Enigma

Enigma Machines and how the internal wiring of their rotors was reverse-engineered is the topic of this section. We will follow a simplified version of Rejewski's description [1] of his work.

An Enigma Machine applies a series of permutations to each typed letter, mapping it to another letter (which lights up on the Lampboard, see Figure 1.1), thus encrypting a message[2].

Electrical current flows from the typed letter through the plugboard, then the right rotor, the middle rotor and the left rotor. It then enters the reflector and goes back in reverse order through the same components ending up on the lampboard where the corresponding encrypted letter lights up. The plugboard, rotors and reflector have internal wirings which correspond to permutations in S_{26}[3]. The resulting permutation applied to a letter by the Enigma Machine is the product[4]:

$$P^{-1}N_k^{-1}M_k^{-1}L_k^{-1}RL_kM_kN_kP$$

The rotors rotate after each typed letter in the style of an odometer: the right rotor rotates one position after each typed letter, the middle rotor rotates one position after each full-circle rotation of the right rotor and the left rotor rotates one position after each full-circle rotation of the middle rotor. Rotating the rotors changes the permutations they will apply to a letter, so their permutations are indexed by k in the product above and in the Figure 1.1. We will see later how we can model these rotations with permutations.

The reflector pairs each letter with another (always different) letter, thus it is a product of 13 disjoint transpositions. A permutation made out of only disjoint transpositions is called a **proper involution**. We will see why the Enigma Machine designers chose a proper involution for the reflector.

First though we need to collect some facts about permutations that we will use in our Enigma Machine analysis.

[1] Marian Rejewski. How Polish mathematicians broke the Enigma cipher. *IEEE Annals of the History of Computing*, 3(3): 213–234, 1981. ISSN 1058-6180

[2] Enigma machines were used by the Nazis in WWII to encrypt/decrypt messages. The machines are rotor-based electromechanical typewriters. http://en.wikipedia.org/wiki/Enigma_machine has a detailed description of their internals.

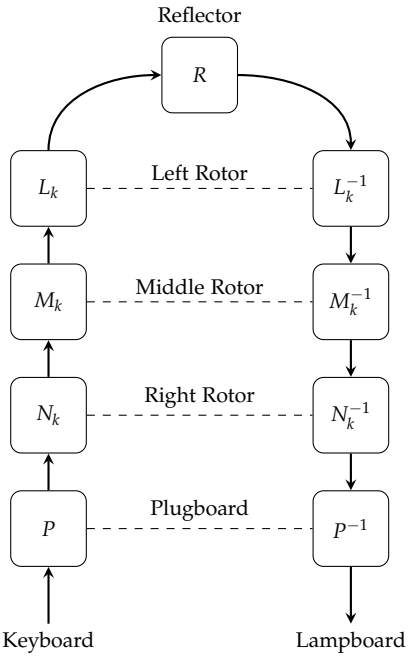


Figure 1.1: Enigma Permutations

[3] S_{26} is the symmetric group of permutations of $\{1, 2, \ldots, 26\}$.

[4] We use the convention of permutation product as function composition, so for $A, B \in S_{26}$ we have $AB = A \circ B$ and $AB(x) = A(B(x))$.

Theorem 1.1. *Every permutation can be written as a product of disjoint cycles. This product is unique (ignoring cycle order and order of elements in cycle).*

Proof. Let $\pi \in S_n$ be a permutation.

We start by choosing an arbitrary $x_1 \in \{1,\dots,n\}$ and define for it the set

$$T_{x_1} = \{x_1, \pi(x_1), \pi^2(x_1), \dots\}$$

$\{1,\dots,n\}$ is finite, $T_{x_1} \subseteq \{1,\dots,n\}$, so T_{x_1} is finite too. This means that sooner or later there exist $i < j$ with $\pi^j(x_1) = \pi^i(x_1)$ or $x_1 = \pi^{j-i}(x_1)$. Then $ord(x_1) = j - i$ is the order of x_1. It follows that

$$T_{x_1} = \{x_1, \pi(x_1), \pi^2(x_1), \dots, \pi^{ord(x_1)-1}(x_1)\}$$

and T_{x_1} implies the cycle $(x_1, \pi(x_1), \pi^2(x_1), \dots, \pi^{ord(x_1)-1}(x_1))$. T_{x_1} is called the π-orbit of x_1. Let's denote this cycle

$$\langle x_1 \rangle = (x_1, \pi(x_1), \pi^2(x_1), \dots, \pi^{ord(x_1)-1}(x_1))$$

We now choose an arbitrary $x_2 \in \{1,\dots,n\} \setminus T_{x_1}$. If there is no such x_2 we stop this process and jump to the section in the proof after all x_k have been chosen. We similarly define T_{x_2} and cycle $\langle x_2 \rangle$.

