*. Then since we have enciphered a letter, the rotor will shift, or rotate, one position. This means that all of the wires will maintain the same pattern but they will shift down by one letter position. For example, instead of *e* being enciphered to *E*, now *f* enciphers to *F*. This is demonstrated in Figure 6.

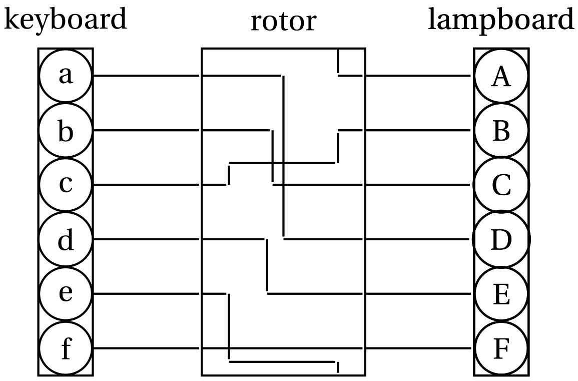

Figure 6: Simplified Enigma machine in the second postion

Next we would press the letter *e* on the keyboard and then we can see from Figure 6 the letter *A* would light up on the lampboard. This means that so far our cipher text is *AA*. Also, since we have enciphered a letter, the rotor will shift another position.

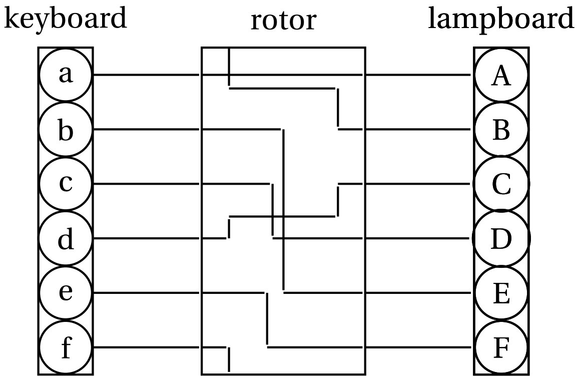

Figure 7: Simplified Enigma machine in the third postion

Finally we press the letter *d* on the keyboard, and then we can see from Figure 7 that the letter *C* will light up on the lampboard. Therefore, our final cipher text is *AAC*.

The rotation of the rotors is a very important feature of the Enigma machine because it makes the cipher more complex. However, if the machine has only one rotor, there is still an obvious flaw. After 26 letters have been inputed, the rotor will be back to its original position. This means there is a possibility of repetition which can then produce regularity and structure in the cipher text, so the encryption is easier to break. This is why a traditional Enigma machine has three rotors that are used to encipher a letter as shown in Figure 8. Since the position of each rotor determines how a letter will be enciphered, the operator needs a way to easily recognize what position each rotor is in. Since a rotor has 26 different positions it could potentially be in, each position is given a letter in the alphabet. For example, the initial conditions of the rotors in an Enigma machine could be VLE.

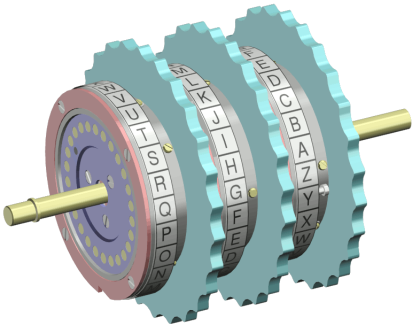

Figure 8: Example three rotors with initial position VLE

Now if every rotor rotated every time a letter is inputed, we would still have the same problem as before. Therefore, Scherbius designed the rotors to rotate in the following way. The first rotor, also known as the fast rotor, shifts each time a letter is inputed into the machine. The second rotor, or middle rotor, rotates only after the fast rotor has made a full revolution. In order for the machine to keep track of when the fast rotor completes a revolution, the fast rotor has a notch cut into it. This notch will connect with a lever and force the middle rotor to rotate whenever the fast rotor completes a revolution. The letter that corresponds to the notch is called the ring setting, so there are 26 different possible ring settings for the fast rotor. Similarly the third rotor, or slow rotor, does not rotate until the middle rotor has made a full revolution. For example, suppose the ring setting of the fast rotor is at position A. Then if the rotors are in position ZAD, the next time a letter is inputed into the machine the rotors will be in position ABD. This means that instead of it taking only 26 rotations to return to the original position, it now takes $26 \times 26 \times 26 = 17576$ rotations for the machine to return to its original position which makes the cryptosystem much more secure.

