Aperiodic Polyalphabetic Ciphers

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

January 14, 2000—English version April 27, 2014—last change January 19, 2021

Overview Over Polyalphabetic Ciphers

	Monoalph. Substitution	**Periodic Polyalph. Substitution**	**Aperiodic Polyalph. Substitution**
Standard Alphabet	Shift Cipher (CAESAR)	BELLASO cipher ("Vigenère")	Running-Text Cipher
Non-Standard Alphabet	General Monoalph. Substitution	PORTA's General Polyalph. Cipher	Stream Cipher

The table is not completely exact. The running-text cipher is only a (but the most important) special case of an aperiodic polyalphabetic substitution using the standard alphabet. An analogous statement holds for PORTA's disk cipher and a general periodic polyalphabetic substitution. In contrast by stream cipher we denote an even more general construct.

1 Running-Text Ciphers

Method

Assume we have a plaintext of length r. We could encrypt it with the BELLASO cipher (and the TRITHEMIUS table). But instead of choosing a keyword and periodically repeating this keyword we use a keytext of the same length r as the plaintext. Then we add plaintext and keytext letter for letter (using the table).

The abstract mathematical description uses a group structure on the alphabet Σ with group operation $*$. For a plaintext $a \in M_r = M \cap \Sigma^r$ we choose a key $k \in \Sigma^r$ and calculate

$$c_i = a_i * k_i \quad \text{for } 0 \le i \le r-1.$$

We may interpret this as shift cipher on Σ^r. The formula for decryption is

$$a_i = c_i * k_i^{-1} \quad \text{for } 0 \le i \le r-1.$$

If the key itself is a meaningful text $k \in M_r$ in the plaintext language, say a section from a book, then we call this a **running-text cipher**.

Example

Equip $\Sigma = \{\mathtt{A}, \ldots, \mathtt{Z}\}$ with the group structure as additive group of integers mod 26.

```
Plaintext:  i a r r i v e t o m o r r o w a t t e n o c l o c k
Keytext:    I F Y O U C A N K E E P Y O U R H E A D W H E N A L
            ---------------------------------------------------
Ciphertext: Q F P F C X E G Y Q S G P C Q R A X E Q K J P B C V
```

A Perl program is `runkey.pl` in the web directory `http://www.staff.uni-mainz.de/pommeren/Cryptology/Classic/Perl/`.

Practical Background

To avoid a period in a polyalphabetic substitution we choose a key that is (at least) as long as the plaintext. On the other hand we need a key that is easily remembered or transferred to a communication partner.

A common method of defining such a key is taking a book and beginning at a certain position. The effective key is the number triple (page, line, letter). This kind of encryption is sometimes called a **book cipher**. Historically the first known reference for this method seems to be

> Arthur Hermann: *Nouveau système de correspondence secrète. Méthode pour chiffrer et déchiffrer les dépêches secrètes.* Paris 1892.

But note that there are also other ways to use a book for encryption, see `http://www.staff.uni-mainz.de/pommeren/Cryptology/Classic/1_Monoalph/Variants.html`.

A modern version could use the contents of a CD beginning with a certain position.

Exercise: How large is the keyspace of this cipher, when the attacker knows which CD was used?

2 Cryptanalytic Approaches to Running-Text Ciphers

Cryptanalysis of running-text ciphers is laborious. There are several approaches that should be combined in practice. Automated procedures are proposed in

> E. Dawson and L. Nielsen: Automated cryptanalysis of XOR plaintext strings. Cryptologia XX (1996), 165–181.
>
> A. Griffing: Solving the running key cipher with the Viterbi algorithm. Cryptologia XXX (2006), 361–367.

The first of these considers running-text ciphers where plaintext and key are combined via binary addition (XOR) instead of addition mod 26. This distinction not essential for the method (but of course for the use of the program).

Approach 0: Exhaustion

Exhaustion of all possible keytexts is practically infeasible when there is no a priori idea what the keytext could be. Exhaustion is feasible when the attacker knows the source of the keytext, say a certain book. If the source text has length q and the ciphertext has length r, then there are only $q - r$ choices for the start of the key text. This is troublesome for the pencil and paper analyst, but easy with machine support.

