The Enigma

Klaus Pommerening
Fachbereich Mathematik
der Johannes-Gutenberg-Universität
Saarstraße 21
D-55099 Mainz

December 3, 1999—Expanded English version November 9, 2013—last change
19. Januar 2021

1 General Description

For a general description of this German World War II cipher machine see the web page `http://www.staff.uni-mainz.de/pommeren/Cryptology/Classic/6_Enigma/EnigmaDescr.html`.

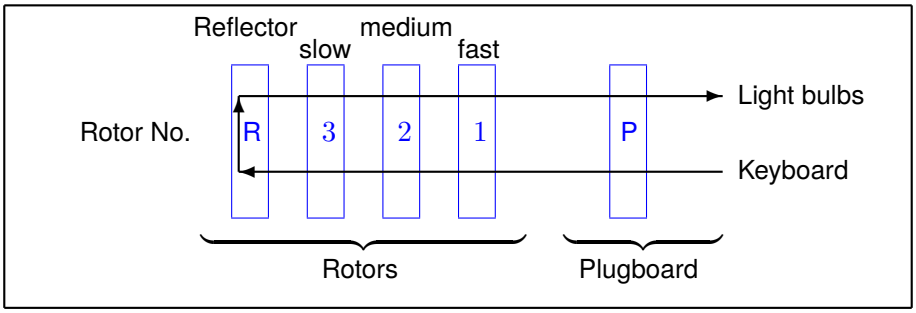

Abbildung 1: *Current flow through Enigma*

2 Mathematical Description

Here we give a mathematical description of the Enigma I ("Wehrmachts-Enigma") with 5 selectable rotors denoted by the roman numerals I to V (whereas the arabic numerals 1 to 3 denote the order in which three rotors are mounted). For a bit of mathematical background on permutations we refer to Appendix A.

The Key Space

The key of an Enigma message has several components:

- The operator choses 3 rotors from a set of 5 and mounts them in a certain order. This gives $\frac{5!}{2!} = 60$ different options ("Walzenlage").
- He adjusts each of the 3 alphabet rings to one of 26 possible positions. This gives another $26^3 = 17576$ options. Since the alphabet ring of the slow rotor has no effect on the encryption, only $26^2 = 676$ of these options contribute to the key space.
- He inserts 10 plugs into the plugboard. Each plug connects 2 letters. He has $\frac{26!}{(2^{10} \cdot 10! \cdot 6!)} = 150,738,274,937,250 \approx 1.5 \cdot 10^{14} \approx 2^{47}$ different choices. This formula is derived in Appendix A. If the operator is allowed to use also less than the maximum 10 plugs this number grows to about $2.1 \cdot 10^{14}$.
- Finally he sets the rotors to their initial positions, another $26^3 = 17576$ possibilities.

Multiplied together these numbers make up a key space of

$$60 \cdot 676 \cdot 150,738,274,937,250 \cdot 17576 = 107,458,687,327,250,619,360,000$$

$$\approx 10^{23} \approx 1.4 \times 2^{76}$$

or a key length of 76 bits (in modern language). However it is not clear at all (and even hardly likely) that all keys define different substitutions. Therefore we can conclude only that the effective key length is *at most* 76 bits. And 47 of these bits are due to the plug-board.

The Control Logic

The current flows through the three movable rotors first from right to left. Accordingly we denote the fast rotor by 1, the middle one by 2, and the slow one by 3. Taking the irregularity in the stepping of rotor 2 into account, and denoting the position of the notch that moves the next rotor by m_i, the formula for the state transition function is

$$g(z_1, z_2, z_3) = (z_1, z_2 + \lambda_1(z_1) + \lambda_1(z_1)\lambda_2(z_2), z_3 + \lambda_1(z_1)\lambda_2(z_2))$$

where $\lambda_i(x) = \delta_{x,m_i}$ is the KRONECKER symbol.

Due to the direction of the labeling of the rotors and the corresponding wiring between input keys or output bulbs and rotors, the substitution by a single rotor in step i is $\rho^{(i)} = \tau^{-i} \circ \rho \circ \tau^i$ where ρ is the rotor substitution and τ the alphabet shift, as explained in Chapter 5.1.

The Enigma Substitution

The rotors being in the state $z = (z_1, z_2, z_3)$ the rotor substitution describes the effect of transversing them from right to left:

$$\sigma_z := \rho_3^{(z_3)} \circ \rho_2^{(z_2)} \circ \rho_1^{(z_1)}$$

The effect of the reflecting rotor is a proper involution π, no element is mapped to itself. The plug-board also provides an involution, η. Together this gives the **Enigma substitution** in state z:

$$\rho_z = \eta^{-1} \circ \sigma_z^{-1} \circ \pi \circ \sigma_z \circ \eta$$

or, with more details, the **Enigma equation** for encryption

$$c_i = \eta^{-1}\tau^{-z_1}\rho_1^{-1}\tau^{z_1-z_2}\rho_2^{-1}\tau^{z_2-z_3}\rho_3^{-1}\tau^{z_3}\pi\tau^{-z_3}\rho_3\tau^{z_3-z_2}\rho_2\tau^{z_2-z_1}\rho_1\tau^{z_1}\eta\,(a_i)$$

Theorem 1 *The Enigma substitution ρ_z in state z is a proper involution.*

Proof. a) Involution:

$$\rho_z^{-1} = \eta^{-1} \circ \sigma_z^{-1} \circ \pi^{-1} \circ \sigma_z \circ \eta = \rho_z$$

since $\pi^{-1} = \pi$.

b) Proper: Assume $\rho_z(s) = s$ for a letter $s \in \Sigma$. Then

$$\sigma_z\eta(s) = \sigma_z\eta\rho_z(s) = \pi\sigma_z\eta(s)$$

hence $\pi(t) = t$ for $t = \sigma_z\eta(s) \in \Sigma$. This contradicts the fact that π is a proper involution. $\diamond$

Note. The proof didn't use the fact that η is an involution. This limitation of the plug-board had purely practical reasons: It reduced errors in operation. Variable plugs between the keyboard or light-bulbs and the first rotor would give more degrees of freedom. But this would require 26 cables instead of the 10 double-plug cables.

3 Cryptanalysis of Enigma: General Remarks

The number of variants of Enigma and of the corresponding appropriate approaches to cryptanalysis is hardly manageable in an introductory text. For this reason we only treat three selected topics:

1. The Enigma without plugboard
2. Message key analysis after REJEWSKI
3. Wehrmacht-Enigma and known plaintext

Special Features of Enigma

- Control logic: Because the middle rotor moves only after 26 steps, and the slow rotor moves almost never, the ciphertext essentially consists of sections of length 26 where only the fast rotor moves by one position with each step.
- The decomposition of a rotor permutation into cycles is not affected by the plugboard. The substitution by the set of rotors is simply conjugated by the plugboard substitution.
  - If the attacker has enough known plaintext she finds cycles, see Section 7.
  - The diverse rotor orders differ by their cycle types [REJEWSKI's catalogue, TURING's "classes"].
  - In this way the attacker gets information on the rotor order.
- Negative pattern search allows to narrow down the position of known plaintext.

In World War II this last effect allowed for the detection of test messages by the Italians that consisted only of LLL...LLL. This was a stroke of genius by the british cryptanalyst Mavis LEVER who noticed that several cipher messages didn't contain any L. This observation turned out to be an essential step in uncovering the wiring of newly introduced rotors.

4 Cryptanalysis of the Enigma Without Plugboard

The Commercial Enigma

The types C and D of Enigma had a reflecting rotor but no plugboard. They were sold on the free market and could be comprehensively analyzed by everyone.

In the Spanish civil war all parties used the Enigma D. All big powers broke it.

The substitution of the commercial Enigma simplifies to

$$c_i = \sigma_z^{-1} \pi \sigma_z(a_i)$$

where σ_z is the substitution by the three rotors in state $z = (z_1, z_2, z_3)$. The reflecting rotor was fixed during encryption but could be inserted in any of 26 positions.

Searching for Isomorphs

In a section of the text where only rotor 1 moves, the two inner rotors together with the reflecting rotor yield a constant involution $\tilde{\pi}$. If the plaintext for this section (say of length m) is known, then we have equations

$$\begin{aligned} c_1 &= \left[\rho_1^{(z_1)}\right]^{-1} \tilde{\pi} \rho_1^{(z_1)}(a_1) \\ c_2 &= \left[\rho_1^{(z_1+1)}\right]^{-1} \tilde{\pi} \rho_1^{(z_1+1)}(a_2) \\ &\dots \\ c_m &= \left[\rho_1^{(z_1+m-1)}\right]^{-1} \tilde{\pi} \rho_1^{(z_1+m-1)}(a_m) \end{aligned}$$

Hence for $i = 1, \dots, m$ the intermediate text

$$c'_i = \rho_1^{(z_1+i-1)}(c_i) = \tilde{\pi} \rho_1^{(z_1+i-1)}(a_i)$$

is the monoalphabetic image $c'_i = \tilde{\pi}(a'_i)$ of the intermediate text

$$a'_i = \rho_1^{(z_1+i-1)}(a_i)$$

under the involution $\tilde{\pi}$.

Therefore pattern search identifies the fast rotor and its state by testing all rotors and all initial states. For determining a'_i from a_i we have to test all three rotors with all 26 start positions, and determine c'_i from c_i with the same rotor in the same position. This exhaustion comprises $3 \times 26 = 78$ different constellations, each of which has to be tested for a matching pattern. Probably there are several false solutions in addition to the correct one.

The next sieving step uses the fact that $\tilde{\pi}$ is a fixed involution. If for a possible solution we find a coincidence $c'_i = a'_j$ with $j \neq i$, then we test for

$$a'_i \mapsto c'_i = a'_j \mapsto c'_j \stackrel{?}{=} a'_i$$

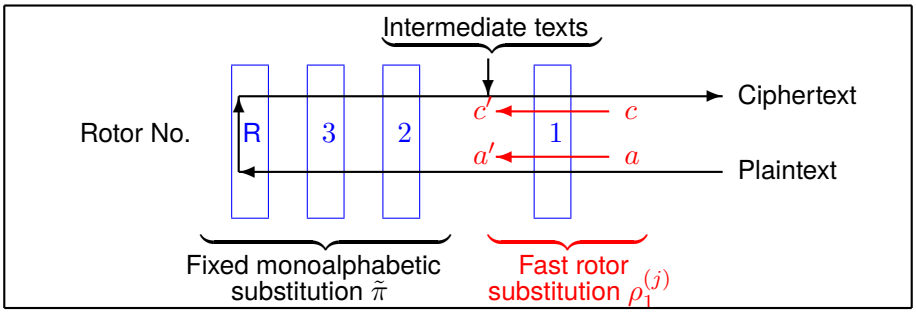

Abbildung 2: *Searching for isomorphs*

If no, we discard the solution. If yes, we even identified a 2-cycle of $\tilde{\pi}$, reducing the number of $26^2 = 676$ possible states of the two inner rotors. A useful tool for this is a precomputed table of length 676 for each of the 6 different combinations of these two rotors that contains the cycle decomposition of $\tilde{\pi}$ for all states, making a total of $6 \times 676 = 4056$ involutions.