T_{x_2} and T_{x_1} are disjoint[5].

We continue and choose an arbitrary $x_3 \in \{1,\dots,n\} \setminus (T_{x_1} \cup T_{x_2})$, and in general an arbitrary

$$x_k \in \{1,\dots,n\} \setminus (\bigcup_{i=1}^{k-1} T_{x_i})$$

Since all the T_{x_i} are non-empty and $\{1,\dots,n\}$ is finite, we eventually have to stop. We then have chosen $x_1, x_2, \dots, x_k$ and the corresponding sets $T_{x_1}, T_{x_2}, \dots, T_{x_k}$ and cycles $\langle x_1 \rangle, \langle x_2 \rangle, \dots, \langle x_k \rangle$.

The sets T_{x_i} and their corresponding cycles are by construction pairwise disjoint. We also have $\{1,\dots,n\} = \bigcup_{i=1}^{k} T_{x_i}$.

We define the permutation ρ as the product of the cycles chosen above:

$$\rho = \prod_{i=1}^{k} \langle x_i \rangle$$

and show that $\rho = \pi$.

For all $y \in \{1,\dots,n\}$ there exists a unique $1 \le i \le k$ such that $y \in T_{x_i}$.[6]

So $y = \pi^j(x_i)$ for some index $0 \le j < ord(x_i)$. Since the cycles are disjoint, only cycle $\langle x_i \rangle$ from ρ affects y. We have

[5] Assume $y \in T_{x_2} \cap T_{x_1}$. Then $y = \pi^i(x_1)$ and $y = \pi^j(x_2)$. It follows that $x_2 \in T_{x_1}$ or $x_1 \in T_{x_2}$, either one of which contradicts how x_2 was chosen. Another way to see this is by defining the following relationship: $\forall a, b \in S_n : a \sim b \equiv \exists n \in \mathbb{N} : b = \pi^n(a)$. It's not hard to see that $a \sim b$ so defined is an equivalence relationship and with it the T_{x_i} become equivalence classes and partition S_n.

[6] Because $\{1,\dots,n\} = \bigcup_{i=1}^{k} T_{x_i}$ and $T_{x_1}, T_{x_2}, \dots, T_{x_k}$ are pairwise disjoint and form a partition of $\{1,\dots,n\}$.

$$\begin{aligned}\rho(y) &= \langle x_i \rangle(y) \\ &= \langle x_i \rangle(\pi^j(x_i)) \\ &= \pi^{j+1}(x_i) \\ &= \pi(\pi^j(x_i)) \\ &= \pi(y)\end{aligned}$$

□

Given two permutations $\pi, \rho \in S_n$, the product $\rho\pi\rho^{-1}$ is called a conjugate of π.

Theorem 1.2. *Conjugation preserves cycle structure, i.e. conjugates have cycles of the same length with the same multiplicity.*

Incidentally Theorem 1.2 is the reason why the products $\pi\rho$ and $\rho\pi$ have the same cycle structure. Even though in general $\pi\rho \neq \rho\pi$, $\pi\rho$ and $\rho\pi$ are conjugate. This was the question asked in Exercise 5.5 on page 34 from Michael Artin. *Algebra*. Addison Wesley, 2 edition, 2010. ISBN 0132413779.

Proof. Consider $\pi, \rho \in S_n$. From Theorem 1.1 we know that π is a product of disjoint cycles $\pi = \prod_{i=1}^{k} \rho_i$. For the conjugate $\rho\pi\rho^{-1}$ we can write:

$$\begin{aligned}\rho\pi\rho^{-1} &= \rho\rho_1\rho_2\rho_3 \dots \rho_k\rho^{-1} \\ &= \rho\rho_1(\rho^{-1}\rho)\rho_2(\rho^{-1} \dots \rho)\rho_k\rho^{-1} \\ &= (\rho\rho_1\rho^{-1})(\rho\rho_2\rho^{-1}) \dots (\rho\rho_k\rho^{-1}) \\ &= \prod_{i=1}^{k} \rho\rho_i\rho^{-1}\end{aligned}$$

so it is enough to prove the theorem for a cycle.