Approach 1: Probable Word and Zigzag Exhaustion

When in the example above the attacker guesses the probable word "arrive" in the plaintext and shifts it along the ciphertext, already for the second position she gets the keytext `FYOUCA`. With a little imagination she guesses the phrase `IFYOUCAN`, yielding the plaintext fragment `IARRIVET`, and expands this fragment to `IARRIVETOMORROW`. This in turn expands the keytext to `IFYOUCANKEEPYOU`. Proceeding in this way alternating between plaintext and keytext is called **zigzag exhaustion** (or cross-ruff method). For some time during this process it may be unclear whether a partial text belongs to plaintext or key.

A dictionary is a useful tool for this task. Or a pattern search in a collection of literary texts may lead to success.

Approach 2: Frequent Word Fragments

If the attacker cannot guess a probable word she might try common word fragments, bearing in mind that plaintext as well as keytext are meaningful texts. Shifting words or word fragments such as

```
THE AND FOR WAS HIS NOT BUT ARE ING ION ENT
THAT THIS FROM WITH HAVE TION
```

along the ciphertext will result in many meaningful trigrams or tetragrams that provide seed crystals for a zigzag exhaustion. Recognizing typical combinations such as

```
THE + THE = MOI
ING + ING = QAM
THAT + THAT = MOAM
```

may be useful.

Approach 3: Frequency Analysis

Let $p_0, \ldots, p_{n-1}$ be the letter frequencies of the (stochastic) language M over the alphabet $\Sigma = \{s_0, \ldots, s_{n-1}\}$. Then running-key ciphertexts will exhibit the typical letter frequencies

$$q_h = \sum_{i+j=h} p_i \cdot p_j \quad \text{for } 0 \leq h \leq n-1.$$

Even though the distribution is much more flat compared with plain language, it is not completely uniform, and therefore leaks some information on the plaintext. For example it gives a hint at the method of encryption.

Example: Letter frequencies of running-text cryptograms in **English** (values in percent). Coincidence index = 0.0400.

A	B	C	D	E	F	G	H	I	J	K	L	M
4.3	3.5	3.2	2.5	4.7	3.8	4.4	4.4	4.8	2.9	3.5	4.5	4.3
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
3.1	3.2	3.6	3.0	4.4	4.5	4.0	3.2	4.9	4.7	3.8	3.3	3.5

Example: Letter frequencies of running-text cryptograms in **German** (values in percent). Coincidence index = 0.0411.

A	B	C	D	E	F	G	H	I	J	K	L	M
4.2	2.6	2.3	2.4	5.0	3.7	3.7	4.3	5.8	2.9	3.7	4.4	4.9
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
3.2	3.0	3.1	3.3	5.7	3.4	3.2	3.4	5.9	4.5	3.9	3.9	3.6

Even more helpful is the distribution of bigrams and trigrams. Each bigram in the ciphertext has $26^2 = 676$ different possible sources whose probabilities however show large differences. For trigrams most sources even have probabilities 0.

A systematic description of this approach is in

> Craig Bauer and Christian N. S. Tate: A statistical attack on the running key cipher. Cryptologia XXVI (2002), 274–282.

Approach 4: Frequent Letter Combinations

Frequency analysis (approach 3) is cumbersome, at least for manual evaluation. Friedman refined this approach in a systematic way that doesn't need known plaintext. See the next section.

3 Cryptanalysis According to Friedman

Friedman proposed a systematic approach to solving running-key ciphers in the article

> W. F. Friedman: *Methods for the Solution of Running-Key Ciphers.* Riverbank Publication No. 16 (1918). In: The Riverbank Publications Vol 1, Aegean Park Press 1979.

Consider a running-text cryptogram. Friedman's method starts from the observation that a significant fraction of the ciphertext letters arise from a combination of two frequent plaintext letters.

The frequency distribution (in percent) of the nine most frequent German letters is:

E	N	I	R	S	A	T	D	U
18.0	10.6	8.1	7.2	6.9	6.0	5.7	5.4	4.6

Therefore these letters account for 72.5% of a German text.

Assuming that the key is sufficiently independent of the plaintext we expect that about 53% ciphertext letters arose from a combination of two of these letters in plaintext or key. This fact is not overly impressive. In the example

```
                                  |   | | |     |   |
  Plaintext:  i c h k o m m e m o r g e n u m z e h n
  Key:        V O M E I S E B E F R E I T S I N D S T
              ---------------------------------------
  Ciphertext: D Q T O W E Q F Q T I K M G M U M H Z G
```

this applies only to 6 of 20 letters. The method won't work well for this example.