In addition to the rotors, another key feature of the scrambling portion of the machine is the reflector. The reflector is very much like a rotor because it is a disc and it rearranges the letters. However, it is static which means it does not rotate. Also, instead of having the wires go from one side of the disc to the other, it goes from a letter on one side to another letter on the same side. This is demonstrated in Figure 9.

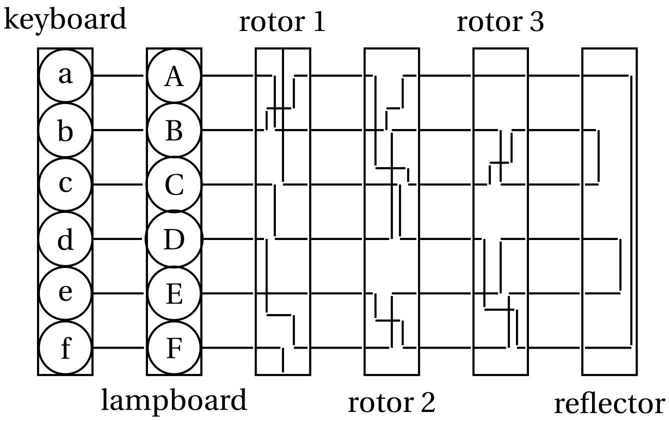

Figure 9: Two dimensional Enigma machine. Note this figure makes it seem like the electric pulse is sent from the keyboard through the lampboard and then to the rotors. However, in reality the paths from the lampboard and the keyboard are separated.

As one can see in Figure 9, when the operator types a letter on the keyboard, a signal travels through the three rotors. Then the reflector receives the signal and sends it back through the same three rotors and to the lampboard.

Even though it may seem as if the reflector does not add anything to the design of the machine because it does not add a lot of complexity to the encryption process, there are some major benefits. The main benefit is that it makes the deciphering of messages a much simpler process. The reflector makes it so that if the letter a is enciphered to the letter P while the machine is in position AXD, then the letter p will encipher to A in position AXD. This is because the reflector connects pairs of letters. For example in Figure 9 we can see that a enciphers to F and f enciphers to A. Thus, a and f are a pair. Similarly, b and c are a pair, and d and e are a pair. Therefore, the reflector can be written as the permutation $(af)(bc)(de)$ which is a product of 2-cycles or transpositions. This makes deciphering a message much easier because typing in the cipher text into the machine will decipher it as long as the machine starts in the same initial position as when the message was enciphered. Because the process of encoding and decoding is very simple, an operator does not necessarily need any special skills to operate the machine. This made the machine even more desirable and one of the main reasons why the Germans chose to use the Enigma machine over other cipher machines.

As mentioned previously, the reflector does not increase the complexity of the enciphering process, so it does not increase the security of the machine. If someone was trying to decipher a message without knowing the intial settings of the rotors, then they would have to try all the 17567 possible settings to see if it would decipher their message. For one person, this seems like a daunting task, especially back when computers were not invented yet. However, if the work was dividing amongst a large group of people, this task was actually achievable.

Since Scherbius was designing the Engima machine to be used by the German military, he needed to find a way to increase the security of the machine, so the Germans would be willing to invest in his invention. There are numerous ways he could have improved the security. One option was to increase the number of rotors because for each rotor added, the number of possible settings would be increased by a factor of 26. However, more rotors would also increase the size of the machine which would not be optimal. Instead, Scherbius decided to make the rotors removable and interchangeable which increased the number of possible positions by a factor of six. Each rotor was given a number, so the initial settings would not only contain the initial positions of the rotors but also what order the rotors were in. For example, an initial setting could be 213 and AXB. This means the fast rotor is rotor 2, and it is in position A. The middle rotor is rotor 1, and it is in position X. Finally, the slow rotor is rotor 3, and it is in position B.

In addition, Sherbius added another feature to the machine called a plugboard. This feature was added between the keyboard and the rotors, and it allowed for the operator to insert wires that would swap letters. For example, a wire could be inserted into the plugboard that connected the letter A and C. This means that when the letter A is inputed, then it follows the path that originally corresponded to the letter C.