Let $\rho = (a_1, a_2, \dots, a_r)$ be a cycle of length r. We have

$$(\rho\rho\rho^{-1})(\rho(a_i)) = (\rho\rho)(a_i) = \rho(a_{i+1})$$

so $\rho\rho\rho^{-1}$ will have the cycle $(\rho(a_1), \rho(a_2), \dots, \rho(a_r))$ with length r. Now assume that x is moved by $\rho\rho\rho^{-1}$, so $(\rho\rho\rho^{-1})(x) \neq x$. It follows that $(\rho\rho^{-1})(x) \neq \rho^{-1}(x)$ or $\rho(\rho^{-1}(x)) \neq \rho^{-1}(x)$. This means that $\rho^{-1}(x) \in (a_1, a_2, \dots, a_r)$ and $x \in (\rho(a_1), \rho(a_2), \dots, \rho(a_r))$. It follows that $\rho\rho\rho^{-1} = (\rho(a_1), \rho(a_2), \dots, \rho(a_r))$.

□

We have seen that an Enigma Machine permutation E is the product

$$\begin{aligned}E &= P^{-1}N_k^{-1}M_k^{-1}L_k^{-1}RL_kM_kN_kP \\ &= (L_kM_kN_kP)^{-1}R(L_kM_kN_kP) \\ &= QRQ^{-1}\end{aligned}$$

with $Q = (L_kM_kN_kP)^{-1}$. This means that E is a conjugate of the reflector permutation R, and according to Theorem 1.2 has the same

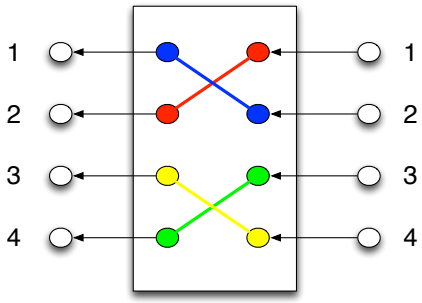


Figure 1.2: Initial Rotor

cycle structure as R. So E is a proper involution (because R is) and also $E^{-1} = E$. The same Enigma Machine configuration was used to encrypt and decrypt a message, which was probably why the Enigma Machine designers chose a proper involution for R and ultimately for E.

Let's analyse the rotor motion on the example in Figure 1.2. It shows a small rotor with an internal wiring doing a permutation from S_4. It has 4 inputs/outputs and permutation $(12)(34)$. Rotating it down one position as in Figure 1.3 doesn't change its internal wiring but shifts the inputs/outputs. Input one is now connected to the yellow wire instead of the red, input two to the red wire instead of the blue, etc. The resulting permutation is $(14)(23)$. The inputs have been shifted according to (4321) and the outputs according to (1234), so $(14)(23) = (1234)(12)(34)(1432)$. In general one rotation of a rotor is equivalent to conjugating it with the full cycle permutation σ, in other words we have $N_{k+1} = \sigma N_k \sigma^{-1}$. To see why this is true, consider input x touches the red wire in the rotor after the rotation. We don't know yet where the rotor will map x. We do know that if x touches the red wire in the rotor after the rotation, then $\sigma^{-1}(x)$ is touching the red wire before the rotation (because all inputs and outputs have been shifted down). Also we know where the rotor maps any input y before the rotation, namely to $N_k(y)$. So $\sigma^{-1}(x)$ is mapped to $N_k(\sigma^{-1}(x))$. And any output from before the rotation is shifted down once after the rotation, in this case to $\sigma(N_k(\sigma^{-1}(x)))$ after the rotation. Collecting this tracing into one expression, we have $N_{k+1}(x) = \sigma(N_k(\sigma^{-1}(x)))$.

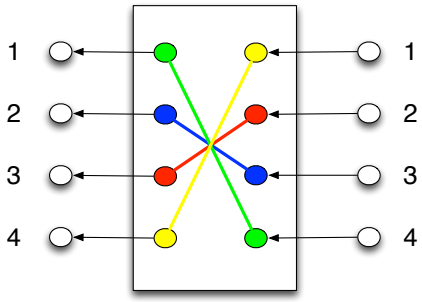
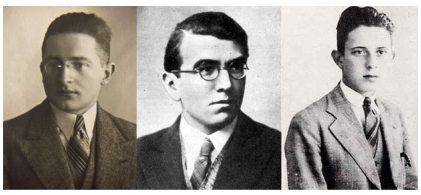