Precomputing the lookup table is easy: Let the cycles of π be $(a_1, b_1), \ldots, (a_{13}, b_{13})$. Let $\xi = \rho_3^{(z_3)} \circ \rho_2^{(z_2)}$ be the combined substitution by rotors 2 and 3. Then the cycle decomposition of $\tilde{\pi} = \xi^{-1} \circ \pi \circ \xi$ is

$$\tilde{\pi} = (\xi^{-1}a_1, \xi^{-1}b_1), \ldots, (\xi^{-1}a_{13}, \xi^{-1}b_{13})$$

We only need to apply the fixed substitution ξ^{-1} to the string $a_1 b_1 \ldots a_{13} b_{13}$.

The location of known plaintext, if not known a priori, may be narrowed down by negative pattern search.

Conclusion

The introduction of the reflecting rotor aimed at a significant gain for the security of Enigma by doubling the number of rotor passages. This turned out to be an illusory complication. The attack by isomorphs reduces the cryptanalysis to the exhaustion of position and state of three rotors only, and even this is reduced in a substantial manner.

To prevent this attack the Wehrmacht (= army) introduced the plugboard when adopting the Enigma.

5 Example

Lacking a working simulation for the commercial Enigma we use a military Enigma I omitting the plugboard. Further differences with the commercial Enigma D are

- The reflector is mounted in a fixed position. This will facilitate our task slightly compared with a true Enigma D.
- The rotors (including the reflectors) are differently wired. We consider the wiring as known.
- The input wiring is from keyboard-A to input-A etc., whereas the commercial Enigma had the contacts wired in the order of the keys, i. e. keyboard-Q to input-A, keyboard-W to input-B and so on. This makes no cryptanalytic difference because it amounts to a known renaming of the standard alphabet.
- The notches that move the rotors are fixed at the alphabet rings instead of the rotor bodies, allowing a displacement with respect to the rotor contacts, and thus effecting a slight variablity in the stepping of the rotors. In our example we ignore this complication that is irrelevant for the commercial Enigma.

The primary rotor alphabets are

```
Clear:   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
Rot I:   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
Rot II:  A J D K S I R U X B L H W T M C Q G Z N P Y F V O E
Rot III: 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
Refl B:  Y R U H Q S L D P X N G O K M I E B F Z C W V J A T
```

The cycle decomposition of the reflector is

```
(AY)(BR)(CU)(DH)(EQ)(FS)(GL)(IP)(JX)(KN)(MO)(TZ)(VW)
```

Now assume we got the ciphertext:

```
NMSHH EZJOU OEAJA IDCWS VVMFY IVZQO QWSYO KCEVE QSTLC YMJKT
PFVK
```

We suspect it to be in Spanish but we don't use this conjecture. However it seems likely that it begins with the probable word GENERAL. Negative pattern search yields no contradiction to this assumed known plaintext, however also excludes only very few of other possible positions.

Now we test all three rotors in each possible position in the search for an isomorph. For Rotor I we get 26 pairs of intermediate texts:

```
Pos A: PBURWXL  Pos B: TNQULJH  Pos C: WRNHVNR  Pos D: JUJMBQY
       XWFPJHW   ===>  FEXJQMI         UTMQRGM         QPWRZNP

Pos E: OHTGVDQ  Pos F: IMANAIF  Pos G: PGSOBCP  Pos H: QNHWTJV
       NMCZOOC         JIWOKWH         TSBKHLB         AZCHDHI
```

```
Pos I: YORLYKP  Pos J: NWXHSSU  Pos K: JLREOHV  Pos L: GHWAADN
       SRUDNEJ         HGZNUAR   ===>  RQTUMKG         XWPMBRC

Pos M: CEXKEAS  Pos N: MAPRHWM  Pos O: TKUJUGI  Pos P: LROYZNU
       RQBBLJZ         WVFLRYV         XWIRLIF         POVLQOM

Pos Q: AJKITFY  Pos R: KYWOAUB  Pos S: QIAIBEO  Pos T: KODNJKT
 ===>  UTAQRIE         ONURJNT         KJBJOOD         WVCOIGJ

Pos U: PIQOYEN  Pos V: QNVGUJU  Pos W: IOPLRKV  Pos X: NGWFNCD
 ===>  AZKIELD         DCZEQFI         QPVQUBJ         VUSUXNB

Pos Y: HLXBXHS  Pos Z: DFFNEBO
       POOXKRG         WVYKPUA
```

We find 4 isomorphs, all with the pattern `1234567`. All four yield a contradiction with the involutory property (a "crash"): For position `B` the letter `Q` crashes, for position `K`, `R`, for position `Q`, `T`, for position `U`, `I`.

The analoguous result for Rotor II is:

```
Pos A: TPNTALS  Pos B: VRCWPNF  Pos C: YTUFHPG  Pos D: HWHAWSO
       LKVDRFK         AZBRNAM         NMQNFOO         CBIFUKR

Pos E: CFIORBU  Pos F: QAQKZWJ  Pos G: MOWCSKB  Pos H: EKLRYGQ
       UTXUHCA         HGSHWRV   ===>  IHAWOEJ         QPTOBTF

Pos I: TCDEFYL  Pos J: GRSTUNT  Pos K: VENLCAM  Pos L: NTVYEPS
 ===>  WVZBCLX         LKGCKYM         DCVKQZZ   ===>  SRDQFHO

Pos M: ALOZGHZ  Pos N: BYUHJUO  Pos O: JZBNSVW  Pos P: PHQCNDY
       NMFFXNG         VUHXMCT         ONKMHUU         UTTHPJC

Pos Q: ENYUBJA  Pos R: WCAJXYD  Pos S: LUCEPQM  Pos T: GJFMEFH
       BAOPIEI   ===>  QPCIOMX         YXYOVFP         AZQVKLE

Pos U: OEOFRAV  Pos V: HMJLGIR  Pos W: NFXSYBJ  Pos X: ULTHLHY
       CBFKSSZ         FESSUHH   ===>  ONHUWPA         JIZWZRG
Pos Y: JSLPMOL  Pos Z: RHARUDA
       XWMZITN         TSNIDWC
```

We find 5 isomorphs, again all with the pattern `1234567`. All five contradict an involution.

Finally for Rotor III:

```
Pos A: OAFNPWZ  Pos B: PMSOMIS  Pos C: QNBRJJB  Pos D: TOUOGKC
```

```
       XWJRURV         CBQUHOH         FENHRRI   ===>  SRKRWEJ

Pos E: QRDRSNJ  Pos F: TOEETKG  Pos G: GRLOUND  Pos H: QEITVAA
       BAHWZOM         UTTZMTJ         DCUMVWM         EDVVOJZ

Pos I: VOFWWKM  Pos J: YTCJXPN  Pos K: LWOSNSO  Pos L: UJPLZFP
       LKWOXSJ         IHXXYLO         FEYYFUR         CBOFCVE

Pos M: NSQUAOQ  Pos N: WLRVBHR  Pos O: XUSCEQH  Pos P: EVTZBRT
       ONACZCN         POBZWZG         QPCWIWP         RQFIJTQ

Pos Q: BCJWEYU  Pos R: YZVTRVV  Pos S: VWWFBSY  Pos T: HTXGGPV
 ===>  SRCJKFX         TSFKLGU         JISLMHR         VUCMNIO

Pos U: IFAHJBY  Pos V: JGXIWCL  Pos W: KHAJFDV  Pos X: LINKYEA
 ===>  WVHNDJA         XWKDPKB         AZXPQAC   ===>  XWGQRMD

Pos Y: MJXAHFD  Pos Z: CKCMIGQ
       AZZRUNE         NMIUROF
```

This time we find 4 isomorphs. Only the last one is compatible with an involution. It gives us 7 cycles of the involution $\tilde{\pi}$: `(AD)(EM)(GN)(IW)(KQ)(LX)(RY)`, the letters `BCFHJOPSTUVZ` remaining.

If our assumption on the probable word `GENERAL` was correct, then the fast rotor is Rotor III with initial position `X`. Now we use the lookup table for the involution $\tilde{\pi}$ containing all $2 \times 26^2 = 1318$ possibilities for Rotors I and II in each order and all initial positions. This is the file `vReflB_tr.xls` in the directory `http://www.staff.uni-mainz.de/pommeren/Cryptology/Classic/Files/`. There is exactly one involution that contains the obligatory cycles: The slow rotor 3 is Rotor I in initial position `H`, and the medium rotor is Rotor II in initial position `D`. Trying these settings on the online simulation at `http://enigmaco.de/` we obtain the plaintext

```
General Franco llegara a Sevilla en la noche. Notifica al
alcalde.
```

For successfully cryptanalyzing the Enigma without plugboard we only needed a short cryptogram (54 letters) and a few letters (only 7) of known plaintext. The attack by isomorphs is quite strong.

Compared with the attack on a linearly ordered (“straight-through”) rotor machine the reflecting rotor *reduces* the workload because the involutory property excludes most isomorphs. On the other hand stripping off the last rotor is easier with a straight-through machine. But in summary the reflecting rotor turns out to be an illusory complication.

6 Message Key Analysis by Rejewski

The German Army adopted the Enigma in 1930 as Enigma I. In the first years this variant of the Enigma also had three rotors only—as had the commercial Enigma—but had the rotors wired in another way. Furthermore the additional plugboard, sitting between in/output and the rotors, substantially increased the key space, see Section 2.

The crucial point for the first break-in by the Polish cryptanalysts was a weakness in key handling:

- The key consisted of a daily basic setting and an individual message key.
- The daily basic setting consisted of the rotor order, the ring positions, and the plug connections—first at most 6 plugs—as well as an initial position of the rotors. This setting was valid for all messages of the day—in the first years even for several days. It was known to all participants of the communication network.
- The message key consisted of the initial positions of the three rotors. These could be changed quickly and were to be set by the operator in a random way. This key changed with every message and thereby precluded the alignment in depth of all the messages encrypted with the same daily basic setting.
- The receiver of the message knew the basic setting but not the message key. Therefore the operator encrypted the message key, consisting of three letters, with the basic setting and prefixed this three-letter-cryptogram to the message. This is no diminution of security as long as the keys are selected in a purely random way. In practice they were not.
- Because the radiocommunication was interference-prone, and a distorted key would garble the entire message, the message key was encrypted twice. Thus the proper message had a six-letter prefix. Adding redundancy to a message is not good idea in classical cryptography.