Each operator was supplied with six wires that could be inserted into the plugboard, and the positioning of the wires were given in the initial settings along with the rotors' order and positions. This meant that up to six different pairs of letters could be swapped, and the number of ways that up to 6 pairs of letters can be swapped is

$$\frac{{}_{26}P_{12}}{2^6 \times 6!}$$

which is equal to

$$\frac{(26\times 25)\times(24\times 23)\times(22\times 21)\times(20\times 19)\times(18\times 17)\times(16\times 15)}{2^6\times 6!} = 100,391,791,500.$$

The security of the Engima machine depends upon its total number of possible initial settings or keys. We know the total number of keys is equal to the number of possible rotor settings × the number of ways to order the rotors × the number of ring settings of the fast rotor and the middle rotor × the number of possible plugboard settings. Therefore, we can find the total number of possible intial settings using the information in the following table.

Rotor Settings	$26 \times 26 \times 26 = 17,567$
Rotor Orderings	$3! = 6$
Ring Settings	$26 \times 26 = 676$
Plugboard Settings	100,391,791,500
Total	7,156,755,732,750,634,000

Also, if the rotor wirings are not known then this would add another factor of 26! since there are 26! different permutations a rotor could implement. This would result in a work factor of more than 10^{100}. When the Polish were analyzing the machine, they knew the rotor wirings because they bought secret Enigma information from the German, Hans-Thilo Schmidt, who had a job at the Chiffrierstelle, the Enigma command center [3]. However, even with knowing the rotor wirings, the Engima still has a work factor of around 10^{20}. This means the initial settings can not be checked by brute force. In addition, the plugboard and the rotation of the rotors protect the machine from frequency analysis. Therefore, the Enigma machine successfully implements a cryptosystem that is very difficult to break but also very easy to use.

4 Decoding Messages

Since the Enigma machine could not be attacked by brute force, the cryptologists had to find a different way to attack it. It was actually the way the machine was operated that eventually led to the security of the machine to become compromised. As mentioned previously, in order to decode a message that was encrypted using the Enigma machine, we would need to know what the initial settings of the machine were when the message was encoded. Therefore, the German operators needed a way to let the recipent of the message know what initial settings they used when encoding the message. One option would be to have every operator use the same daily setting, but this would mean a lot of messages would be encoded using the same settings. Every message would have its first letter encoded with the same permutation, its second letter encoded with the same permutation, and so on. If a cryptologist intercepts say 100 messages all encoded with the same daily setting, then they could strip off the first letter of every message, so they would have 100 letters all encoded with the same subsitution cipher. The cryptologist could then apply a modified frequency analysis. Note that the analysis must be modified because we have to take into account the location of the letter. For example, the letter e is a very common letter in the English language but it is not a very common letter to begin a word with. A cryptologist could break each permutation separately which would then allow them to break the entire code. Therefore, every operator using the same daily setting would not be a cryptorgraphically secure option.

Instead, the Germans decided to have every operator chose their own initial settings. Then in order to communicate their initial settings with their intended recipient, they would use a universal daily initial setting to encode their personal initial settings at the beginning of the message. Then they would change the settings to their personal initial settings to encode the rest of the message. For example, suppose the universal setting is CHT and the operator has chosen the setting KNB. The operator would set the machine to setting CHT as shown in Figure 10 and first encode $knbknb$.

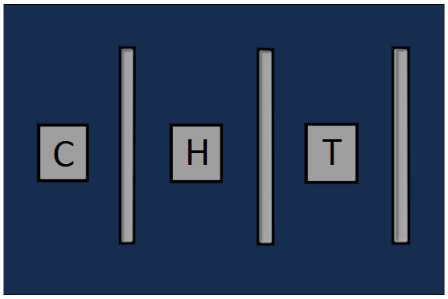

Figure 10: Example of rotor display on the machine. The silver strips are the rotors and the letters represent the current position of the three rotors.

Suppose this results in $SUIJQX$, then that would be the beginning of the message. Then the operator would change the machine setting to KNB and encode the actual message. In order to decode the message, the recipient would set the machine to the setting CHT and decode the first six letters of the message. This would result in $KNBKNB$, so the recipient would know to set the machine to the setting KNB to decode the rest of the message. The reason they chose to encode the personal settings twice at the beginnning was just in case the operator accidently hit the wrong letter when they were encoding their personal settings or there was a scrambled signal. If the recipient notices the first three letters do not match with the second three letters after they have been decoded, then they know to tell the sender to resend the message.