Let us take another example (from the football world championships 2002):

```
            | | | | |       | |         | |   |       |   |   |
Plaintext:  d e u t s c h l a n d b e s i e g t p a r a g u a y
Key:        E I N E N A T U E R L I C H E S P R A C H E H A T T
            ---------------------------------------------------
Ciphertext: H M H X F C A F E E O J G Z M W V L P C Y E N U T R
```

Here we see 13 of 26 letters as interesting. We use this example to explain the method.

Let's begin with the first four letters, and consider all combinations that lead to them

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

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

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

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

The most probable pairs are flagged. We condense this observation:

```
DENU EISTU DENU DETU
EDUN IEUTS EDUN UTED
H    M     H    X
```

There is a total of $4 \cdot 5 \cdot 4 \cdot 4 = 320$ possible combinations of these pairs. Some of them may be eliminated immediately, for example we may exclude that plaintext or key begin with the letters `DS`.

If we start with the pair `D-E` we might continue with `E-I` or `U-S`. The first case has only one meaningful continuation:

```
DEUT
EINE
```

The second case could proceed with `D-E`, but no fourth pair fits. A possible pair number 3 is `N-U` also, and then `E-T` or `T-E` fit as pair number 4. Therefore we note two more options, both of them not really convincing:

```
DEUT  DUNE  DUNT
EINE  ESUT  ESUE
```

Starting with `E-D` we find an exactly symmetric situation and get the same three options but with plaintext and key interchanged.

Starting with `N-U` we might continue with `I-E` or `U-S`. The first case has `E-D` as only plausible continuation, and then `T-E`:

```
DEUT  DUNE  DUNT  NIET
EINE  ESUT  ESUE  UEDE
```

The second case could proceed with `D-E` (and then `E-T`) or `N-U` (and then there is no good continuation). So we found one more option:

```
DEUT  DUNE  DUNT  NIET  NUDE
EINE  ESUT  ESUE  UEDE  USET
```

Taking all the symmetric ones into account we face a total of 10 somewhat plausible options—*under the assumption that the first four letters of plaintext and key belong to the nine most frequent German letters.*

Of our five (+ five symmetric) options the first looks best. But also the fourth is reasonably good, bearing in mind that the keytext might begin in the middle of a word (for example “mï¿½de” = `(M)UEDE`). In any case let’s begin with the first option that looks somewhat better. It suggests the continuation `SCH`. This seems promising:

```
DEUTSCH
EINENAT
```

Of course if this fails we would also try for example `DEUTLICH` or `DEUTEN`.

As next letter in the first row we would try `E` or `L` and note that `L` gives a better continuation in the second row (`U` better than `B`). Therefore the begin `DEUTSCHLAND` is decrypted—but we don’t yet know whether it is plaintext or key. From this point we struggle ahead in zigzag as noted before.

4 Other Applications of Running-Text Analysis

Key Re-Use

Consider an alphabet Σ with a group structure, and consider an (aperiodic or periodic) polyalphabetic cipher that uses the CAESAR operation: For a plaintext $a = (a_0, a_1, a_2, \ldots)$ and a keystream $k = (k_0, k_1, k_2, \ldots)$ the ciphertext $c = (c_0, c_1, c_2, \ldots)$ is given by

$$c_i = a_i * k_i \quad \text{for } i = 0, 1, 2, \ldots.$$

Because the key is not necessarily meaningful text the cryptanalytic methods for running-text ciphers don't apply.

But suppose another plaintext $b = (b_0, b_1, b_2, \ldots)$ is encrypted with the *same* key k, resulting in the ciphertext $d = (d_0, d_1, d_2, \ldots)$,

$$d_i = b_i * k_i \quad \text{for } i = 0, 1, 2, \ldots.$$

The attacker recognizes this situation by coincidence analysis.

Then the difference (or quotient, depending on the notation of the group law) is given by

$$d_i * c_i^{-1} = b_i * k_i * k_i^{-1} * a_i^{-1} = b_i * a_i^{-1} \quad \text{for } i = 0, 1, 2, \ldots.$$

In this way the attacker who knows the ciphertexts c and d finds the difference $b_i * a_i^{-1}$ that is the composition of two meaningful texts she doesn't know but wants to. She therefore applies the methods for running-text encryption and eventually finds a and b and then even k.

