The Enigma Machine

Grace Howard

August 2024

Introduction

The Enigma machine is perhaps one of the most notorious cipher devices ever used. To begin, some historical context is provided before the machine is introduced in more detail. Then, the components of the army Enigma machine are detailed and explained. From there, an explanation of some of the methods used for decryption are included. Some explanations of difficulty are also detailed. After this, the symmetric group and its applications to decryption are described. Finally, the Bombe is briefly explained.

Historical Context

The Enigma machines were a series of cipher machines which used rotors, a plugboard, and a reflector to encrypt letters. The machines would have been considered unbreakable had they been operated exactly as intended. It was in late 1932 that Marian Rejewski broke the Enigma using group theory. He also invented a device, the bomba kryptologiczna, which would aid in decrypting intercepted messages.

Following the outbreak of World War II, the British Government Code and Cypher School at Bletchley Park worked to decrypt messages which had been encrypted by the German navy much more carefully than before. A number of individuals, including Alan Turing, worked to upgrade the bomba kryptologiczna. Although this was successful for some time, the German navy eventually added an additional rotor making the Enigma functionally unbreakable once more.

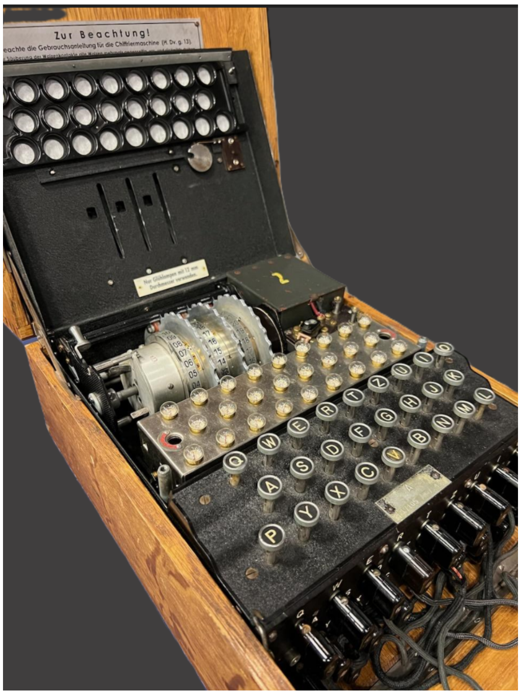


Figure 1: A commercial Enigma which has been re-wired to allow for modern battery use. Note that the cover over the lampboard and rotors has been lifted.

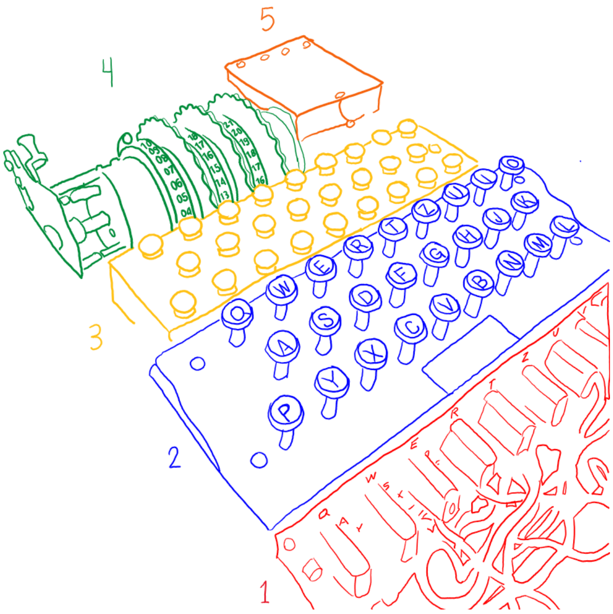


Figure 2: The main parts of the Enigma are labeled. In red (1) is the plugboard. In blue (2) is the keyboard. In yellow (3) is the lampboard. In green (4) are the rotors and reflector. In orange (5) is the power source.

Enigma Machine

Definition 0.1 (Enigma Machine)**.** An army Enigma machine consists of a keyboard, a plugboard, a lampboard, an entry drum, three rotors, and a reflector.