Even though this seems like a much more secure option than having a fixed universal daily setting, the enciphering of the same three letters twice is what eventually helped cryptanalysts find a method for deciphering German messages. This is because enciphering the same three letters twice told the cryptanlaysts that there was a connection between the first and fourth letters, the second and fifth letters, and the third and sixth letters because they are both produced by the same letter. Even though this information would not be useful if you only have one message, cryptanlaysts had access to a lot of messages sent each day that all had their first six letters encoded with the same universal settings. Therefore, they could use this information to help them decode the messages and find the universal initial settings.

4.1 Method of Exploitation

Consider an Enigma machine with three rotors, R_1, R_2, and R_3, and a reflector U. Suppose we are trying to decipher messages from the German military encoded using the method described above. We need to find the universal initial settings that were used to encode each of the messages. In order to do this we need to decipher the first six letters of each message. Now during the encipherment of these letters we know that the fast rotor, R_1, moved 6 positions. Now it is also possible that after one of the times R_1 moves it finishes a revolution which causes the middle rotor, R_2, to move. In addition there is a possibility that this will cause R_2 to finish a revolution which causes the slow rotor, R_3, to move. These cases would cause the decipherment of the messages to be a lot more complicated and they would require a more difficult solution. Therefore, we are only going to consider the case where R_2 and R_3 remain stationary which occurs 21/26 of the time. Also, if we only look at the first and fourth letters, R_2 and R_3 remain stationary 23/26 of the time.

Now when considering the encipherment of a single letter, we can think of each of the rotors and the reflector as a permutation. Then the encipherment process could be represented as the following product of permutations:

$$(R_1)(R_2)(R_3)(U)(R_3)^{-1}(R_2)^{-1}(R_1)^{-1}$$

where R_n represents the permutation corresponding to the rotor R_n and $(R_n)^{-1}$ represents the inverse of the permutation corresponding to the rotor R_n. For example, suppose we have the permutation

$$(aeltphqxru)(bknw)(cmoy)(dfg)(iv)(jz)(s)$$

then the inverse of this permutation would be

$$(urxqhptlea)(wnkb)(yomc)(gfd)(vi)(zj)(s)$$

because the product of these two permutations is the identity permutation which fixes every letter.

Now since we are assuming R_2 and R_3 remain stationary during the enciphering of the first six letters, we can represent $(R_2)(R_3)(U)(R_3)^{-1}(R_2)^{-1}$ as one permutation, S. Therefore our enciphering process for a single letter is

$$(R_1)(S)(R_1)^{-1}.$$

Also it is important to notice that S is a reflector type of substitution which means that S can be represented as a permutation where no letter is sent to itself.

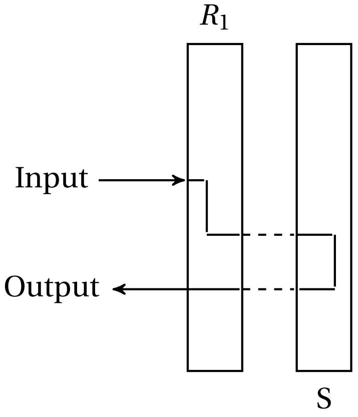

Figure 11: Enigma Machine when R_2 and R_3 are stationary.

For example, let us consider our simple Enigma machine from Figure 9. Since R_2 and R_3 are stationary, we can reduce R_2, R_3, and the reflector to the reflector permutation S.

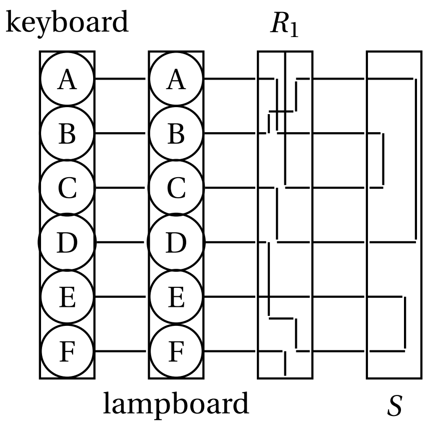

Figure 12: Simple Enigma machine with rotor 2, rotor 3, and reflector all represented as the reflector S.

Now suppose that Figure 12 is the Enigma machine that is used to encode a group of messages. At the beginning of each message we know that the same three letters are encrypted twice. Therefore the first letter and the fourth letter are encryptions of the same letter. Now in order to further analyze this information we can create an encipherment matrix of our Enigma machine.