Figure 1.3: Rotor after one rotation

Assuming only the right rotor moves, the first six Enigma permutations are:

$$\begin{aligned}
A &= P^{-1}N_0^{-1}M_0^{-1}L_0^{-1}RL_0M_0N_0P \\
B &= P^{-1}\sigma N_0^{-1}\sigma^{-1}M_0^{-1}L_0^{-1}RL_0M_0\sigma N_0\sigma^{-1}P \\
C &= P^{-1}\sigma^2 N_0^{-1}\sigma^{-2}M_0^{-1}L_0^{-1}RL_0M_0\sigma^2 N_0\sigma^{-2}P \\
D &= P^{-1}\sigma^3 N_0^{-1}\sigma^{-3}M_0^{-1}L_0^{-1}RL_0M_0\sigma^3 N_0\sigma^{-3}P \\
E &= P^{-1}\sigma^4 N_0^{-1}\sigma^{-4}M_0^{-1}L_0^{-1}RL_0M_0\sigma^4 N_0\sigma^{-4}P \\
F &= P^{-1}\sigma^5 N_0^{-1}\sigma^{-5}M_0^{-1}L_0^{-1}RL_0M_0\sigma^5 N_0\sigma^{-5}P
\end{aligned} \tag{1.1}$$

The first six permutations are important because of how the German Nazis chose to operate Enigma. It was known to the code breakers[7] that after configuring Enigma to its daily settings and before sending a message, an operator would send a block of three letters twice. The three letters encoded a message key and because transmission lines were deemed unreliable, these three letters would be sent twice. This means that for each message transmission the input to permutations A and D was the same letter (similar for B and E and for C and F). The


[7] Marian Rejewski, Henryk Zygalski and Jerzy Różycki. http://en.wikipedia.org/wiki/Marian_Rejewski

code breakers had access to two months of intercepted messages and daily key settings. So they could determine that an unknown letter u was mapped by A to the observed letter x and by D to the observed letter y, so $A(u) = x$ and $D(u) = y$. Because A and D are each proper involutions[8] it also holds that $A(x) = u$ and $D(y) = u$. It follows that $AD(y) = A(D(y)) = A(u) = x$. So AD maps one observed letter to another observed letter. With enough messages in a given day, each letter of the alphabet will be observed which then completely defines AD and similarly BE and CF. So for a given day AD, BE and CF were known permutations. The goal now is to factor AD into A and D.

We need a way to compute how many possible factorizations there are and a way to generate all possibilities. To accomplish this we need to collect some properties of products of proper involutions. We will use a simplified approach similar to the approach described in chapter 3.8 of Lawrence and Zorzitto [2021][9].

Theorem 1.3. *Let $\pi = \tau\rho$ be the product of proper involutions τ and ρ and let $x \in \{1, \ldots, n\}$. Then the π-orbits of x and $\rho(x)$ are disjoint and have equal length.*

Proof. Reminder here that the π-orbit of x is

$$T_x = \{x, \pi(x), \pi^2(x), \ldots, \pi^{ord(x)-1}(x)\}$$

Assume the two orbits are not disjoint and $y \in T_x \cap T_{\rho(x)}$. For some integers i and j we have $y = \pi^i(x) = \pi^j(\rho(x))$. Let $m = ord(\rho(x))$ and let $(k-1)m < j \leq km$ for some k. Then

$$\pi^{i+km-j}(x) = \pi^{km}(\rho(x)) = \rho(x)$$

Let $n = i + km - j$ and so

$$\rho(\pi^n(x)) = \rho^2(x) = x$$

because ρ is a proper involution (so it is its own inverse).
We now have two cases: n can be even or odd.
When $n = 2l$:

$$\begin{aligned}
\rho\pi^n &= \rho\pi^l\pi^l \\
&= \rho \underbrace{(\tau\rho)(\tau\rho)\ldots(\tau\rho)}_{l\text{-times}} \pi^l \\
&= \underbrace{(\rho\tau)(\rho\tau)\ldots(\rho\tau)}_{l\text{-times}} \rho\pi^l \\
&= \pi^{-l}\rho\pi^l
\end{aligned}$$

This means that $\rho\pi^n$ is a conjugate of ρ and thus it is a proper involution and cannot have x mapping to itself. We have a contradiction.