The operator hence had to encrypt six letters, a repeated trigram, using the basic setting, then to set the message key—the rotor positions—and then to encrypt the proper message.

The Polish intercepted the encrypted radio messages of the German Army but couldn't read them—until in 1932 they hired the mathematician Rejewski and his colleagues Różicky und Zygalski.

We describe their approach following Bauer's book [1] whose presentation relies on Rejewski's own description. At first we disregard the obstruction of the analysis that is caused by the (unknown) ring setting, that is, by the unknown stepping of the middle and maybe also the slow rotor.

Some Intercepted Messages

Suppose the first six letters of each of 65 intercepted messages from a single day were (in alphabetic order)

```
AUQ AMN | IND JHU | PVJ FEG | SJM SPO | WTM RAO
BNH CHL | JWF MIC | QGA LYB | SJM SPO | WTM RAO
BCT CGJ | JWF MIC | QGA LYB | SLM SPO | WTM RAO
CIK BZT | KHB XJV | RJL WPX | SUG SMF | WKI RKK
DDB VDV | KHB XJV | RJL WPX | SUG SMF | XRS GNM
EJP IPS | LDR HDE | RJL WPX | TMN EBY | XRS GNM
FBR KLE | LDR HDE | RJL WPX | TMN EBY | XOI GUK
GPB ZSV | MAW UXP | RFC WQQ | TAA EXB | XYW GCP
HNO THD | MAW UXP | SYX SCW | USE NWH | YPC OSQ
HNO THD | NXD QTU | SYX SCW | VII PZK | YPC OSQ
HXV TTI | NXD QTU | SYX SCW | VII PZK | ZZY YRA
IKG JKF | NLU QFZ | SYX SCW | VQZ PVR | ZEF YOC
IKG JKF | OBU DLZ | SYX SCW | VQZ PVR | ZSJ YWG
```

Two observations catch the eye:

1. Frequently even different operators use the same message keys. This could hint at certain stereotypes. Looking for different messages with the same six-letter prefix a coincidence calculation shows that they in fact are encrypted with the same key.

2. The repetition of the three letters of the message key is obvious: If two messages coincide in the first letters, then also their fourth letters coincide. For example a `Z` at position 1 implies a `Y` at position 4. The same holds for positions 2 and 5 (`U` implies `M`) and 3 and 6 (`W` implies `P`).

Therefore the handling of the message keys could be detected from the pure ciphertext, if it was not known already. In any case the cryptanalyst has a lot of ciphertext in depth: The first six letters of each message. If according to the operating instructions the message keys were randomly selected, this observation wouldn't be of much use. However, as it turned out, the message keys were non-random!

REJEWSKI's Approach

REJEWSKI started his analysis by looking at the repeated message keys. Suppose

- $a_1a_2a_3$ is the message key, hence the plaintext starts with the six letters $a_1a_2a_3a_1a_2a_3$.
- The ciphertext starts with the six letters $c_1c_2c_3c_4c_5c_6$.
- The first six Enigma substitutions, starting with the basic setting (+ the first rotor stepping before the first letter is encrypted), are $\rho_1, \rho_2, \rho_3, \rho_4, \rho_5, \rho_6$.

Then we have

$$
\begin{aligned}
c_1 &= \rho_1 a_1, \quad c_4 = \rho_4 a_1, \quad a_1 = \rho_1 c_1, \quad c_4 = \rho_4 \rho_1 c_1 \\
c_2 &= \rho_2 a_2, \quad c_5 = \rho_5 a_2, \quad a_2 = \rho_2 c_2, \quad c_5 = \rho_5 \rho_2 c_2 \\
c_3 &= \rho_3 a_3, \quad c_6 = \rho_6 a_3, \quad a_3 = \rho_3 c_3, \quad c_6 = \rho_6 \rho_3 c_3
\end{aligned}
$$

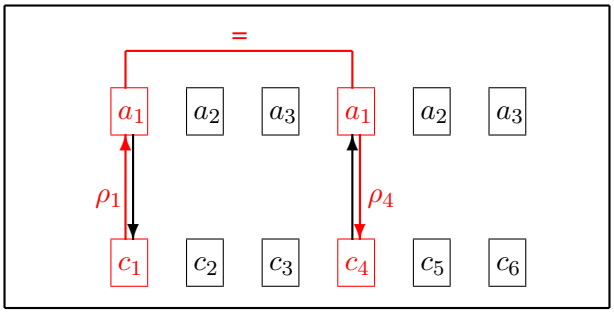

Abbildung 3: *Repeated message key*

Figure 3 illustrates this situation.

The combined permutations $\tau_1 = \rho_4\rho_1$, $\tau_2 = \rho_5\rho_2$, $\tau_3 = \rho_6\rho_3$ are known if we have enough different message keys. In the example the 40 different six-letter groups completely determine τ_1:

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

and τ_2:

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

and τ_3:

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

In REJEWSKI's terminology the triple (τ_1, τ_2, τ_3) was called the **characteristic of the day**.

However we are far from knowing $\rho_1, \ldots, \rho_6$, and far from knowing the basic setting, or even a single message key!

At first sight the plugboard makes trouble. But REJEWSKI as a mathematician knew that the Enigma substitutions with or without plugboard differ only by conjugation with the plugboard substitution η. Therefore there is an *invariant* immune to the effect of the plugboard: the cycle type of the permutations τ_1, τ_2, τ_3, see Appendix A. The cycle decompositions are

τ_1 : `(A)(BC)(DVPFKXGZYO)(EIJMUNQLHT)(RW)(S)` of type $[10, 10, 2, 2, 1, 1]$

τ_2 : `(AXT)(BLFQVEOUM)(CGY)(D)(HJPSWIZRN)(K)` of type $[9, 9, 3, 3, 1, 1]$

τ_3 : `(ABVIKTJGFCQNY)(DUZREHLXWPSMO)` of type $[13, 13]$

From this point the analysis has two possible continuations:

- Assume the rotor wirings are unknown. The cryptanalyst assumes that the message keys are chosen in a stereotypic way—an assumption that in the case of the Wehrmacht-Enigma turned out to be true, see below. This assumption and the material delivered be a German spy and containing the basic settings for a few days including the plug connections enabled RÓŚICKY to derive the wiring of the fast rotor. Since the basic settings changed, each rotor sometimes occupied position 1, so eventually the wirings of all three rotors became known.
- Assume the wirings are known. Then the basic setting can be completely determined and all the messages of the day can be decrypted.

These approaches lead to successes, but not always. REJEWSKI and his colleagues also found some other attack methods, in particular using known plaintext. Here we omit this.

Determining the Enigma Substitution from the Characteristics of the Day

We return to our example and try to determine the first six Enigma substitutions in basic setting, $\rho_1, \ldots, \rho_6$, from the known products $\tau_1 = \rho_4\rho_1$, $\tau_2 = \rho_5\rho_2$, $\tau_3 = \rho_6\rho_3$ whose cycle decomposition is given above. We start with the schema

```
(A) (BC) (DVPFKXGZYO)
(S) (WR) (THLQNUMJIE)

(D) (AXT) (BLFQVEOUM)
(K) (YGC) (NRZIWSPJH)

(ABVIKTJGFCQNY)
(OMSPWXLHERZUD)
```

see Appendix A. We immediately coinclude that ρ_1 and ρ_4 both have the 2-cycle `(AS)`, and ρ_2 and ρ_5 both have the 2-cycle `(DK)`. But even for the 2-cycles of τ_1 we don't get a unique solution: ρ_1 could have the cycles `(BW)(CR)` and ρ_4 the cycles `(BR)(CW)`, or conversely.

To get on we assume—following REJEWSKI—that `aaa` is the most popular message key with the German operators. (If this would turn out as erroneous we would try some other stereotype.) If we are right, then this corresponds to the encrypted message key `SYX SCW` that occurs five times, and implies the cycles

$$\begin{aligned} &\texttt{(AS) in } \rho_1, \quad &&\texttt{(AS) in } \rho_4, \\ &\texttt{(AY) in } \rho_2, \quad &&\texttt{(AC) in } \rho_5, \\ &\texttt{(AX) in } \rho_3, \quad &&\texttt{(AW) in } \rho_6. \end{aligned}$$

This is nothing new for ρ_1 and ρ_4. But for τ_2 it means that the alignment of the 3-cycles is correct, and we read off the 2-cycles

$$\texttt{(AY)(XG)(TC) in } \rho_2, \quad \texttt{(AC)(GT)(XY) in } \rho_5.$$

For τ_3 the correct alignment is

```
(ABVIKTJGFCQNY)
(XLHERZUDOMSPW)
```

and we find the unique solution

$$\begin{aligned}\rho_3 &= \texttt{(AX)(BL)(CM)(DG)(EI)(FO)(HV)(JU)(KR)(NP)(QS)(TZ)(WY)}\\ \rho_6 &= \texttt{(AW)(BX)(CO)(DF)(EK)(GU)(HI)(JZ)(LV)(MQ)(NS)(PY)(RT)}\end{aligned}$$

Now let's look at other encrypted message keys. The first one in our table is `AUQ AMN`, partially decrypting to the plaintext

```
s?s s?s
```

We suspect the stereotypical message key `sss`. If we are right, then ρ_2 has the 2-cycle `(SU)`, and ρ_5 has the 2-cycle `(MS)`. This gives the correct alignment of the 9-cycles ot τ_2:

```
(D) (AXT) (BLFQVEOUM)
(K) (YGC) (JHNRZIWSP)
```

and completely determines ρ_2 and ρ_5:

$$\begin{aligned}\rho_2 &= \texttt{(AY)(BJ)(CT)(DK)(EI)(FN)(GX)(HL)(MP)(OW)(QR)(SU)(VZ)}\\ \rho_5 &= \texttt{(AC)(BP)(DK)(EZ)(FH)(GT)(IO)(JL)(MS)(NQ)(RV)(UW)(XY)}\end{aligned}$$

The encrypted message key `RJL WPX` occurs four times, and partially decrypts as

```
?bb ?bb
```

Again we are quite sure that this reveals a stereotypical message key: `bbb`. We conclude that ρ_1 has the cycle `(BR)`—hence also the cycle `(CW)`—and ρ_4 has the cycle `(BW)`, hence also the cycle `(CR)`.