A	B	C	A	D
B	A	B	B	C
C	D	F	C	B
D	F	E	D	A
E	E	D	E	F
F	C	A	F	E
Input	R_1		S	

Figure 13: Encipherment matrix of Enigma machine in Figure 12.

In this matrix the first column represents all the possible input letters. The second column represents what letter R_1 would encipher the input to while in position 1. The third column represents what letter R_1 would encipher the input to while in position 4. Finally the last two columns represent the transpositions or pairs of reflector S.

We can use this matrix to trace the encipherment of an input letter. For example, while R_1 is in position 1, the encipherment of A would be

$$\begin{array}{ccccccc} A & \rightarrow & B & \rightarrow & C & \rightarrow & F \\ & & R_1 & & S & & R_1^{-1} \end{array}$$

which we can see is correct from Figure 12. If R_1 is in position 4, the encipherment of A would be

$$\begin{array}{ccccccc} A & \rightarrow & C & \rightarrow & B & \rightarrow & B \\ & & R_1 & & S & & R_1^{-1} \end{array}$$

which we can see is also correct if we first rotate the rotor in Figure 12 four times and then encipher A. Now we call the letters FB an encipherment pair because we know they are both enciphered from the same letter, A. This implies that if A was the first letter in the personal initial settings that are being encoded in the first six letters of the message, the first letter of the ciphertext would be F and the fourth letter would be B.

We can then follow a similar process and find that the six encipherment pairs are FB, CA, BD, EC, DF, and AE. Then we can use these pairs to generate cycles. For example,

$$F \rightarrow B \rightarrow D \rightarrow F$$

so (FBD) is a cycle. Similarly we can conclude (CAE) is a cycle. Notice these cycles are the same length, and we can line these cycles up so that the letters in the same column are enciphered to each other while R_1 is in position 1, and the letters that are diagonals from each other are enciphered to each other while R_1 is in position 4. The following is the correct alignment of our two cycles:

$$\begin{array}{ccc} F & B & D \\ A & C & E \end{array}$$

This cycle structure was the key discovery that helped cryptanlaysts to find a way to decode messages encrypted by the Engima machine. Marian Rejewski, the Polish mathematician who helped develop this method of attack on the Enigma machine, was the first to discover the following fact. For any Enigma machine, the cycles that are generated by the encipherment pairs will always come in pairs, and the pairs will share the relationship described above. This relationship between cycles will be further discussed in more general terms in the following section.

4.2 Relationship between Cycles

Recall we are assuming that R_2 and R_3 remain stationary during the encipherment of the first six letters. Let X be a letter in our alphabet, and let $F(X)$ be the encipherment of X by R_1 while in position 1, and let $S(X)$ be the encipherment of X by R_1 while in position 4. Note that we know $F(X)$ and $S(X)$ but we do not know X.

Now suppose we have collected a large number of messages and then we choose one of these messages. Let $F(X_1) = Y_1$ be the first letter of the message, and let $S(X_1) = Y_2$ be the fourth letter of the message. Then we can search through our other messages until we find one that begins with the letter Y_2. Then we let $F(X_2) = Y_2$ and $S(X_2) = Y_3$. We can then continue this process until we have completed a chain of letters. Note that since our alphabet contains a finite number of letters, the chain must eventually close. Let n be the length of our chain, then $F(X_n) = Y_n$ and $S(X_n) = Y_1$. Thus we have the following closed n-cycle:

$$Y_1 Y_2 Y_3 \dots Y_n.$$

Now we can choose another message which begins with a letter that does not occur in the cycle and follow the same process to produce another cycle. We can continue creating cycles until all of the letters in the alphabet have been used. Since every letter in the alphabet must occur in a cycle, one of the cycles must contain the letter X_1. Suppose X_1 is the start of a new cycle. Then since $F(X_1) = Y_1$ we know that $F(Y_1) = X_1$ since the Enigma machine has the reflective property. Also we know $S(Y_1) = X_n$ since $S(X_n) = Y_1$ for the same reason.