[8] This is one example of why choosing a proper involution as the encryption permutation was a bad idea for Enigma Machines. As it turns out it was fatally bad: It was the main weakness that allowed the British *bombe machine* built at Bletchley Park by Alan Turing and Gordon Welchman to decrypt Enigma encrypted messages. http://en.wikipedia.org/wiki/Cryptanalysis_of_the_Enigma#British_bombe

[9] J.W. Lawrence and F.A. Zorzitto. *An Introduction to Abstract Algebra: A Comprehensive Introduction*. Cambridge Mathematical Textbooks. Cambridge University Press, 2021. ISBN 9781108836654. URL https://books.google.com/books?id=PvQgEAAAQBAJ

When $n = 2l + 1$:

$$\begin{aligned}\rho\pi^n &= \rho\pi^l\tau\rho\pi^l \\ &= \rho\underbrace{(\tau\rho)(\tau\rho)\dots(\tau\rho)}_{l\text{-times}}\tau\rho\pi^l \\ &= \underbrace{(\rho\tau)(\rho\tau)\dots(\rho\tau)}_{l\text{-times}}\rho\tau\rho\pi^l \\ &= (\rho\pi)^{-1}\tau(\rho\pi^l)\end{aligned}$$

This means that $\rho\pi^n$ is a conjugate of τ and thus it is a proper involution and cannot have x mapping to itself. Again we have a contradiction.

We just showed that the π-orbits of x and $\rho(x)$ are disjoint. To show that the orbits have the same length, we again reach for this useful identity: for any integer m we have $\rho\pi^m = \pi^{-m}\rho$. This is because

$$\begin{aligned}\rho\pi^m &= \rho\underbrace{(\tau\rho)(\tau\rho)\dots(\tau\rho)}_{m\text{-times}} \\ &= \underbrace{(\rho\tau)(\rho\tau)\dots(\rho\tau)}_{m\text{-times}}\rho \\ &= \pi^{-m}\rho\end{aligned}$$

The identity allows for this equivalence:

$$\pi^{-m}(\rho(x)) = \rho(x) \Leftrightarrow \rho(x) = \rho(\pi^m(x)) \Leftrightarrow \pi^m(x) = x$$

It means that the π^{-1}-orbit of $\rho(x)$ has the same length as the π-orbit of x. But π^{-1}-orbit and π-orbit of an element are the same[10]. So the π-orbits of x and $\rho(x)$ have the same length. □

[10] Just walk the cycle backwards.

Theorem 1.4. *Let $\pi = \tau\rho$ be the product of proper involutions τ and ρ and let $x \in \{1, \dots, n\}$. Then the π-orbits of $\tau(x)$ and $\rho(x)$ are equal. In addition to that, the π-orbit of x is mapped by τ and ρ onto this common π-orbit of $\tau(x)$ and $\rho(x)$.*

Proof. Keeping in mind that a proper inversion is its own inverse, we have:

$$\begin{aligned}\rho(x) &= \rho(x) \\ &= \rho(\tau^2(x)) \\ &= (\rho\tau)(\tau(x)) \\ &= \pi^{-1}(\tau(x))\end{aligned}$$

so then $\pi(\rho(x)) = \tau(x)$ and $\tau(x)$ is in the π-orbit of $\rho(x)$.
Using the $\rho\pi^m = \pi^{-m}\rho$ identity again, we see that

$$\rho(\pi^m(x)) = \pi^{-m}(\rho(x))$$

so ρ maps the π-orbit of x onto the π-orbit of $\rho(x)$.
To see where τ maps the π-orbit of x we need a similar identity, so lets deduce it:

$$\begin{aligned}
\tau\pi^m &= \tau \underbrace{(\tau\rho)(\tau\rho)\dots(\tau\rho)}_{m\text{-times}} \\
&= \tau\tau \underbrace{(\rho\tau)(\rho\tau)\dots(\rho\tau)}_{m-1\text{-times}}\rho \\
&= \pi^{-m+1}\rho \\
&= \pi^{-m+1}\pi^{-1}\tau \\
&= \pi^{-m}\tau
\end{aligned}$$

We can use this identity for:

$$\begin{aligned}
\tau(\pi^m(x)) &= \pi^{-m}(\tau(x)) \\
&= \pi^{-m}(\pi\rho)(x) \\
&= \pi^{1-m}(\rho(x))
\end{aligned}$$

so τ also maps the π-orbit of x onto the π-orbit of $\rho(x)$.