The Enigma machines have many components, but only some contributed to the encryption

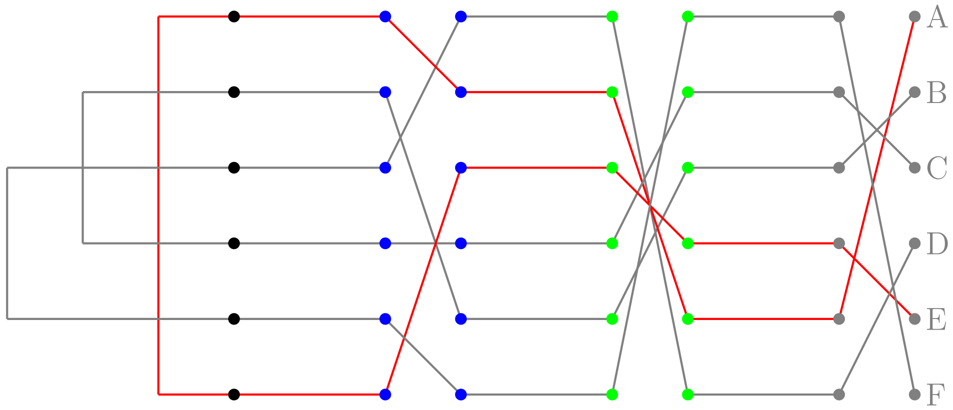


Figure 3: A simplified example of configuration before input. First the current enters as an `A` before going through the three rotors. After going through, it exists as an `E`. Its path is marked in red. The first rotor would then turn one position, leading to the configuration in Figure 4.

of the letters:

Definition 0.2 (Keyboard and Lampboard)**.** The keyboard was the input mechanism of the machine. When an input letter was encoded, the output would light up on the lampboard, which had the same arrangement of letters as the keyboard.

Definition 0.3 (Plugboard)**.** The plugboard, which had the same arrangement of letters as both the keyboard and lampboard, consisted of 26 sockets. Some letters were connected via a short cable to other letters.

Definition 0.4 (Entry Drum)**.** The entry drum is a stationary connection between the plugboard and rotors. It does not aid in encryption and functions only as a means for the current to flow.

Definition 0.5 (Rotors)**.** An Enigma rotor has 26 electrical contacts (per side) equally spaced around the circumference. Within the rotor, each contact is connected to another of the other side of the rotor in a random fashion. When a letter is input, the first rotor N turns one position. When N passes a specific letter (dependent on their configuration), the second rotor M turns one position. Similarly, when M passes a specific letter, the third rotor L turns one position. This *turnover* position was determined by the rotor's ring which could be rotated and locked into place. This allowed for the position of the turnover to vary.

Definition 0.6 (Reflector)**.** Once the current has gone through all three rotors, it passes through the reflector. The reflector is stationary with 26 contacts on only one side. The current would enter the reflector through the right side and exit on the right side, unlike the rotors. Now, the current would flow from left to right through the rotors, eventually sending it back through the plugboard to the lampboard. This ability to "reverse" the current allowed the same machine to be used for encryption and decryption *assuming one knew the correct settings.*

Example 0.1. Suppose the plugboard settings were as follows:

$$\mathrm{A/R - N/F - E/W.}$$

Only considering the plugboard, `FEWER` becomes `NWEWA`.

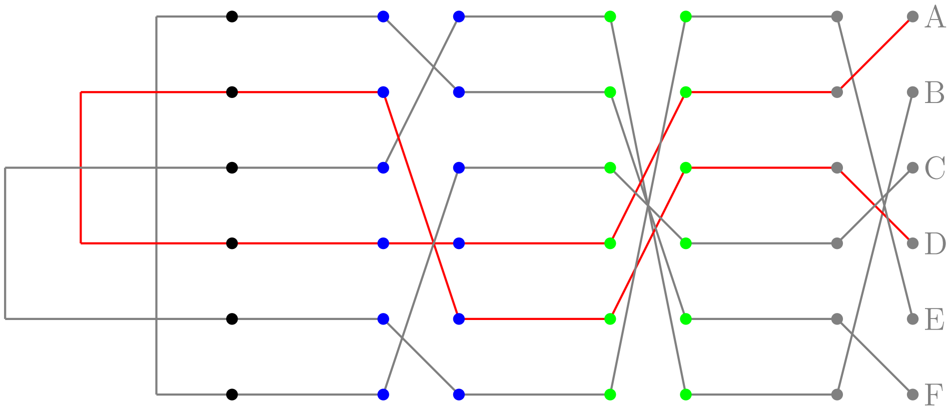


Figure 4: A simplified example of configuration after input. First the current enters as an `A` before going through the three rotors. After going through, it exists as a `D`. Its path is marked in red.