For the complete solution the only open problem left is the alignment of the two 10-cycles of τ_1. We look at the group `LDR HDE` and partially decrypt it as

```
?kk ?kk
```

We are quite sure of the message key `kkk`. Then ρ_1 has the 2-cycle `(KL)`, the correct alignment is

```
(A) (BC) (DVPFKXGZYO)
(S) (RW) (IETHLQNUMJ)
```

and the complete solution is

$$\begin{aligned}\rho_1 &= \texttt{(AS)(BR)(CW)(DI)(EV)(FH)(GN)(JO)(KL)(MY)(PT)(QX)(UZ)}\\ \rho_4 &= \texttt{(AS)(BW)(CR)(DJ)(EP)(FT)(GQ)(HK)(IV)(LX)(MO)(NZ)(UY)}\end{aligned}$$

Now we can decrypt all message keys for the actual basic setting. However we do not yet know the basic setting itself, and we cannot decrypt a single message. In particular we do not know the ring setting and the positions of the rotors corresponding to the message keys.

REJEWSKI's Catalogue

In our example the permutations $\tau_1 = \rho_4\rho_1$, $\tau_2 = \rho_5\rho_2$, and $\tau_3 = \rho_6\rho_3$ are completely determined and their cycle types are the partitions

```
[10 10 2 2 1 1], [9 9 3 3 1 1], [13 13]
```

of the number 26. Now we ask how characteristic is this triple of partitions for the basic setting of the Enigma. The plug connections are irrelevant for this problem. We consider the rotor order as an element of the permutation group $\mathcal{S}_3$, and the initial positions of the three rotors as elements of the cyclic group $\mathbb{Z}/26\mathbb{Z}$. If we disregard the plugboard and the ring settings, the possible basic settings form the set $\mathcal{S}_3 \times (\mathbb{Z}/26\mathbb{Z})^3$. On the other hand we have the set $\mathcal{P}_{13}$ consisting of all the 101 partitions of the number 13 (in bijective correspondence with the partitions of the number 26 in pairwise equal parts), and we have a map

$$\mathcal{S}_3 \times (\mathbb{Z}/26\mathbb{Z})^3 \longrightarrow (\mathcal{P}_{13})^3$$

We would like this map to be injective. This seems not unrealistic in view of the cardinalities: 105,456 different basic settings, $101^3 = 1{,}030{,}301$ different partitions.

To get the complete value table of this map REJEWSKI designed a simple Enigma simulator called Cyclometer that run through all basic settings in about one year. The result, called REJEWSKI's Catalogue, got lost. But there is a recent reconstruction in the paper

Alex KUHL: Rejewski's Catalog. Cryptologia 31 (2007), 326–331.

It turned out that the above map is *not* injective, but "almost" so: Many triples of partitions have a unique preimage, most have only a few ones. However a few triples occur quite frequently, the top ten being

Triple of partitions			**Frequency**
[13 13]	[13 13]	[13 13]	1771
[12 12 1 1]	[13 13]	[13 13]	898
[13 13]	[13 13]	[12 12 1 1]	866
[13 13]	[12 12 1 1]	[13 13]	854
[11 11 2 2]	[13 13]	[13 13]	509
[13 13]	[12 12 1 1]	[12 12 1 1]	494
[13 13]	[13 13]	[11 11 2 2]	480
[12 12 1 1]	[13 13]	[12 12 1 1]	479
[13 13]	[11 11 2 2]	[13 13]	469
[12 12 1 1]	[12 12 1 1]	[13 13]	466

All in all there are 21230 different triples in the image of the map. 19604 of these, that is 92%, occur at most ten times, the numbers of these rare triples are

Pre-Im	1	2	3	4	5	6	7	8	9	10
Freq	11466	3381	1658	958	660	456	343	265	234	183

Using the catalogue the Polish cryptanalysts usually found the correct basic setting in at most 20 minutes. It is unknown what they did in the exceptional situations where there are too many false positives. Certainly some other useful details could be used. In any case we may assume that the method was successful for at least 92% of all triples, corresponding to roughly 50% of all cases.

We neglected the effect of the ring setting. This causes a rotor movement because the stepping mechanism is connected with the alphabet ring. Now, what could happen? As long as only the fast rotor moves we are in a situation included in the catalogue. The analysis is hampered if the middle rotor moves between two of the first six letters. The chances are 5 of 26 ring settings, that is about 19%. This lowers the total probability of success from 50% to about 40%.

There is even more potential for drawing conclusions from the collected message keys. For example the moving of the middle rotor gives information about the ring setting of the first rotor. An approach to determining the plugboard connections uses the fact that in the first years at most six letter pairs were interchanged. If the cryptanalysts assume that there are no plugs at all, then some true plaintext shows through the tentatively decrypted text. This enables them to reconstruct the plugboard connections.

Epilogue

The plugboard turns out to be an illusory complication: It slows the cryptanalyst down a bit, but not as much as the increase in keylength from 29 to 76 bits—expressed in terms of today—suggested. The main cost of the cryptanalysis is exhausting the rotor order and positions, and this could be made efficient by compiling lookup tables.

By the way the decrypted 40 different message keys from the list of 65 above are:

```
AUQ AMN : sss | IKG JKF : ddd | QGA LYB : xxx | VQZ PVR : ert
BNH CHL : rfv | IND JHU : dfg | RJL WPX : bbb | WTM RAO : ccc
BCT CGJ : rtz | JWF MIC : ooo | RFC WQQ : bnm | WKI RKK : cde
CIK BZT : wer | KHB XJV : lll | SYX SCW : aaa | XRS GNM : qqq
DDB VDV : ikl | LDR HDE : kkk | SJM SPO : abc | XOI GUK : qwe
EJP IPS : vbn | MAW UXP : yyy | SUG SMF : asd | XYW GCP : qay
FBR KLE : hjk | NXD QTU : ggg | TMN EBY : ppp | YPC OSQ : mmm
GPB ZSV : nml | NLU QFZ : ghj | TAA EXB : pyx | ZZY YRA : uvw
HNO THD : fff | OBU DLZ : jjj | USE NWH : zui | ZEF YOC : uio
HXV TTI : fgh | PVJ FEG : tzu | VII PZK : eee | ZSJ YWG : uuu
```

The astonishingly naive habits of the German cipher operators become obvious by looking at the keyboard layout of Enigma:

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

All message keys belong to one of three groups of stereotypes

- iterated letters: `sss`, `fff`, `ddd`, `ooo`, . . .

- three consecutive keys: `rfv`, `rtz`, `wer`, `ikl`, ...
- three letters in alphabetic order: `abc`, `uvw`

Before World War II the British cryptanalysts failed with the cryptanalysis of Enigma because they tried to determine the wiring between in-/output and first rotor. The commercial Enigma D connected `Q` with `A`, `W` with `B`, `E` with `C` and so on in the order of the keyboard. Assuming this for Enigma I didn't work. Rejewski who knew the Germans since he was a student at Göttingen simply assumed that the wiring in any case should follow a simple scheme, and succeeded with the assumption "`A` is connected to `A`, `B` to `B` etc."

The point: Enigma C also had had this simple wiring, and this information could be found in the patent file in the British Patent Office ...

For later attacks (from 1938 on) of the Polish cryptanalysts against the Enigma, including a complete example, see the paper

> David Link, Resurrecting *Bomba Kryptologiczna*: Archeology of Algorithmic Artefacts, I. Cryptologia 33 (2009), 166–182.

7 Wehrmacht Enigma and Known Plaintext

The Polish break into the Enigma relies on the way in which the German operators handled the message keys. With the beginning of the war the method of message keying changed and the pre-war cryptanalytic approaches broke down.

Equations for Known Plaintext

Already the Polish cryptanalysts had exlored the idea of using known plaintext—starting from the observation that the German military in their messages used a lot of stereotypical phrases such as "Heil Hitler" or "Oberkommando der Wehrmacht" (= Army's High Command). Chunks of known plaintext (called "cribs" by the british cryptanalysts) allow narrowing down the exhaustive search to an amount that eventually may be mastered with the help of some cleverly constructed electro-mechanical machines. Alan TURING largely and systematically expanded this approach.

Here is an example (Example 1, taken from [2] as virtually all authors of cryptographic texts do). Let the ciphertext

```
ULOEB ZMGER FEWML KMTAW XTSWV UINZP R ...
```

be given. We suppose the message contains the phrase "Oberkommando der Wehrmacht" near the beginning. A negative pattern search over the first 12 possible positions yields exactly one hit:

```
U L O E B Z M G E R F E W M L K M T A W X T S W V U I N Z P R
o b e r k o = m a n d o d e r w e h r m a c h t
  o b = r k o m m a n d o d e r w e h r m a c h t
    = b e r k o m m a n d o d e r w e h r m a c h t
      o = e r k o m m a n d o d e r w e h r m a c h t
        o b e r k o m m a n d o d e r = e h r m a c h t
===>      o b e r k o m m a n d o d e r w e h r m a c h t
            o b = = k o m = a n d o d e r w e h r m a c h t
              o b e r k o = m a n d o d e r w e h r m a c h t
                o b e r k o m m a n d o d e r = e h r m a c h
                  o b = r k o m = a n d o d e r w e h r m a c
                    o b e r k o = m = n d o d e r w e h r m a
                      o b e r = o m m a n d o d e r w e h r m
```

We assume the rotor wirings of all five rotors as known. The naive approach—exhaustion by brute force and assuming that the ring settings don't interfere with the crib—would go through all 60 possible rotor orders, all $26^3 = 17576$ start positions, and all $> 10^{14}$ plug configurations, each time decrypt the ciphertext, and look if the known plaintext results. The huge number of plug configurations makes this approach hopeless, the "virtual" keylength for this approach being about 67 bits ($10^{23}/26^2 \approx 1.6 \cdot 10^{20} \approx 2^{67}$). (We first neglect the ring settings that have little impact on the cryptanalysis.)

Fortunately, using known plaintext, we may find conditions that involve *only a single plug* . Recall the general situation as shown in Figure 4.