□

Theorem 1.5. *Let $\pi = \tau\rho$ be the product of proper involutions τ and ρ and let $x \in \{1, \dots, n\}$. Let $y \notin T_x \cup T_{\rho(x)}$. Then $\rho(y) \notin T_x \cup T_{\rho(x)}$.*

Proof. Assume $\rho(y) \in T_x \cup T_{\rho(x)}$. Two cases:
First case: $\rho(y) \in T_x$. Then $\rho(y) = \pi^m(x)$ for some $m \in \mathbb{N}$. Keeping in mind again that the a proper involution is its own inverse, we apply ρ to both sides to get

$$y = (\rho\pi^m)(x) = \pi^{-m}(\rho(x))$$

so $y \in T_{\rho(x)}$ which is a contradiction.
Second case: $\rho(y) \in T_{\rho(x)}$. We proceed similarly:
$\rho(y) = \pi^m(\rho(x))$ for some $m \in \mathbb{N}$ so

$$y = (\rho\pi^m)(\rho(x)) = \pi^{-m}(\rho\rho)(x) = \pi^{-m}(x)$$

so $y \in T_x$ which is a contradiction.

□

Theorem 1.6. *Let $\pi = \tau\rho$ be the product of proper involutions τ and ρ. Then the cycle lengths of π that are greater than one come in even numbers.*

Proof. Let (ab) be a cycle of τ. We have two cases:

Case 1: (ab) is also a cycle of ρ. Then the product π has cycles (a) and (b) of length one.

Case 2: (ab) is not a cycle of ρ. Then it must have a cycle (ac_1) for some c_1. In τ there must be a cycle (c_1c_2) for some c_2. In ρ again there must be a cycle (c_2c_3) for some c_3, ... (remember, ρ and τ are proper involutions, so each element participates in one and only one 2-cycle). We stop with a cycle $(c_{2k}b)$ in ρ, which eventually must happen. Then the product π has cycles $(c_{2k}c_{2k-2}\dots c_2a)$ and $(c_1c_3c_{2k-1}b)$ of length k. □

We are ready to tackle the factorization. To recap, we have a permutation π that we know and we also know it is a product of two proper involutions. Our goal is to find out how many possible factorizations into two proper involutions there are and how do we generate all the factorizations (because we need the factors to determine the first rotor wiring).

Theorem 1.7. *Let $\pi \in S_{2n}$ be a permutation composed of just two disjoint cycles of length n. Then π has exactly n factorizations into two proper involutions.*

Proof. Pick an $a \in \{1,\dots,2n\}$. It is part of one of the two cycles. We are looking for possible $\pi = \tau\rho$ factorizations, with both τ and ρ being proper involutions. The two π-orbits of the two cycles are $A := T_a = \{a, \pi(a), \dots, \pi^{n-1}(a)\}$ and $B := S_{2n} \setminus A$.

We are going to construct all the possible ρ using the previous theorems as constraints (once a possible ρ is constructed, it also fully determines the other factor, τ).

For example, because of theorem 1.3 we have to pick some element from B for $\rho(a)$: $\rho(a) \in B$. We will argue that once this choice has been made, the complete factorization has been determined. Let's see why. What value should $\rho(\pi(a))$ take ? Again, using the identity $\rho\pi^m = \pi^{-m}\rho$, we get $\rho(\pi(a)) = \pi^{-1}(\rho(a))$. By repeatedly using the identity as we move π-forward in the cycle with a, we move π-backwards in the other cycle and at each stop we make another pair for the proper involution ρ.

Figure 1.4 shows the process for an example with two cycles of length four. After setting an a and picking where to map $\rho(a)$, everything else is determined (the labels in the figure show the expressions determining the relationships).

There are n ways to pick an element from B, hence we can construct n different ρ, so n different factorizations $\pi = \tau\rho$. It doesn't matter

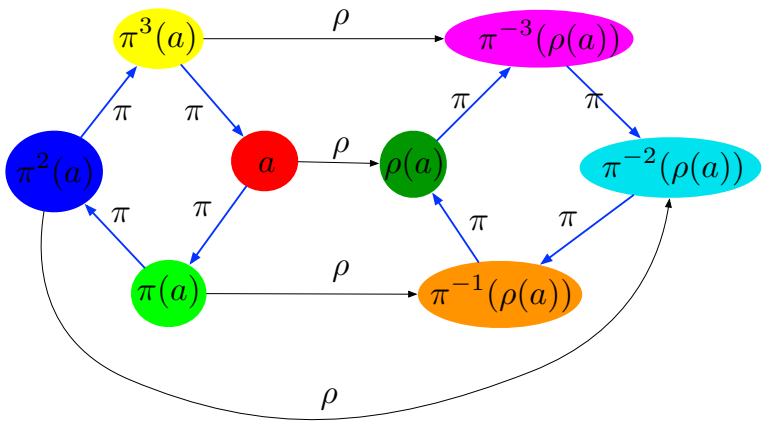


Figure 1.4: Constructing ρ.