Historical Notes

This kind of analysis was a main occupation of the cryptanalysts in World War II and in the following Cold War. In particular teleprinter communication used additive stream ciphers (mostly XOR) with keystreams from key generators and very long periods. In case of heavy message traffic often passages of different messages were encrypted with the key generator in the same state. Searching such passages was called "in-depth-analysis" and relied on coincidence calculations. Then the second step was to subtract the identified passages and to apply running-text analysis.

Some known examples for this are:

- Breaking the Lorenz cipher teleprinter SZ42 ("Schlüsselzusatz") by the British cryptanalysts at Bletchley Park in World War II (project "Tunny").
- Breaking HAGELIN's B21 in 1931 and the Siemens-Geheimschreiber T52 in 1940 by the Swedish mathematician Arne BEURLING. The T52 was also partially broken at Bletchley Park (project "Sturgeon").

- The latest politically relevant application of this cryptanalytic technique occurred in the 1950es. US cryptanalysts broke Sovjet ciphertexts and by the way debunked the spy couple Ethel und Julius Rosenberg (project “Venona”). The Sovjet spys used a one-time pad—in principle. But because key material was rare keys were partly reused.

Large Periods

Another application is the Trithemius-Belaso cipher with a large period l, large enough that the standard procedure of arranging the ciphertext in columns and shifting the alphabets fails.

Then the attacker may consider the ciphertext shifted by l positions and subtract it from the original ciphertext:

$$c_{i+l} - c_i = a_{i+l} - a_i.$$

Or, if the key consists of meaningful text, directly treat the cipher as a running-text cipher.

Exercise.

```
BOEKV HWXRW VMSIB UXBRK HYQLR OYFWR KODHR JQUMM SJIQA THWSK
CRUBJ IELLM QSGEQ GSJFT USEWT VTBPI JMPNH IGUSQ HDXBR ANVIS
VEHJL VJGDS LVFAM YIPJY JM
```

Hints.

- Find evidence for a period of 38 or 76.
- Try the probable word `AMERICA` as part of the key.

5 Random Keys

All cryptanalytic methods collapse when the key is a random letter sequence, chosen in an independent way for each plaintext, and never repeated. In particular all the letters in the ciphertexts occur with the same probability. Or in other words, the distribution of the ciphertext letters is completely flat.

This encryption method is called **One-Time Pad** (OTP). Usually Gilbert VERNAM (1890–1960) is considered as the inventor in the World War II year 1917. But the idea of a *random* key is due to MAUBORGNE who improved VERNAM's periodic XOR cipher in this way. The German cryptologists KUNZE, SCHAUFFLER, and LANGLOTZ in 1921—presumably independently from MAUBORGNE—proposed the "individuellen Schlüssel" ("individual key") for running-text encryption of texts over the alphabet $\{\texttt{A}, \ldots, \texttt{Z}\}$.

In other words: The idea "was in the air". In 2011 Steve Bellovin discovered a much earlier proposal of the method by one Frank MILLER in 1882 who however was completely unknown as a crypologist and didn't have any influence on the history of cryptography.

> Steven M. Bellovin. *Frank Miller: Inventor of the One-Time Pad.* Cryptologia 35 (2011), 203–222.

Uniformly Distributed Random Variables in Groups

This subsection contains evidence for the security of using random keys. The general idea is:

> "Something + Random = Random" or "Chaos Beats Order"
> (the Cildren's Room Theorem)

We use the language of Measure Theory.

Theorem 1 *Let G be a group with a finite, translation invariant measure μ and Ω, a probability space. Let $X, Y : \Omega \longrightarrow G$ be random variables, X uniformly distributed, and X, Y independent. Let $Z = X * Y$ (where $*$ is the group law of composition). Then:*

(i) *Z is uniformly distributed.*

(ii) *Y and Z are independent.*

Comment The independency of X and Y means that

$$P(X^{-1}A \cap Y^{-1}B) = P(X^{-1}A) \cdot P(Y^{-1}B) \quad \text{for all measurable } A, B \subseteq G.$$

The uniform distribution of X means that

$$P(X^{-1}A) = \frac{\mu(A)}{\mu(G)} \quad \text{for all measurable } A \subseteq G.$$

In particular the measure P_X on G defined by $P_X(A) = P(X^{-1}A)$ is translation invariant, if μ is so.