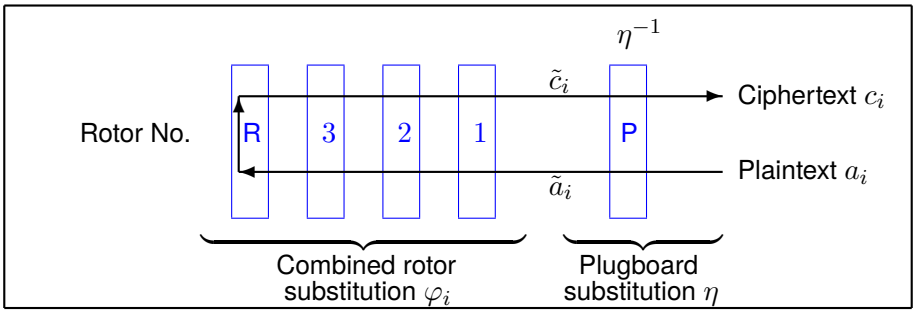

Abbildung 4: *Enigma with plugboard*

Assume a sequence $a_1 \dots a_m$ of known plaintext is given with corresponding ciphertext $c_1 \dots c_m$, the respective combined rotor substitutions being $\varphi_1, \dots, \varphi_m$ and the "full" Enigma substitutions, $\rho_i = \eta^{-1}\varphi_i\eta$. This gives the equations

$$\begin{aligned} c_1 = \rho_1 a_1 &= \eta^{-1}\,\varphi_1\,\eta\,a_1 \\ &\vdots \\ c_m = \rho_m a_m &= \eta^{-1}\,\varphi_m\,\eta\,a_m \end{aligned}$$

or $\eta c_i = \varphi_i \eta a_i$. Denoting the image of a letter under the (fixed but unknown) plugboard substitution by a tilde we get:

Lemma 1 *For a sequence $a_1 \dots a_m$ of known plaintext we have*

$$\tilde{c}_i = \varphi_i\,\tilde{a}_i \quad \textit{and} \quad \tilde{a}_i = \varphi_i\,\tilde{c}_i \quad \textit{for } i = 1, \dots, m.$$

For the second equation we used the fact that the combined rotor substitutions φ_i are involutions.

Looking for Cycles

Returning to Example 1 we consider the special situation

```
                          1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
i   = 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4
c_i = Z M G E R F E W M L K M T A W X T S W V U I N Z
a_i = O B E R K O M M A N D O D E R W E H R M A C H T
```

From such a plaintext-ciphertext pair we extract the TURING **graph**: The nodes correspond to the letters `A` ... `Z` of the standard alphabet. For each pair (a_i, c_i) of plaintext letter and corresponding ciphertext letter an edge is drawn between these two letters, and this edge is labeled by the index i. Due to the reciprocity between plaintext and ciphertext, the situation is modeled by an undirected graph. An edge with label j between nodes s and t means that $t = \rho_j s$ and $s = \rho_j t$—or $\tilde{t} = \varphi_j \tilde{s}$ and $\tilde{s} = \varphi_j \tilde{t}$. Figure 5 shows the TURING graph for Example 1.

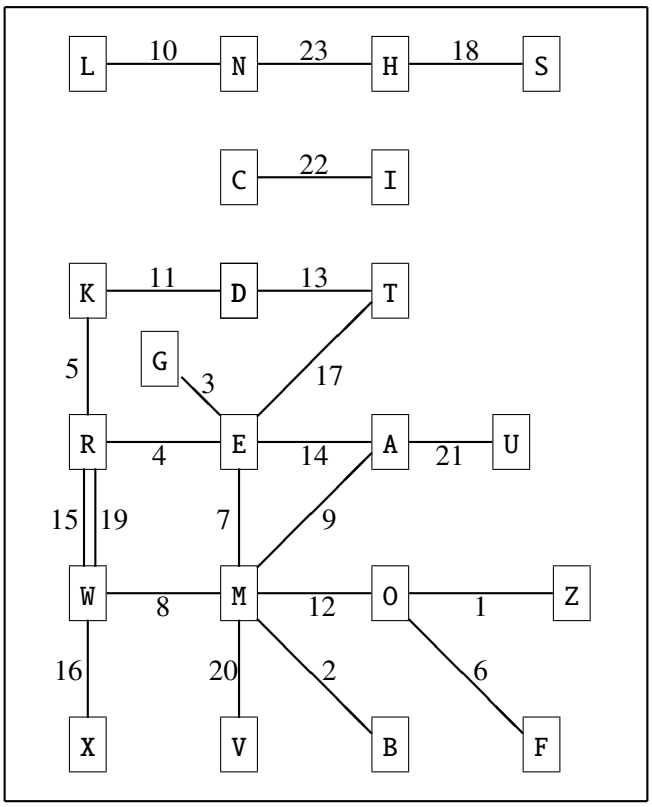

Abbildung 5: TURING *graph for Example 1*

TURINGs approach uses the cycles in this graph ("closures" in TURING's way of speaking). In the notation of Lemma 1 we find:

$$\mathtt{E} = \rho_7\,\mathtt{M}, \quad \mathtt{M} = \rho_9\,\mathtt{A}, \quad \mathtt{A} = \rho_{14}\,\mathtt{E}, \quad \text{and} \quad \tilde{\mathtt{E}} = \varphi_7\,\tilde{\mathtt{M}}, \quad \tilde{\mathtt{M}} = \varphi_9\,\tilde{\mathtt{A}}, \quad \tilde{\mathtt{A}} = \varphi_{14}\,\tilde{\mathtt{E}},$$

and combine these three equations into one **cycle equation**

$$\mathtt{E} = \rho_7\,\rho_9\,\rho_{14}\,\mathtt{E}. \quad \text{and} \quad \tilde{\mathtt{E}} = \varphi_7\,\varphi_9\,\varphi_{14}\,\tilde{\mathtt{E}}.$$

In general we have:

Theorem 2 (Fixed Point Theorem of REJEWSKI/TURING**)** *Let ρ_i be the Enigma substitution in position i, and $\varphi_i = \eta\rho_i\eta^{-1}$ be the substitution without plugs. Then a letter a is a fixed point of a composition $\rho_{i_1} \cdots \rho_{i_k}$ if and only if the plugged letter $\tilde{a}$ is a fixed point of $\varphi_{i_1} \cdots \varphi_{i_k}$.*

Thus the fixed point property of a cycle is in a certain sense independent of the plug connections.

Corollary 1 (TURING**'s cycle condition)** *Each loop in the* TURING *graph gives a necessary condition for the correct key of the Enigma encryption in the form*

$$\tilde{a} = \varphi_{i_1}\,\ldots\,\varphi_{i_k}\,\tilde{a}$$

for a letter a. In particular $\tilde{a}$ is a fixed point of the corresponding composition of unplugged Enigma substitutions.

Although mathematically trivial this theorem and its corollary are the keys to eliminating the complexity of the plugboard by a meet-in-the-middle attack.

What is the benefit of TURING's cycle condition? Suppose in Example 1 we try all 26 possible values for $\tilde{\mathtt{E}} = \eta\,\mathtt{E}$ and all 26^3 possible rotor positions for all 60 possible rotor orders, searching for fixed points of $\varphi_7\,\varphi_9\,\varphi_{14}$—an exhaustion of $60 \times 26^4 = 27,418,560$ cases. Then the probability that the cycle condition is fulfilled is about $1/26$. This rules out $\approx 25/26 \approx 96\%$ of all cases and leaves us with $\approx 60 \times 26^3$ cases—not really impressive, but it could be a good start: Suppose we find two cycles involving E, then we are left with $\approx 60 \times 26^2$ cases, for three cycles with $\approx 60 \times 26$ cases, for four cycles with ≈ 60 cases, i. e. with the exhaustion of the possible rotor orders. And the outcome of this search is:

- The correct initial rotor positions for our known plaintext
- The correct plugboard images for all letters that occur in one of the cycles—a significant part of the complete plug configuration

Now in our Example 1 (that is in fact DEAVOUR's and KRUH's) we see two other cycles involving E:

$$\tilde{\mathtt{E}} = \varphi_4\,\tilde{\mathtt{R}}, \quad \tilde{\mathtt{R}} = \varphi_{15}\,\tilde{\mathtt{W}}, \quad \tilde{\mathtt{W}} = \varphi_8\,\tilde{\mathtt{M}}, \quad \tilde{\mathtt{M}} = \varphi_7\,\tilde{\mathtt{E}},$$

and

$$\tilde{\mathtt{E}} = \varphi_4\,\tilde{\mathtt{R}}, \quad \tilde{\mathtt{R}} = \varphi_5\,\tilde{\mathtt{K}}, \quad \tilde{\mathtt{K}} = \varphi_{11}\,\tilde{\mathtt{D}}, \quad \tilde{\mathtt{D}} = \varphi_{13}\,\tilde{\mathtt{T}}, \quad \tilde{\mathtt{T}} = \varphi_{17}\,\tilde{\mathtt{E}},$$

giving the two additional cycle conditions

$$\tilde{\mathtt{E}} = \varphi_4\,\varphi_{15}\,\varphi_8\,\varphi_7\,\tilde{\mathtt{E}}, \quad \tilde{\mathtt{E}} = \varphi_4\,\varphi_5\,\varphi_{11}\,\varphi_{13}\,\varphi_{17}\,\tilde{\mathtt{E}}.$$

The complete cycle constellation may be visualized by Figure 6.

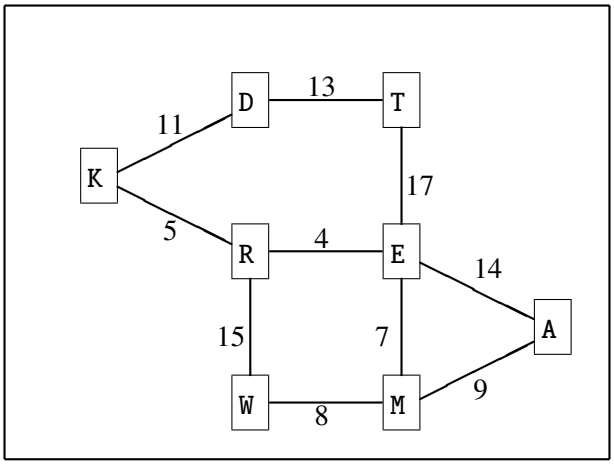

Abbildung 6: TURING *cycles in Example 1*

Evaluating the Cycle Conditions

In evaluating the cycle conditions one sets the rotors to start positions and then steps Rotor 1 only. In lucky cases also in the real situation only Rotor 1 moves. In bad cases Rotor 2 moves, maybe even Rotor 3. Since the ring setting is unknown, these stepping positions are unknown. Because in the example all the cycles are between plaintext positions 4 and 17, the length of the effectively used plaintext segment is 14, and the probability for a stepping of Rotor 2 in between is $13/26 = 50\%$, a stepping that would invalidate the approach, and a good argument for using rather short cribs.