which a we start with. Through the cycle-wise rotation in cycle with a and counter-cycle rotation in the cycle with $\rho(a)$, we see all n possible ρ constructions, regardless which a is our anchor. Does it matter from which cycle we choose the anchor ? It doesn't because again the same factorizations would be produced if all the ρ-arrows in figure 1.4 were reversed.

□

Now theorem 1.6 assures us that any product of two proper involutions has cycle lengths greater than one occurring an even number of times. We can always pair up two cycles of the same length. Theorems 1.4 and 1.5 help us isolate the pairings and construct the factors by restricting ourselves to each pairing and using the construction from theorem 1.7 to build the possible factors for each paired restriction. We multiply all the restricted τ's to get the unrestricted τ and multiply all the restricted ρ's to get the unrestricted ρ.

So how many factorizations are there for a given product π? Lets say π has $2m_k$ cycles of length k. In how many ways can we pair up these $2m_k$ cycles?

Theorem 1.8. *The number of ways W to form m pairs from the integers $\{1, 2, \ldots, 2m\}$ is*

$$W = \frac{(2m)!}{2^m m!}$$

Proof. Integer one can be paired with $2m - 1$ other integers. Picking an unpaired remaining integer, it can be paired with $2m - 3$ other integers, etc.

It follows that

$$\begin{aligned} W &= (2n-1)(2n-3)\ldots 5\cdot 3\cdot 1 \\ &= (2n-1)(2n-3)\ldots 5\cdot 3\cdot 1\cdot \frac{(2m)(2n-2)(2n-4)\ldots 4\cdot 2}{(2m)(2n-2)(2n-4)\ldots 4\cdot 2} \\ &= \frac{(2m)!}{(2m)(2n-2)(2n-4)\ldots 4\cdot 2} \\ &= \frac{(2m)!}{2^m m!} \end{aligned}$$

□

Which means that if π has $2m_k$ cycles of length k, we can produce

$$\frac{k^{m_k}(2m_k)!}{2^{m_k}m_k!}$$

factorizations restricted to those cycles. Multiplying over all the possibly cycle lengths greater than one gives us the number of factorizations

$$\prod_{k \text{ cycle length}} \frac{k^{m_k}(2m_k)!}{2^{m_k}m_k!}$$

We return to the first six Enigma permutations 1.1. After using the factorization to factor AD, BE and CF, we know possible solutions for A, B, C, D, E and F. The plugboard settings P were found out from French spies, σ is the full cycle permutation. We can drop the subscripts from M and L because we assume they don't rotate for the first six typed letters and from N because we know how to express its rotations. We get:

$$\begin{aligned} A &= P^{-1}N^{-1}M^{-1}L^{-1}RLMNP \\ B &= P^{-1}\sigma^{-1}N^{-1}\sigma M^{-1}L^{-1}RLM\sigma^{-1}N\sigma P \\ C &= P^{-1}\sigma^{-2}N^{-1}\sigma^2 M^{-1}L^{-1}RLM\sigma^{-2}N\sigma^2 P \\ D &= P^{-1}\sigma^{-3}N^{-1}\sigma^3 M^{-1}L^{-1}RLM\sigma^{-3}N\sigma^3 P \\ E &= P^{-1}\sigma^{-4}N^{-1}\sigma^4 M^{-1}L^{-1}RLM\sigma^{-4}N\sigma^4 P \\ F &= P^{-1}\sigma^{-5}N^{-1}\sigma^5 M^{-1}L^{-1}RLM\sigma^{-5}N\sigma^5 P \end{aligned} \tag{1.2}$$