Remark Z is a random variable because $Z = m^{-1} \circ (X, Y)$ with $m = *$, the group law of composition. This is measurable because its g-sections,

$$(m^{-1}A)_g = \{h \in G \mid gh \in A\}$$

are all measurable, and the function

$$g \mapsto \mu(m^{-1}A)_g = \mu(g^{-1}A) = \mu(A)$$

is also measurable. A weak form of FUBINI's theorem gives that $m^{-1}A \subseteq G \times G$ is measurable, and

$$(\mu \otimes \mu)(m^{-1}A) = \int_G (m^{-1}A)_g \, dg = \mu(A) \int_G dg = \mu(A)\mu(G).$$

Counterexamples We analyze whether the conditions of the theorem can be weakened.

1. What if we don't assume X is uniformly distributed? As an example take $X = \mathbf{1}$ (unity element of group) constant and Y arbitrary; then X and Y are independent, but $Z = Y$ in general is not uniformly distributed nor independent from Y.
2. What if we don't assume X and Y are independent? As an example take $Y = X^{-1}$ (the group inverse); the product $Z = \mathbf{1}$ in general is not uniformly distributed. Choosing $Y = X$ we get $Z = X^2$ that in general is not uniformly distributed nor independent from Y. (More concrete example: $\Omega = G = \mathbb{Z}/4\mathbb{Z}, X =$ identity map, $Z =$ squaring map.)

General proof of the Theorem

(For an elementary proof of a practically relevant special case see below.)

Consider the product map

$$(X, Y) \colon \Omega \longrightarrow G \times G$$

and the extended composition

$$\sigma \colon G \times G \longrightarrow G \times G, \quad (g, h) \mapsto (g * h, h).$$

For $A, B \subseteq G$ we have (by definition of the product probability)

$$(P_X \otimes P_Y)(A \times B) = P_X(A) \cdot P_Y(B) = P(X^{-1}A) \cdot P(Y^{-1}B);$$

because X and Y are independent we may continue this equation:

$$\begin{aligned} &= P(X^{-1}A \cap Y^{-1}B) = P\{\omega \mid X\omega \in A, Y\omega \in B\} \\ &= P((X,Y)^{-1}(A \times B)) = P_{(X,Y)}(A \times B). \end{aligned}$$

Therefore $P_{(X,Y)} = P_X \otimes P_Y$, and for $S \subseteq G \times G$ we apply Fubini's theorem:

$$P_{(X,Y)}(S) = \int_{h \in G} P_X(S_h) \cdot P_Y(dh).$$

Especially for $S = \sigma^{-1}(A \times B)$ we get

$$\begin{aligned} S_h &= \{g \in G \mid (g * h, h) \in A \times B\} = \begin{cases} A * h^{-1}, & \text{if } h \in B, \\ \emptyset & \text{else,} \end{cases} \\ P_X(S_h) &= \begin{cases} P_X(A * h^{-1}) = \frac{\mu(A)}{\mu(G)}, & \text{if } h \in B, \\ 0 & \text{else.} \end{cases} \end{aligned}$$

Therefore

$$\begin{aligned} P(Z^{-1}A \cap Y^{-1}B) &= P\{\omega \in \Omega \mid X(\omega) * Y(\omega) \in A, Y(\omega) \in B\} \\ &= P((X,Y)^{-1}S) = P_{(X,Y)}(S) \\ &= \int_{h \in B} \frac{\mu(A)}{\mu(G)} \cdot P_Y(dh) = \frac{\mu(A)}{\mu(G)} \cdot P(Y^{-1}B). \end{aligned}$$

Setting $B = G$ we conclude $P(Z^{-1}A) = \frac{\mu(A)}{\mu(G)}$, which gives (i), and from this we immediately conclude

$$P(Z^{-1}A \cap Y^{-1}B) = P(Z^{-1}A) \cdot P(Y^{-1}B)$$

which proves also (ii). $\diamond$

Proof for countable groups

In the above proof we used general measure theory, but the idea was fairly simple. Therefore we repeat the proof for the countable case, where integrals become sums and the argumentation is elementary. For the cryptographic application the measure spaces are even finite, so this elementary proof is completely adequate.

Lemma 1 *Let G, Ω, X, Y, and Z be as in the theorem. Then*

$$Z^{-1}(A) \cap Y^{-1}(B) = \bigcup_{h \in B} [X^{-1}(A * h^{-1}) \cap Y^{-1}h]$$

for all measurable $A, B \subseteq G$.