Now assume that we have identified the correct rotor order and the correct initial positions of all the rotors, and no interfering movement of Rotors 2 and 3 occurs for the involved plaintext section $a_1 \dots a_m$. Then the combined rotor substitutions $\varphi_1, \dots, \varphi_m$ are known, and the plug image $\tilde{s} = \eta s$ is known for all letters s that occur in the cycles. In the example we know $\tilde{\mathtt{E}} = \eta\mathtt{E}$ and consequently

$$\tilde{\mathtt{R}} = \varphi_4\tilde{\mathtt{E}}, \quad \tilde{\mathtt{K}} = \varphi_5\tilde{\mathtt{R}}, \quad \tilde{\mathtt{M}} = \varphi_7\tilde{\mathtt{E}}, \quad \tilde{\mathtt{W}} = \varphi_8\tilde{\mathtt{M}}, \quad \tilde{\mathtt{A}} = \varphi_9\tilde{\mathtt{M}},$$
$$\tilde{\mathtt{D}} = \varphi_{11}\tilde{\mathtt{K}}, \quad \tilde{\mathtt{O}} = \varphi_{12}\tilde{\mathtt{M}}, \quad \tilde{\mathtt{T}} = \varphi_{13}\tilde{\mathtt{D}}, \quad \tilde{\mathtt{X}} = \varphi_{16}\tilde{\mathtt{W}}.$$

Furthermore we find $\tilde{\mathtt{F}} = \varphi_6\tilde{\mathtt{O}}$. Since η is an involution the inverse relations might involve further letters. That is we know the plugboard substitutes of at least 11 letters.

What is yet missing is

- The plugboard substitutes of the remaining letters
- The stepping position of Rotor 2

To continue assume first that the remaining letters are unchanged by the plugboard and decrypt $c_{m+1}, \dots$ As soon as the resulting plaintext is unreadable either a new plugboard connection or the stepping position is detected. If the crib occurred in the middle of the ciphertext, we run the same procedure backwards to the beginning of the message.

Conclusion

The huge number of possible plug settings turns out to be an illusory complication: The exhaustion used the plug connection of a single letter only. In good cases where the procedure yields a unique solution of the cycle conditions the effort was testing 26 plug connections with 26^3 start positions for each of the 60 rotor orders, that is $27,418,560 \approx 1.6 \cdot 2^{24}$ cases. In each case we have to do some trial encryptions for the letters in the cycles plus some house-keeping plus some finishing. So we may guess that the search space is dropped to about 30 bits.

As soon as the daily key—rotor order, ring settings, plug connections, initial positions of the rotors—is known, reading all further messages of the day comes for almost no additional costs because all message keys are encrypted with the same initial rotor positions.

A Note on the Technical Realization: TURING's Bombe

TURING's Bombe consisted of a battery of several Enigmas (without plugboards), called "scramblers" and in one-to-one correspondence with the nodes of the TURING graph, synchronously stepping through all 26^3 rotor positions. For each edge two scramblers were connected by a cable, and set to start positions differing by the number that corresponded to the label of the edge. Therefore the physical arrangement of the components was an exact model of the graph. The cable had 26 wires, so all choices for the plug connection of a selected letter ($\tilde{\mathrm{E}}$ in Example 1) could be tested in parallel. The cycle conditions corresponded to closed electrical circuits that made the bombe stop. Then the operator noted the actual rotor positions and restarted the bombe with the next set of positions.

Using enough scramblers even all the sixty rotor orders could be tested in parallel, dropping the effective search costs to 26^3, equivalent with a complexity of 14 bits only. A complete run of the bombe took 11 minutes. (Today a simulation on a PC without parallel execution takes about 5 minutes.)

Unfortunately in general the solution was far from unique, so the bombe produced a huge number of "false positive" stops. An idea of WELCHMAN largely reduced the number of false positives by a clever add-on to the bombe, see Section 8 below, and this was crucial for the success of the British cryptanalysts against the Enigma.

8 Example 2

Now we go through an example step by step and produce a complete solution for the ciphertext

```
ZIDPV USABH HEABG RZMOP UWVJD MLPCS PFTSH ISJMR RFSKU KHUAT
SFDNB GWTAN CSZZW HPHNP DDSAX GTRGY OZPKO EAGRG YSGQD KKNIT
DWFZZ INSYH UTSZR KJDVJ JLJIJ MQHCB RINYI
```

Aligning Known Plaintext

We believe the plaintext contains "Oberleutnant zur See" as the rank of the sender, that is we assume a crib near the end of the message, and assume that at most 20 letters follow, containing the name. The scheme

```
      RGYSGQDKKNITDWFZZINSYHUTSZRKJDVJJLJIJMQHCBRINYI
[ 89] xstopxoberleutnantxzurxseex
[ 90]  xstopxoberleutnantxzurx=eex
[ 91]   x=topxoberleutna=txz=rxseex
[ 92]    xstopxoberleutnantxzurxseex
[ 93]     xstopxoberleut=antxzurxseex
[ 94]      xstopxoberleutnantxzu=xseex
[ 95]       xstopxoberleutnan=x=urxseex
[ 96]        xstopxoberleutnantxzurxseex
[ 97]         xstopxoberleutnantxzurxseex
[ 98]          xs=opxoberleutnantxzurxseex
[ 99]           xstopxoberle==nantxzurxseex
[100]            xstopxoberleutnantxzurxseex
[101]             xstopxoberleutnantxzurxseex
[102]              xstopxoberleutnantxzurxseex
[103]               xstopxoberleutnantxzurxseex
[104]                xstopxoberleutnantxzurxseex
[105]                 xstopxoberleutnantxzurxseex
[106]                  xstopxobe=leutnantxzurxseex
[107]                   x=topxoberleutnantxzurxseex
[108]                    xstopxoberleutnantxzurxseex
[109]                     xstopxoberleutnantxzurxseex
      RGYSGQDKKNITDWFZZINSYHUTSZRKJDVJJLJIJMQHCBRINYI
```

gives 12 hits for the negative pattern search among the 21 considered positions: 89, 92, 96, 97, 100, 101, 102, 103, 104, 105, 108, 109—at least a slight reduction for manual cryptanalysis.

Constructing a TURING Graph

Somewhere along the way we test position 103 and consider the crib

```
FZZINSYHUTSZRKJDVJJLJIJMQHC
xstopxoberleutnantxzurxseex
```

We derive the cycle diagram in Figure 7.

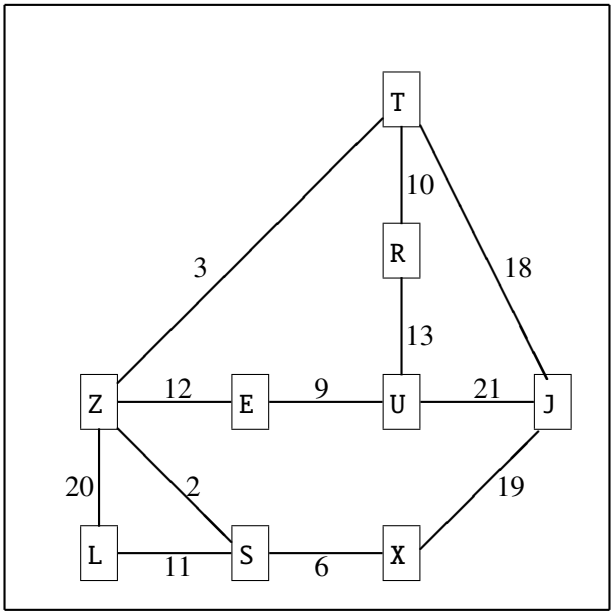

Abbildung 7: TURING *cycles for Example 2*

Therefore as "menu"—the chunk of known plaintext to be examined—we use the sequence of length 20 starting from position 104 (that corresponds to the edge with label 2):

```
ZZINSYHUTSZRKJDVJJLJ
STOPXOBERLEUTNANTXZU
```

To exhaust all the rotor orders, starting positions, and plug connections for this chunk of known plaintext we use Jean-François Bouchaudy's TURING Bombe Simulator, to be found at `http://cryptocellar.web.cern.ch/cryptocellar/simula/jfb/BP12.zip`.

> In a virtual machine on a 2.93 GHz Intel Core-i7 processor it needed 5 minutes for all 60 rotor orders and produced exactly one solution in "WELCHMAN mode" (the diagonal board, see later).

Using only the rotors I, II, and III and disabling the diagonal board—that we haven't introduced yet—we get 6 "solutions" in a few seconds

```
(1)                I II III KFX
(2)                I II III WHV
(3)                II I III ZYN
(4)                III I II JXS
```

```
(5)                 III II I IES
(6)                 III II I QSV
```

Exploring Solution (1)

Let us try the first proposed solution. We begin by decrypting the ciphertext with a ring setting that causes no stepping of the middle rotor for the next 20 positions, and no plugs in the plugboard. Missing plugs will be detected by the following considerations.

> The assumption on the ring setting is somewhat optimistic. It it fails for all of the solutions, we have to try harder, experimenting with shorter cribs or guessing the ring setting of the fast rotor.

We use the rotor order I (slow), II (middle), III (fast), and the start positions `KFX`. This gives the trial decryption

```
ZZINSYHUTSZRKJDVJJLJIJMQHCBRINYI
XPMEJJXPGQBMIVVUKRSISPTNFVAZEQTG
```

This doesn't look like plaintext, but we have not yet explored the plugs. We start with the plug connection $\tilde{\mathtt{Z}}$ of `Z`, the letter with the maximum number of edges in the graph. We try all 26 possible connections, see Table 1

Only line `X` is compatible with the cycle, giving $\tilde{\mathtt{Z}} = \mathtt{X}$. For a manual check of the other cycles we use the complete description of the combined rotor substitutions $\varphi_2, \ldots, \varphi_{21}$ given in Table 2. The "plugged" cycles fit "unplugged" ones:

$$\tilde{\mathtt{Z}} \xrightarrow{3} \tilde{\mathtt{T}} \xrightarrow{10} \tilde{\mathtt{R}} \xrightarrow{13} \tilde{\mathtt{U}} \xrightarrow{9} \tilde{\mathtt{E}} \xrightarrow{12} \tilde{\mathtt{Z}} \quad \text{fits} \quad \mathtt{X} \xrightarrow{3} \mathtt{I} \xrightarrow{10} \mathtt{Y} \xrightarrow{13} \mathtt{F} \xrightarrow{9} \mathtt{L} \xrightarrow{12} \mathtt{X}$$

$$\tilde{\mathtt{Z}} \xrightarrow{2} \tilde{\mathtt{S}} \xrightarrow{6} \tilde{\mathtt{X}} \xrightarrow{19} \tilde{\mathtt{J}} \xrightarrow{21} \tilde{\mathtt{U}} \xrightarrow{9} \tilde{\mathtt{E}} \xrightarrow{12} \tilde{\mathtt{Z}} \quad \text{fits}$$

$$\mathtt{X} \xrightarrow{2} \mathtt{Z} \xrightarrow{6} \mathtt{F} \xrightarrow{19} \mathtt{N} \xrightarrow{21} \mathtt{F} \xrightarrow{9} \mathtt{L} \xrightarrow{12} \mathtt{X}$$

$$\tilde{\mathtt{T}} \xrightarrow{10} \tilde{\mathtt{R}} \xrightarrow{13} \tilde{\mathtt{U}} \xrightarrow{21} \tilde{\mathtt{J}} \xrightarrow{18} \tilde{\mathtt{T}} \quad \text{fits} \quad \mathtt{I} \xrightarrow{10} \mathtt{Y} \xrightarrow{13} \mathtt{F} \xrightarrow{21} \mathtt{N} \xrightarrow{18} \mathtt{I}$$

Therefore the cycle conditions hold indeed.