The unknowns in equations 1.2 are M, L, R and N. Our goal is to compute N. To simplify working with these equations, we define $G = M^{-1}L^{-1}RLM$, move as many known permutations as we can to the left side of the equations and name the left sides U, V, W, X, Y, Z:

$$
\begin{aligned}
U &:= PAP^{-1} = N^{-1}GN \\
V &:= \sigma PBP^{-1}\sigma^{-1} = N^{-1}\sigma G\sigma^{-1}N \\
W &:= \sigma^2 PCP^{-1}\sigma^{-2} = N^{-1}\sigma^2 G\sigma^{-2}N \\
X &:= \sigma^3 PDP^{-1}\sigma^{-3} = N^{-1}\sigma^3 G\sigma^{-3}N \\
Y &:= \sigma^4 PEP^{-1}\sigma^{-4} = N^{-1}\sigma^4 G\sigma^{-4}N \\
Z &:= \sigma^5 PFP^{-1}\sigma^{-5} = N^{-1}\sigma^5 G\sigma^{-5}N
\end{aligned}
\tag{1.3}
$$

We now multiply subsequent equations to get the following five equations:

$$
\begin{aligned}
UV &= N^{-1}G\sigma G\sigma^{-1}N \\
VW &= N^{-1}\sigma G\sigma G\sigma^{-2}N \\
WX &= N^{-1}\sigma^2 G\sigma G\sigma^{-3}N \\
XY &= N^{-1}\sigma^3 G\sigma G\sigma^{-4}N \\
YZ &= N^{-1}\sigma^4 G\sigma G\sigma^{-5}N
\end{aligned}
\tag{1.4}
$$

We eliminate G by inserting VW into the first equation, WX into the second etc:

$$
\begin{aligned}
UV &= N^{-1}\sigma^{-1}NVWN\sigma N \\
VW &= N^{-1}\sigma^{-1}NWXN\sigma N \\
WX &= N^{-1}\sigma^{-1}NXYN\sigma N \\
XY &= N^{-1}\sigma^{-1}NYZN\sigma N
\end{aligned}
\tag{1.5}
$$

We define the new unknown $H = N^{-1}\sigma^{-1}N$ and get

$$
\begin{aligned}
UV &= H(VW)H^{-1} \\
VW &= H(WX)H^{-1} \\
WX &= H(XY)H^{-1} \\
XY &= H(YZ)H^{-1}
\end{aligned}
\tag{1.6}
$$

So UV, VW etc are conjugated by H. Each of the four equations in 1.6 usually yielded several dozen solutions for H and usually there is only one common solution to all four equations. This gave the code breakers H and thus N, the internal wiring of the right rotor. The second rotor was cracked the same way because the German Nazis switched rotor positions every 3 months[11] and a new rotor slid into the rightmost position. Rejewski and his team had daily keys for two months which happened to overlap with one rotor switching. They didn't have daily keys for a longer period that would span two rotor

Figure 1.5: An Enigma on display at the *Museum für Kommunikation Frankfurt* http://www.mfk-frankfurt.de

[11] It is amazing how little things in cryptography can trip up security of a system and open the doors to attackers. The German Nazis no doubt believed that by switching rotors they would increase the number of possible permutations (correct) and thus increase the security of their system (incorrect).

switchings, so they couldn't use this method to deduce the wiring of the third rotor. It's not clear how Rejewski and his colleagues cracked the wiring of the third rotor and the wiring of the reflector[12], but they did. Using only two months worth of daily keys and intercepted messages the Polish cryptologists were able to deduce the internal wirings of the rotors of the Enigma Machine and with that were able to build a functioning replica of it. This achievement jumpstarted the effort of the British team at Bletchley Park and eventually resulted in the capability of the Allied Forces to listen in on all the transmissions encrypted with Enigma.

[12] For more details and possible solutions, see J. Vábek. On Rejewski's solution of Enigma cipher. In *PROCEEDINGS OF WDS 2006*. MATFYZPRESS, 2006 http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.186.9963&rank=1.

Bibliography

Michael Artin. *Algebra*. Addison Wesley, 2 edition, 2010. ISBN 0132413779.

J.W. Lawrence and F.A. Zorzitto. *An Introduction to Abstract Algebra: A Comprehensive Introduction*. Cambridge Mathematical Textbooks. Cambridge University Press, 2021. ISBN 9781108836654. URL https://books.google.com/books?id=PvQgEAAAQBAJ.

Marian Rejewski. How Polish mathematicians broke the Enigma cipher. *IEEE Annals of the History of Computing*, 3(3):213–234, 1981. ISSN 1058-6180.

J. Vábek. On Rejewski's solution of Enigma cipher. In *PROCEEDINGS OF WDS 2006*. MATFYZPRESS, 2006.