Decryption

Theorem 0.1. No letter can be encrypted as itself.

Proof. Firstly, note that the plugboard maps letters symmetrically. This means, in the absence of the rotors, every letter gets mapped to itself. Suppose, momentarily, that the rotors did not possess the ability to rotate. In the end, they are just a substitution cipher (the intermediate mappings have no impact on the final output). They are also symmetric. However, the reflector means that a letter never gets mapped to itself. So, in the scenario wherein there is no movement from the rotors, the reflector solely ensures a letter cannot be mapped to itself.

This is, however, not the reality of how the machine functioned. In the end, it is not actually relevant that the rotors rotated. It is relevant only *when* they rotated. The encryption of a letter depends on the path of the electric current through the machine. As such, it is important to consider if the position of the rotors moves once the current has entered the circuit. It turns out that it does not. Pressing the keyboard turns the rotors. This means that the mapping will not change until it is pressed again, meaning the restriction to the case of no movement is actually the same as if movement is permitted. With that, no letter can be encrypted as itself. □

This was a major flaw in the Enigma machine. Since some messages were predictable (for example, a weather report was often sent at the beginning of a message), this could be used to start decrypting.

Example 0.2. Consider the following: It is known that the word `FOG` will likely appear in the message. With this, the following process would be used.

Ciphertext	F	D	Q	G	O	P
Phrase	F	O	G	%	%	%

- `FOG` cannot be encrypted as `FDQ` because the `F` matches.

Ciphertext	F	D	Q	G	O	P
Phrase	%	F	O	G	%	%

- `FOG` cannot be encrypted as `DQG` because the `G` matches.

Ciphertext	F	D	Q	G	O	P
Phrase	%	%	F	O	G	%

- FOG can be encrypted as QGO because no letters match.

Continuing in this way, the codebreakers could begin to decrypt messages.

Note, this did not mean that FOG would occur in the message at all. It was, however a starting point.

Remark 0.1. To decipher a message, the following information was needed:

- Information pertaining to the structure of the machine:
  - The wiring between the keyboard, lampboard, and entry drum.
  - The wiring of each rotor.
  - The number and positions of turnover notches on the rotor's rings.
  - The wiring of the reflector.
- Information pertaining to the internal settings:
  - The selection of rotors in use and their ordering.
  - The position of the alphabet ring relative to the core of the rotor.
- Information pertaining to the external settings:
  - The plugboard connections.
  - The rotor positions at the start of the encryption of the message.

The army Enigma had 5 rotors from which 3 were selected, so there were

$$\frac{5!}{(5-3)!} = 60$$

possible wheel orders. Using ten leads, there were

$$\frac{\binom{26}{2} \times \binom{24}{2} \times \cdots \times \binom{8}{2}}{10!} = 150738274937250$$

ways to arrange them on the plugboard. There are

$$26 \times 26 \times 26 = 17576$$

possible three-letter message keys indicating the rotor position at the beginning of the message.

Following the convention of Marian Rejewski, let S = plugboard, L = rotor 3, M = rotor 2, N = rotor 1, R = reflector, and H = entry drum. Therefore, a message's path through the Enigma is:

$$SHNMLRL^{-1}M^{-1}N^{-1}H^{-1}S^{-1}.$$

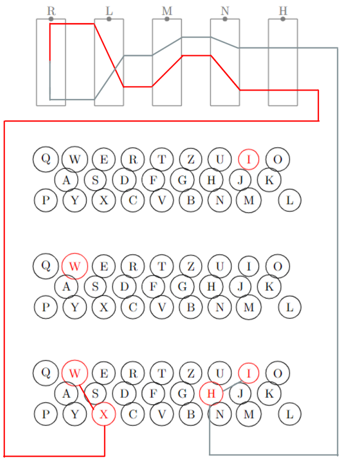


Figure 5: When the `W` is pushed on the keyboard, the current flows from the `W` on the plugboard to the `X`, since those are connected. Then it enters the rotors through the entry drum, marked H. It then enters the first, second, and third rotors before going through the reflector and passing back through the rotors to the entry drum. It then enters the plugboard as an `H` and gets passed to the `I`, since they are connected. So, finally the `I` lights up on the lampboard.