However we didn't need to check them because reading off the plug connections from the first loop, row "X" in Table 1, we get $\tilde{\mathtt{Z}} = \mathtt{X}$, $\tilde{\mathtt{S}} = \mathtt{Z}$, and this already is a contradiction.

Therefore solution (1) was a false alarm. This observation leads to WELCHMAN's **plug condition** expressing the fact that the plug substitution is an involution:

> *If $\tilde{a} = b$, then also $\tilde{b} = a$ for each pair of letters $a, b \in \Sigma$.*

Exploring Solution (2)

We try the second proposed solution. As before we begin by decrypting the ciphertext, starting from position 103, rotor order I, II, III. Because `V` is the turnover point of Rotor III we have to turn Rotor II back by one position in order to get the correct start positions `WGV`. The trial decryption gives

$\tilde{\mathtt{Z}}$	$\xrightarrow{2}$	$\tilde{\mathtt{S}}$	$\xrightarrow{11}$	$\tilde{\mathtt{L}}$	$\xrightarrow{20}$	$\tilde{\mathtt{Z}}$
A		C		V		W
B		L		H		G
C		A		M		B
D		F		N		R
E		G		K		U
F		D		Z		E
G		E		T		A
H		O		R		N
I		V		C		P
J		M		A		T
K		U		W		V
L		B		I		F
M		J		P		C
N		S		Q		J
O		H		L		S
P		R		O		Y
Q		Y		X		D
R		P		J		Q
S		N		F		I
T		W		U		K
U		K		G		H
V		I		B		M
W		T		E		Z
X		Z		D		X
Y		Q		S		L
Z		X		Y		O

Tabelle 1: Example 2—Possible plug connections for the first cycle

Substition in rotor position	Substitution table 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
φ_2: KFX	C L A F G D E O V M U B J S H R Y P N W K I T Z Q X
φ_3: KFY	D C B A Y S L J X H O G N M K Z R Q F V W T U I E P
φ_4: KFZ	N X E F C D P S M Q U Y I A V G J T H R K O Z B L W
φ_5: KFA	B A X V N Y K Q O Z G M L E I U H T W R P D S C F J
φ_5: KFB	U D L B M Z O Y V S T C E Q G W N X J K A I P R H F
φ_5: KFC	Z U O T X H L F P Y Q G V S C I K W N D B M R E J A
φ_5: KFD	J D U B Y Q R X S A T P O Z M L F G I K C W V H E N
φ_5: KFE	R C B W H L O E J I M F K S G U T A N Q P X D V Z Y
φ_{10}: KFF	M Z H X W P T C Y R O U A Q K F N J V G L S E D I B
φ_{11}: KFG	M I V Z T N K L B P G H A F R J S O Q E W C U Y X D
φ_{12}: KFH	F Z R W V A T I H Y O X N M K U S C Q G P E D L J B
φ_{13}: KFI	J S U G W Y D K L A H I R P Q N O M B V C T E Z F X
φ_{14}: KFJ	V Y O W F E H G K S I P T R C L U N J M Q A D Z B X
φ_{15}: KFK	F R W K Y A M P X V D N G L Q H O B U Z S J C I E T
φ_{16}: KFL	B A I V J S H G C E Q O N M L T K U F P R D Z Y X W
φ_{17}: KFM	R J I O K Y M X C B E P G Q D L N A Z W V U T H F S
φ_{18}: KFN	R Q S P U H L F N K J G T I Z D B A C M E W V Y Z O
φ_{19}: KFO	W V E K C N X Z O R D Y P F I M S J Q U T B A G L H
φ_{20}: KFP	T M P X Z I H G F Q U S B R Y C J N L A K W V D O E
φ_{21}: KFQ	C T A V M N Y Z J I Q O E F L X K W U B S D R P G H

Tabelle 2: Example 2—Combined rotor substitutions for rotor order I, II, III without turnover of Rotor II. Calculated using the online Enigma simulation at `http://enigmaco.de/`.

```
ZZINSYHUTSZRKJDVJJLJIJMQHCBRINYI
STOPXOBERLEUTNANTXZURXSEEXJAEGER
```

—a perfect result. We see that indeed V is the true turnover point of Rotor III, that means that the ring setting of this rotor is A. Moreover all letters except F and W occur, proving that they are unplugged, and the only possible plug connection could be between F and W.

From position 103 we go back for 26 positions and start with the rotor setting WFV. We get

```
RGYOZPKOEAGRGYSGQDKKNITDWF
ISTXLEUCHTTONNEXKNULLNEUNX
```

This proves that also F and W are unplugged. The only key element yet unknown is the ring setting of rotor II.

We go back for another 26 letters and start with the rotor positions WEV. This gives the trial decryption

```
FDNBGWTANCSZZWHPHNPDDSAXGT
SHKTDFEEFXMAMPPGAGRJIXKMXN
```

and the end rotor positions XFV instead of WFV. Something must have happened in between, and this could only be the stepping of Rotor I. The position of Rotor II then must have been E. Because of the double stepping of Rotor II the rotor start positions for this section of text must be VDV. Let's try this:

```
FDNBGWTANCSZZWHPHNPDDSAXGT
XHDREIZEHNXSTOPXERLOSCHENX
```

This is correct plaintext and proves that Rotor II has turnover point E, corresponding to ring setting A.

We conclude that the rotor start positions for the complete text are VCW, and get the decryption

```
ZIDPVUSABHHEABGRZMOPUWVJDMLPCSPFTSHISJMRRFSKUKHUATS
MELDUNGXVONXFREGATTEXGERMANIAXSTOPXPLANQUADRATXQELF

FDNBGWTANCSZZWHPHNPDDSAXGTRGYOZPKOEAGRGYSGQDKKNITDWF
XHDREIZEHNXSTOPXERLOSCHENXISTXLEUCHTTONNEXKNULLNEUNX

ZZINSYHUTSZRKJDVJJLJIJMQHCBRINYI
STOPXOBERLEUTNANTXZURXSEEXJAEGER
```

or, written in a more readable form,

> Meldung X von X Fregatte X Germania X Stop X Planquadrat X Qelf X Hdreizehn X Stop X Erloschen X ist X Leuchttonne X Knullneun X Stop X Oberleutnant X zur X See X Jaeger

A Note on the Technical Realization: WELCHMAN's Diagonal Board

To systematically explore WELCHMAN's plug conditions we consider the connected component of the TURING graph that we used. Assume it consists of the set $M = \{s_1, \ldots, s_r\}$ of letters. When the bombe stops it also provides the plug connection of the selected letter, say s_1 with $\tilde{s}_1$, and allows to derive the set of plug connections $\tilde{M} = \{\tilde{s}_1, \ldots, \tilde{s}_r\}$.

For the false "solution" (1) we had $M = \{\mathtt{E,J,L,R,S,T,U,X,Z}\}$, and the provided or derived plug connections

$$\tilde{\mathtt{E}} = \mathtt{L},\ \tilde{\mathtt{J}} = \mathtt{N},\ \tilde{\mathtt{L}} = \mathtt{D},\ \tilde{\mathtt{R}} = \mathtt{Y},\ \tilde{\mathtt{S}} = \mathtt{Z},\ \tilde{\mathtt{T}} = \mathtt{I},\ \tilde{\mathtt{U}} = \mathtt{F},\ \tilde{\mathtt{X}} = \mathtt{F},\ \tilde{\mathtt{Z}} = \mathtt{X}.$$

We observe two kinds of contradictions:

1. $\tilde{\mathtt{U}} = \mathtt{F}$, $\tilde{\mathtt{X}} = \mathtt{F}$: Two letters in M cannot be connected to the same letter in $\tilde{M}$.
2. $\tilde{\mathtt{E}} = \mathtt{L}$, $\tilde{\mathtt{L}} = \mathtt{D}$, hence $\eta\mathtt{E} = \tilde{\mathtt{E}} \in M \cap \tilde{M}$ and $\eta^2\mathtt{E} \neq \mathtt{E}$. In the same way $\tilde{\mathtt{S}} = \mathtt{Z}$, $\tilde{\mathtt{Z}} = \mathtt{X}$, $\eta^2\mathtt{S} \neq \mathtt{S}$, and $\tilde{\mathtt{Z}} = \mathtt{X}$, $\tilde{\mathtt{X}} = \mathtt{F}$, $\eta^2\mathtt{Z} \neq \mathtt{Z}$.

Checking for these contradictions in software is easy. WELCHMAN's ingenious idea was to imagine and construct a simple device, the diagonal board, that was attached to the bombe and prevented stops in situations that contained contradictions to the plug conditions.

The improved bombe, called TURING-WELCHMAN Bombe, provided only very few false positives. Moreover it not only used the letters in the cycles but also "non-cycle" letters connected to a cycle, in other words, a complete connected component of the TURING graph. In fact it even worked when the graph didn't have any cycles.

9 Example 3

Since Example 2 turned out to be quite simple, we analyze one more example. The ciphertext is

```
CZSTQ GJYNF ZYOLR TLXBR YXJCE MONAS XIPHU CXSAD BGEEQ ROBPI
QMUDP LWYDD GRCMC MJLGW TWBDK BHCPM UMEIB TMCUR DOVPU XNGBZ
QRBKD RPCKL XQKYM CSLGP NHIGD LOHBM PYPNV MTZVU EBDCZ AZLSX
OSZHL GSSZN MBBWS FDTUW IAXEH HLQGR LXMVA MXLWF QGOOA RZXUH
VUAWM KQDXH ZOIJI AMXCI TQNUM ZTZIW CKSBH HRZBH HRNZE WZCGV
BQ
```

and we are quite sure that the plaintext begins with “Befehl X des X Fuehrers X Stop X”. We align this with the ciphertext:

```
CZSTQ GJYNF ZYOLR TLXBR YXJCE
BEFEH LXDES XFUEH RERSX STOPX
```

Negative pattern search yields no contradiction. From positions 1 to 20 we derive the TURING graph whose largest connected component is shown in Figure 8. It has three cycles that overlap, two of them of length 2. Running the Bombe Simulator in “TURING mode” for these three cycles yields about $1500 \approx 60 \cdot 26$ solutions, as expected. The (lexicographically) first of them is

Rotor order	I II III
Start position	ZPB

Table 3 describes the transformations $\varphi_2, \ldots, \varphi_{20}$.