Now since we know that $F(X_n) = Y_n$ and $S(X_{n-1}) = Y_n$, we know that $F(Y_n) = X_n$ and $S(Y_n) = X_{n-1}$. Thus we can continue to extend our chain in the following way until we end with $F(Y_2) = X_2$ and $S(Y_2) = X_1$, so we end with the following n-cycle:

$$X_1 X_n X_{n-1} \dots X_2.$$

Therefore, we have shown that for each cycle that we produce, we can produce another cycle of the same length. This implies that the cycles occur in pairs, and it also shows that the pairs share a particular relationship. If we take the two cycles, one written forward and the other backward, then we can align them in such a way that the letters in the same column encipher to each other when R_1 is in positon 1, and the letters that are diagonal encipher to each other when R_1 is in position 4:

$$\begin{array}{ccccc} Y_1 & Y_2 & Y_3 & \dots & Y_n \\ X_1 & X_2 & X_3 & \dots & X_n. \end{array}$$

This means that after we have found all of the cycles, we can align all of them as shown above. We can then decipher the first six letters of any message which allows for us to find the initial settings that can be used to decipher the rest of the message.

4.3 Example of finding the setting of the fast rotor

Suppose we have a large number of messages, and we want to figure out the universal daily setting of the fast rotor. Recall that we are assuming that the middle rotor and the slow rotor do not move, so we can consider the middle rotor, the slow rotor, and the reflector as just one reflector. Since the Polish knew the rotor wirings, we are also assuming that we know the rotor wirings for the three possible rotors: Rotor 1, Rotor 2, and Rotor 3. In addition, in this example we are assuming we know the fast rotor is Rotor 3. In reality we would have to go through each possible rotor to check if it is the fast rotor, but in order to eliminate having to check hundreds of incorrect rotor positions in this example, we can just assume we know the fast rotor is Rotor 3.

In all of our collected messages we know that the first six letters are encoded using the same universal initial settings. We also know that the first and the fourth letters of each message correspond to the same entry letter. Therefore, we can collect all of the first and fourth letters of each message to find all the possible encipherment pairs which are shown below in alphabetical order:

$$\begin{array}{ccccc} AM & GY & ML & SI & YN \\ BG & HP & NU & TT & ZO \\ CK & IZ & OX & UV & \\ DH & JE & PJ & VW & \\ ED & KA & QF & WB & \\ FC & LQ & RR & XS & \end{array}$$

Now we want to find the cycles generated by these pairs as explained in Section 5.1. Recall that we know these cycles must occur in pairs and the pairs must have equal length. Let us start with the first pair AM. Then we chain it with the pair ML so we now have the chain AML. We then continue this process until we have cycled back to

A. This results in the cycle $(AMLQFCK)$. We then continue this until each letter is in one of the cycles. We will then find that we have the following cycles:

$$\begin{array}{ccccccc}
A & M & L & Q & F & C & K \\
B & G & Y & N & U & V & W \\
D & H & P & J & E & & \\
I & Z & O & X & S & & \\
R & & & & & & \\
T & & & & & &
\end{array}$$

Now in order to find the initial setting of the fast rotor, we have to align all of these cycles correctly with one of each pair of cycles written in reverse order. Recall that when we have the correct alignment, the letters in the same column encipher to each other when the fast rotor is in the first position and the diagonal pairs encipher to each other when the fast rotor is in the fourth position. Fortunately in this case we have two one cycles which have only one possible alignment:

$$\begin{array}{c}
R \\
T
\end{array}$$

This implies that R enciphers to T when the rotor is in the first and fourth position.

Now let us consider the encipher table for Rotor 3. An encipher table is a 26×26 table that lays out the substitutions corresponding to the rotor in each of its possible positions. For this example, we are only going to include the first 6 positions of our encipher table for Rotor 3.

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

Figure 14: Encipher table of Rotor III. Note the left column corresponds to the setting or position the rotor is in, the top row corresponds to the input letters, and the body of the table contains the output letters that result when the input letter is entered into the rotor while it is in the particular setting.

Suppose the initial setting of Rotor 3 is setting A. Then we can see Rotor 3 enciphers R to W and enciphers T to A. Since we know the entire Enigma machine enciphers R to T we know that the reflector $S = R_2R_3U(R_3)^{-1}(R_2)^{-2}$ must have a connection between W and A as seen in the following figure:

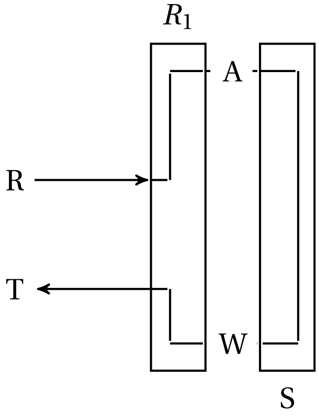

Figure 15: Enigma machine with fast rotor in setting A

Now when the fast rotor is in the fourth position which corresponds to setting D, we can see the rotor deciphers W to J and deciphers A to Y. Since the reflector S does not change when the fast rotor rotates, there is still a connection between A and W. Thus we can assume that when the fast rotor is in the fourth position, J and Y encipher to each other.