The Symmetric Group

The Symmetric group $S(n)$ was instrumental in the work of Marian Rejewski, and ultimately led to the first breaking of the Enigma.

Definition 0.7 (Permutation)**.** Let X be a finite set. A *permutation* π of X is a bijection from X to X which can be shown as

$$\pi = \begin{pmatrix} a & b & c & \cdots \\ \pi(a) & \pi(b) & \pi(c) & \cdots \end{pmatrix}.$$

Definition 0.8 (Cycle)**.** A permutation π of a set X is called a cycle of length r if there exist r distinct elements $x_1, x_2, \dots, x_r \in X$ such that

$$\pi(x_1) = x_2, \pi(x_2) = x_3, \dots, \pi(x_{r-1}) = x_r, \pi(x_r) = x_1,$$

and $\pi(x) = x$ for any other $x \in X$.

Definition 0.9 (Product)**.** Given two permutations π and σ, the composition $\pi\sigma$, defined as

$$\pi\sigma(x) = \pi(\sigma(x)),$$

is called the product of these permutations.

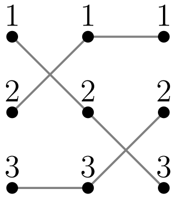
Definition 0.10 (Symmetric Group)**.** The *symmetric group* S_n is the group of all permutations of the set $\{1, 2, \dots, n\}$.

Definition 0.11 (Disjoint Permutations)**.** Two permutations π and σ are called *disjoint* if the set of elements moved by π is disjoint from the set of elements moved by σ.

Definition 0.12 (Cycle Decomposition)**.** Any permutation of a finite set can be expresses as a product of disjoint cycles. This is a cycle decomposition.

Definition 0.13 (Transposition)**.** A permutation σ which changes two characters and leaves all other letters unchanged is called a *transposition*. A transposition which swaps i and j is denoted (ij).

Example 0.3. Consider (23)(12). This represents



or

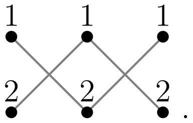
$$
\begin{aligned}
1 \mapsto 2 \mapsto 3 \\
2 \mapsto 1 \mapsto 1 \\
3 \mapsto 3 \mapsto 2.
\end{aligned}
$$

So, $(23)(12) = [3, 1, 2]$.

Definition 0.14 (Pairing)**.** A *pairing* is an element of $S(n)$ which is equal to its own inverse.

So, to formalize the discussion from before each rotor acts as an element of $S(26)$. Additionally, the reflector, which also acts as an element of $S(26)$, is, by construction, a pairing.

Example 0.4. Consider (12)(12):



In general $(ab)(ab) = (1)$. In other words, transpositions are pairings. Additionally, elements that can be written as products of disjoint transpositions are pairings.

Example 0.5. Let $\sigma = (ab)(cd)$ and $\tau = (ad)(bc)$. Firstly,

$$((ab)(cd))^2 = (ab)^2(cd)^2 = (1)$$

and

$$((ad)(bc))^2 = (ad)^2(bc)^2 = (1)$$

so σ and τ are both pairings in $S(4)$. The goal is to compute the product $\tau\sigma$ and $\sigma\tau$. Firstly,

$$\tau\sigma = (ad)(bc)(ab)(cd)$$

which is the permutation

$$a \mapsto c, \qquad b \mapsto d, \qquad c \mapsto a, \qquad d \mapsto b.$$

Therefore, $\tau\sigma = (ac)(bd)$. Next,

$$\sigma\tau = (ab)(cd)(ad)(bc)$$

which is the permutation

$$a \mapsto c, \qquad b \mapsto d, \qquad c \mapsto a, \qquad d \mapsto b.$$

So, $\sigma\tau = (ac)(bd)$. Note that in general the product of two pairings is not a pairing.