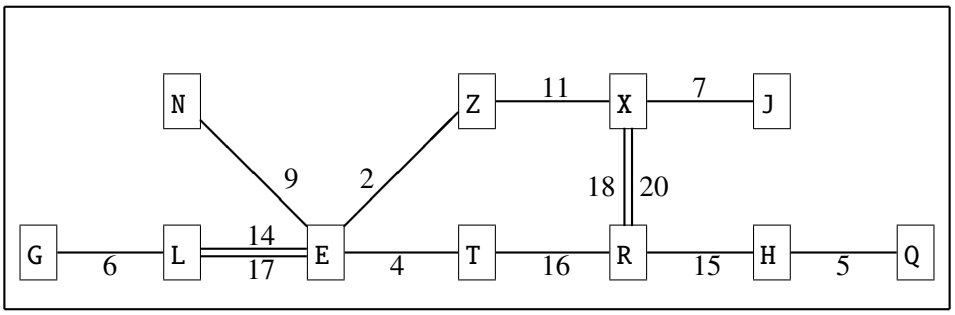

Abbildung 8: TURING *graph for Example 3, largest connected component*

Now we consider the E-L-E cycle and the E-Z-X-R-T-E cycle, see Table 4. The L-E cycle has 6 compatible plug connections for E and L. The E-Z-X-R-T-E cycle boils this number down to 1. The third cycle, X-R-X, fits into the picture, because $\varphi_{20}\tilde{\mathtt{X}} = \varphi_{20}\mathtt{I} = \mathtt{B} = \tilde{\mathtt{R}}$.

Again the WELCHMAN conditions rule out this solution because of the contradiction in the first row: $\tilde{\mathtt{L}} = \mathtt{B}$ in column 2, $\tilde{\mathtt{R}} = \mathtt{B}$ in column 6. And indeed, running the Bombe Simulator in “WELCHMAN mode” yields a unique solution:

Substition in rotor position	Substitution table 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
φ_2: ZPB	N G E S C I B R F W X U O A M Y Z H D V L T J K P Q
φ_3: ZPC	M J S H Q O K D W B G V A U F Z E Y C X N L I T R P
φ_4: ZPD	F L H N I A T C E R X B Y D Z Q P J V G W S U K M O
φ_5: ZPE	V D G B J T C K U E H Y W Z S R X P O F I A M Q L N
φ_6: ZPF	P T I U J Z Q M C E Y S H W X A G V L B D R N O K F
φ_7: ZPG	R D I B M Q U V C Y O T E X K Z F A W L G H S N J P
φ_8: ZPH	Q L F T K C P R Z S E B X W U G A H J D O Y N M V I
φ_9: ZPI	D X J A L Q I S G C U E W R Z V F N H Y K P M B T O
φ_{10}: ZPJ	S W X L R U Q T O M Y D J Z I V G E A H F P B C K N
φ_{11}: ZPK	P E O H B Z Q D N R W Y U I C A G J X V M T K S L F
φ_{12}: ZPL	R M S Y L U T Q P X Z E B V W I H A C G F N O J D K
φ_{13}: ZPM	J P S G Y N D Z Q A T U V F X B I W C K L M R O E H
φ_{14}: ZPN	B A Z W Y R I O G T U X Q V H S M F P J K N D L E C
φ_{15}: ZPO	H M S Y O R L A T U P G B X E K W F C I J Z Q N D V
φ_{16}: ZPP	K F D C R B S T U N A P V J Z L X E G H I M Y Q W O
φ_{17}: ZPQ	B A V L Y S U O K M I D J P H N Z X F W G C T R E Q
φ_{18}: ZPR	N I J Q T U M W B C V S G A Y X D Z L E F K H P O R
φ_{19}: ZPS	Q P K R U J Z N L F C I W H T B A D Y O E X M V S G
φ_{20}: ZPT	V I G L Z P C M B N S D H J Y F X U K W R A T Q O E

Tabelle 3: Example 3—Combined rotor substitutions for rotor order I, II, III without turnover of Rotor II. Calculated using the online Enigma simulation at `http://enigmaco.de/`.

$\tilde{\mathtt{E}}$	$\xrightarrow{14}$	$\tilde{\mathtt{L}}$	$\xrightarrow{17}$	$\tilde{\mathtt{E}}$	$\xrightarrow{2}$	$\tilde{\mathtt{Z}}$	$\xrightarrow{11}$	$\tilde{\mathtt{X}}$	$\xrightarrow{18}$	$\tilde{\mathtt{R}}$	$\xrightarrow{16}$	$\tilde{\mathtt{T}}$	$\xrightarrow{4}$	$\tilde{\mathtt{E}}$
A		B		A		N		I		B		F		A
B		A		B		G		Q		D		C		H
C		Z		Q	†									
D		W		T	†									
E		Y		E		C		O		Y		W		U
F		R		X	†									
G		I		K	†									
H		O		H		R		J		C		D		N
I		G		U	†									
J		T		W	†									
K		U		G	†									
L		X		R	†									
M		Q		Z	†									
N		V		C	†									
O		H		O		M		U		F		B		L
P		S		F	†									
Q		M		J	†									
R		F		S	†									
S		P		N	†									
T		J		M	†									
U		K		I	†									
V		N		P	†									
W		D		L	†									
X		L		D	†									
Y		E		Y		P		A		N		J		R
Z		C		V	†									

Tabelle 4: Example 3—Possible plug connections for the first two loops

Rotor order	III II I
Start position	BMX

with the plugs A-Z, C-X, E-V. A trial decryption with these plugs and ring settings AAA shows parts, but not all of the known plaintext:

```
EUEHLXHECXGFEHRERLXZTOPX
 *     * * **      * *
(B)EFEHLXDESXFUEHRERSXSTOPX
```

To get on we use a second connected component of the TURING graph, see Figure 9.

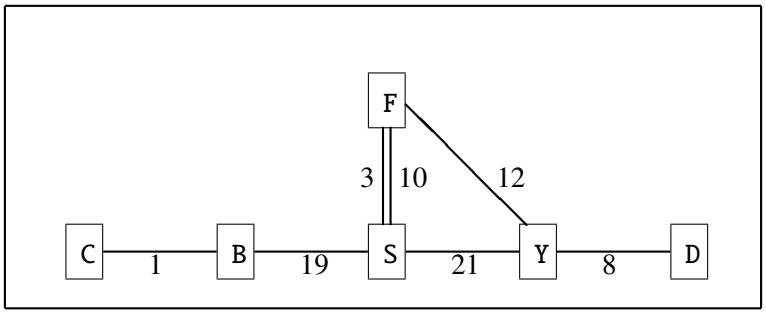

Abbildung 9: TURING *graph for Example 3, second connected component*

Trying the cycle S-F-S with φ_3 and φ_{10} using all the plugs for S that are yet free gives two possible solutions: S-U-S and U-S-U. The second one violates the WELCHMAN condition for S. The first one yields the plugs S-S and F-U. Furthermore we get $\tilde{\mathtt{Y}} = \varphi_{12}\tilde{\mathtt{F}} = \varphi_{12}\mathtt{U} = \mathtt{B}$, and $\tilde{\mathtt{D}} = \varphi_8\tilde{\mathtt{Y}} = \varphi_8\mathtt{B} = \mathtt{W}$.

Up to now we identified the plugs A-Z, B-Y, C-X, D-W, E-V, F-U. Trial decryption yields the perfect plaintext

```
EFEHLXDESXFUEHRERSXSTOPX
```

So we try to decrypt the complete ciphertext with the rotor order III II I, the ring settings AAA, the plugs A-Z, B-Y, C-X, D-W, E-V, F-U, and the start positions BMW, and get

```
BEFEH LXDES XFUEH RERSX STOPX IMXFA LLEXZ XZTXU NWAHR SQEIN
LIQEN XFRAN ZOESI SQENX ANGRI FFSXS INDXD IEXWE STBEF ESTIG
UNGEN XJEDE RXZAH LENMA ESSIG ENXUE BERLE GENHE ITXZU MXTRO
TZXZU XHALT ENXST OPXFU EHRUN GXUND XTRUP PEXMU ESSEN XVONX
DIESE RXEHR ENPFL IQTXD URQDR UNGEN XSEIN XSTOP XHEIL XHITL
ER
```

> Befehl des Fuehrers STOP Im Falle z. Zt. unwahrscheinlichen franzoesischen Angriffs sind die Westbefestigungen jeder zahlenmaessigen Ueberlegenheit zum Trotz zu halten STOP Fuehrung und Truppe muessen von dieser Ehrenpflicht durchdrungen sein STOP Heil Hitler

We observe that the slow rotor didn't step during this decryption. In general the a priori probability for its stepping was 257 letters of text divided by 676 possible positions of the other two rotors ≈ 0.38.

10 Discussion

- TURING's attack against the cycles of the graph also works for non-involutory rotor machines. Then the graph model is a *directed* graph and the attacker has to find directed cycles. These are quite rare, therefore the attack loses most of its power.
- **Exercise.** Find the directed cycles in Figures 5, 7, 8, 9.
- The TURING-WELCHMAN Bombe used the involutary characters of the complete Enigma substitution as well as of the plugboard. The inventors of both of these "features" apparently didn't see the weaknesses.
- Nevertheless the addition of the plugboard made the machine much stronger. The isomorph attack worked by paper and pencil. Attacking the Wehrmacht Enigma only worked with the help of heavy machinery.

Literatur

[1] Bauer F. L. *Decrypted Secrets; Methods and Maxims of Cryptology*. Berlin: Springer 1997.

[2] Deavours C. A., Kruh L. *Machine Cryptography and Modern Cryptanalysis*. Norwood: Artech House 1985.

[3] Ganesan R., Sherman A. T., *Statistical Techniques for Language Recognition: An Introduction and Guide for Cryptanalysts*. Cryptologia 17 (1993), 321–366.

[4] Ganesan R., Sherman A. T., *Statistical Techniques for Language Recognition: An Empirical Study Using Real and Simulated English*. Cryptologia 18 (1994), 289–331.

[5] Kullback S. *Statistical Methods in Cryptanalysis*. Laguna Hills: Aegean Park Press 1976.

[6] Sinkov A. *Elementary Cryptanalysis*. Washington: The Mathematical Association of America 1966.