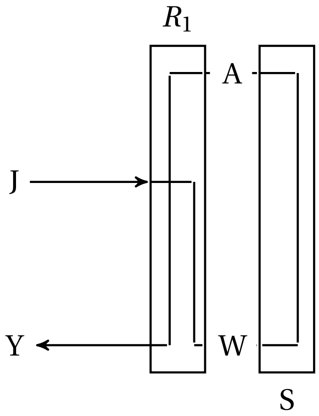

Figure 16: Enigma machine with fast rotor in setting D

We can then conclude that J and Y must be diagonal pairs in a pair of cycles. However this is impossible because J is in the cycle $(DHPJE)$ and Y is in the cycle

$(BGYNUVW)$ which are not of the same length. This means that the fast rotor's initial setting cannot be setting A.

We can follow a similar process and conclude that setting B is incorrect. Now suppose that the fast rotor starts at setting C. Then the fast rotor enciphers R to Y and enciphers T to K. This means that S must have a connection between Y and K. Then when the fast rotor is in the fourth position which corresponds to setting F, we can see the rotor deciphers Y to W and deciphers K to C. Thus, W and C must be diagonal pairs in a pair of cycles. Since C is in the cycle $(AMLQFCK)$ and W is in the cycle $(BGYNUVW)$ and these cycles are of the same length, then in order for W and C to be diagonal pairs, the cycles must be aligned in the following way since the second cycle must be written in reverse order:

$$\begin{array}{ccccccc} A & M & L & Q & F & C & K \\ N & Y & G & B & W & V & U \end{array}$$

In order to check this is the correct alignment we need to check the letters in the columns encipher to each other when the fast rotor is in setting C and the diagonals encipher to each other when the fast rotor is in setting F. Consider the letters A and N. These are in the same column, so they encipher to each other when the fast rotor is in setting C. Since the fast rotor enciphers A to D and N to C, then S connects D and C. Then when the fast rotor is at setting F, the fast rotor deciphers D to L and C to Y. Therefore, L and Y must be diagonal pairs which we can see is correct. We can then follow this same process for each of the letters in the same column, and we will be able to see that this alignment is correct. This means we can conclude that the initial settings of the fast rotor is C.

In this example, we happened to have two one cycles which was very useful because we were able to know two letters that had to be enciphered to each other. However, this will not always be the case. For example, suppose we have the following encipherment pairs:

$$\begin{array}{ccccc} AZ & GH & MO & SQ & YR \\ BG & HX & NK & TV & ZA \\ CE & IW & OM & UP & \\ DC & JB & PY & VF & \\ EU & KJ & QI & WD & \\ FN & LT & RS & XL & \end{array}$$

Then the following cycles are produced:

$$\begin{array}{ccccccccccc}
B & G & H & X & L & T & V & F & N & K & J \\
C & E & U & P & Y & R & S & Q & I & W & D \\
A & Z \\
M & O
\end{array}$$

Since we do not have a pair of one cycles, we consider the smallest pair of cycles which are the pair of 2-cycles. Now these cycles have two possible alignments

$$\begin{array}{cc}
A & Z \\
M & O
\end{array}
\qquad\qquad
\begin{array}{cc}
A & Z \\
O & M
\end{array}$$

and we have to consider both of alignments when we are considering a specific setting.

Suppose the fast rotor is at setting A, and we are using the first alignment of the 2-cycles. Then the fast rotor in the first position enciphers A to B and M to Z. Then in the fourth position or setting D, the fast rotor deciphers B to M and Z to D. This implies M and D are diagonal pairs in a pair of cycles which is impossible. Therefore, let us consider the second alignment of the 2-cycles. Then the fast rotor in setting A will encipher A to B and O to Y. Then in setting D, the fast rotor deciphers B to M and Y to X. This implies M and X are diagonal pairs in a pair of cycles which is impossible. This means we can conclude the fast rotor cannot begin at setting A.