Theorem 0.2. Let σ and τ be pairings in $S(n)$ which move and fix exactly the same letters. Then, the number of disjoint cycles in $\sigma\tau$ of every length is even.

Consider the pairings

$$\sigma = (ae)(bf)(cg)(hd)$$

and

$$\tau = (be)(fc)(hg)(ad)$$

which move only the letters a, b, c, d, e, f, g. The theorem says that there an an even number of cycles of each length in $\sigma\tau$. Consider the following arrangement:

$$\begin{array}{ccccccccc} a & & h & & c & & b & & a \\ & d & & g & & f & & e. & \end{array}$$

With this, τ is diagonally down from left to right, whereas σ is diagonally up from left to right. So, the disjoint cycle decomposition of $\sigma\tau$ is seen by reading the top row from left to right to get one cycle, and the bottom row from right to left to get the other. Thus, the cycles of each length occur in pairs. Considering this, it is apparent that this can be used on the product of any two pairings that move the same letters.

Remark 0.2. A substitution cipher is a permutation of the alphabet, and thus an element of the group $S(26)$.

Suppose that the following first six letter groups came from fifteen messages all intercepted on the same day:

```
FOWVAT   WRTYUO   QVTNMO   KOPHAU   EVPRMU
QMLNXZ   WVQYMK   DGYBHJ   ORCDLA   MIJWCE
ABOCRH   COEIAW   NTPLBU   ZUGMCF   LHMQZP
```

.

There is an interesting pattern amongst them; when two of the strings start with the same letter, they have the same fourth letter as well. Similarly, when two of the strings have the same second letter, they have the same fifth letter and when two of the strings have the same third letter, they have the same sixth as well. Now consider stringing together the first and fourth letters for the intercepted strings. Continuing the previous example, this gives

$$aci \ldots zmwy \ldots qnlq \ldots odb \ldots fv \ldots er \ldots kh \ldots$$

From this, sometimes one could string together the entire alphabet, which is an element $\pi \in S(26)$. This permutation π can be factored into disjoint cycles. Continuing with the above example, (qnl) is a cycle. It was noted that the cycles of length k occurred in pairs. From this, the aforementioned mathematician Marian Rejewski realized that he could decompose the element of $S(26)$ into two pairings. He would do this again for the second and fifth and third and sixth.

Bombe

The Polish Bomba was produced by the same company which produced the Polish copies of the Enigma used for analysis during World War I. It was three pairs of Enigma duplicates which had been wired together. At the time, the machine found solutions in less than 2 hours. However, around 1938 additional rotors were added to the rotation employed by the Enigma. With no way of knowing which three of the five possible rotors had been selected, the Bomba would now need sixty Enigma duplicates.

Definition 0.15 (Bombe)**.** The bombe was a machine which acted as several Enigma machines acting together. The British bombe contained the equivalent of 36 Enigma machines which each had three drums acting as rotors.

Later, at Bletchley Park, the Turing-Welchman Bombe would be created. Similar to the Polish Bomba, the Bombe would also run through all possible settings. However, as opposed to looking for the one correct rotor setting based on indicators, the Bombe searched all the settings and disregarded those that were incorrect.

Definition 0.16 (Crib)**.** A crib is a plaintext passage of some length which was typically obtained by decrypting some number of ciphertext messages that was believed to occur in a different piece of ciphertext in the hopes of using it as a means to a solution.

For example, if the assumed letter was `H` and the corresponding cipher letter was `R`, the machine ignored any results that did not allow the electrical current to pass from `H` to `R`. By disproving thousands of rotor settings, those which were left were the (possibly) correct settings. Along with advancements made in wiring by Gordon Welchman, the number of possible rotor settings decreased from thousands to only a few.

Definition 0.17 (Known-Plaintext Attack)**.** The *known-plaintext attack* is an attack model where the attacker has access to both the crib and its ciphertext.

Definition 0.18 (Menu)**.** Once a crib was obtained, the operator of the bombe would produce a *menu* for wiring up the bombe to test the crib against the ciphertext.

In February 1942, the naval Enigma machines were finally changed.The Bombe developed by Alan Turing and Gordon Welchman found the rotor settings for an Enigma with three rotors. It could not find the settings for four. This meant that, once again, the messages were indecipherable.