Now suppose the fast rotor is at setting B and we are using the second alignment of the 2-cycles. Then the fast rotor in the first position, or setting B, will encipher A to C and O to D. Then in the fourth position or setting E, the fast rotor deciphers C to O and D to Z. This implies O and Z are diagonal pairs which is true. Also the fast rotor in setting B enciphers Z to A and M to M. Then in setting E, the fast rotor deciphers A to L and M to U. This implies L and U are diagonal pairs in a pair of cycles. Therefore the two 11-cycles must be aligned in the following way:

$$\begin{array}{ccccccccccc}
B & G & H & X & L & T & V & F & N & K & J \\
R & Y & P & U & E & C & D & W & I & Q & S
\end{array}$$

Then if we check all of the letters in the same columns, we will see that this alignment is correct, so we can conclude that the fast rotor's initial setting is B. Therefore, even though we did not have a pair of 1-cycles, we were still able to find the initial setting.

4.4 Finding the setting of the middle and slow rotors

After the setting of R_1 has been found, then we must find the setting and rotor number for R_2 and R_3. Since we know the permutation corresponding to R_1 and we know the permutation corresponding to $(R_1)(S)(R_1)^{-1}$ then we can solve for S. Now we know $S = (R_2)(R_3)(U)(R_3)^{-1}(R_2)^{-1}$ and we know the permutation corresponding to the reflector U. Then we can go through each possible combination of rotors and rotor positions until we have found the combination that produces the permutation S.

In order to make this easier we can construct a set of tables which represents all of the possible permutations produced by the different combinations of rotors and rotor positions. There are $2 \times 26 \times 26$ different possible combinations which means there are 1352 possible combinations to check. Without computers this is a difficult task, so the Polish cryptanalysts developed a simulator called the Bombe that checked all of the combinations.

Once the Bombe has found the correct combination, then we have successfully found the order of the rotors and the universal initial setting. This means we can then decode any of the messages sent during the day that those universal initial settings are being used.

5 Conclusion

Even though the Polish were able to solve the Enigma code in 1932 which was a very remarkable feat, the Enigma code continued to be altered throughout the rest of the war. In 1938, the Germans decided to increase the security of the Enigma by adding two new scramblers. Therefore, the operators had the choice of using any three of the now five rotors to encode their messages. This made the number of possible arrangements of the rotors change from 6 to 60. Then another month later the Germans also decided to increase the number of plugboard cables from six to ten. With these new alterations to the machine, the code was much more difficult to break and Rejewski did not have the resources in order try and find a way to enhance his work and break the Engima code again.

However, his work was still beneficial to other countries. In 1939, the Polish decided to share Rejewski's work with the Allies which greatly helped the British mathematicians at Bletchley Park. Because Bletchley Park had a larger staff and more resources, they were able to cope with the fact that the Germans had added rotors and plugboard cables. Eventually Alan Turing along with his fellow mathematicians at Bletchley Park were able to again crack the Engima code, and they continued to inovate their methods as the Germans altered the Enigma machine throughout the rest of the war [3].

The ability of the Allies to decode the German messages has been argued as the key factor in the Allies victory. Even though it is not certain whether the Allies would have been able to win the war without breaking the Enigma code, historians assert it shortened the war by 6 months to a year which resulted in saving lives on both sides of the war [2].

There are two main lessons that can be learned from the history of the Enigma machine. One is that mathematics is not only a beautiful subject but it can also have a large impact on world events. The other is that even the most secure cryptosystems can be broken if not used correctly. The only reason the Polish mathematicans were able to break the Enigma code was because of the way the German operators used the machines. Similarly we have the same problem today with hackers being able to break into an individual's account because of the person's poor choice of a computer password. Therefore, we should not only appreciate the beautiful mathematics used in the process of breaking the Enigma machine but also regard it as a warning to not use poor operational procedures when it comes to using cryptosystems.

References

[1] Chris Christensen. Polish mathematicians finding patterns in enigma messages. *Mathematics Magazine*, 80:247–273, 2007.

[2] R. F. Churchhouse. A classical cipher machine: The enigma - some aspects of its history and solution. *The Institute of Mathematics and its Applications*, 27:129–137, 1991.

[3] Simon Singh. *The Code Book: How to Make It, Break It, Hack It, Crack It.* Delacorte Press, New York, NY, 2001.