The proof follows from the equations

$$\begin{aligned}
Z^{-1}A &= (X,Y)^{-1}\{(g,h) \in G \times G \mid g * h \in A\} \\
&= (X,Y)^{-1}\left[\bigcup_{h \in G} A * h^{-1} \times \{h\}\right] \\
&= \bigcup_{h \in G} (X,Y)^{-1}(A * h^{-1} \times \{h\}) \\
&= \bigcup_{h \in G} [X^{-1}(A * h^{-1}) \cap Y^{-1}h], \\
Z^{-1}A \cap Y^{-1}B &= \bigcup_{h \in G} [X^{-1}(A * h^{-1}) \cap Y^{-1}h \cap Y^{-1}B] \\
&= \bigcup_{h \in B} [X^{-1}(A * h^{-1}) \cap Y^{-1}h].
\end{aligned}$$

Now let G be countable. Then

$$\begin{aligned}
P(Z^{-1}A \cap Y^{-1}B) &= \sum_{h \in B} P[X^{-1}(A * h^{-1}) \cap Y^{-1}h] \\
&= \sum_{h \in B} P[X^{-1}(A * h^{-1})] \cdot P[Y^{-1}h] \quad \text{(because } X, Y \text{ are independent)} \\
&= \sum_{h \in B} \frac{\mu(A * h^{-1})}{\mu(G)} \cdot P[Y^{-1}h] \quad \text{(because } X \text{ is uniformly distributed)} \\
&= \frac{\mu(A)}{\mu(G)} \cdot \sum_{h \in B} P[Y^{-1}h] \\
&= \frac{\mu(A)}{\mu(G)} \cdot P\left[\bigcup_{h \in B} Y^{-1}h\right] \\
&= \frac{\mu(A)}{\mu(G)} \cdot P(Y^{-1}B).
\end{aligned}$$

Setting $B = G$ we get $P(Z^{-1}A) = \frac{\mu(A)}{\mu(G)}$, which gives (i), and immediately conclude

$$P(Z^{-1}A \cap Y^{-1}B) = P(Z^{-1}A) \cdot P(Y^{-1}B),$$

which proves (ii). $\diamond$

Discussion

The theorem says that a One-Time Pad encryption results in a ciphertext that "has nothing to do" with the plaintext, in particular doesn't offer any lever for the cryptanalyst.

Why then isn't the One-Time Pad the universally accepted standard method of encryption?

- Agreeing upon a key is a major problem—if we can securely transmit a key of this length, why not immediately transmit the message over the same secure message channel? Or if the key is agreed upon some time in advance—how to remember it?

- The method is suited at best for a two-party communication. For a multiparty communication the complexity of key distribution becomes prohibitive.

- When the attacker has known plaintext she is not able to draw any conclusions about other parts of the text. But she can exchange the known plaintext with another text she likes more: *The integrity of the message is at risk.*

6 Autokey Ciphers

The first one to propose autokey ciphers was BELLASO in 1564. Also this cipher is often attributed to VIGENÈRE.

Encryption and Decryption

The alphabet Σ is equipped with a group operation $*$. As key chose a string $k \in \Sigma^l$ of length l. For encrypting a plaintext $a \in \Sigma^r$ one concatenates k and a and truncates this string to r letters. This truncated string then serves as keytext for a running-key encryption:

Plaintext:	a_0	a_1	$\ldots$	a_{l-1}	a_l	$\ldots$	a_{r-1}
Keytext:	k_0	k_1	$\ldots$	k_{l-1}	a_0	$\ldots$	a_{r-l-1}
Ciphertext:	c_0	c_1	$\ldots$	c_{l-1}	c_l	$\ldots$	c_{r-1}

The formula for encryption is

$$c_i = \begin{cases} a_i * k_i & \text{for } i = 0, \ldots l-1, \\ a_i * a_{i-l} & \text{for } i = l, \ldots r-1. \end{cases}$$

Example, $\Sigma = \{\mathtt{A}, \ldots, \mathtt{Z}\}$, $l = 2$, $k = \mathtt{XY}$:

```
P L A I N T E X T
X Y P L A I N T E
-----------------
M J P T N B R Q X
```

Remark: Instead of the standard alphabet (or the TRITHEMIUS table) one could also use a permuted primary alphabet.

Here is the formula for decryption

$$a_i = \begin{cases} c_i * k_i^{-1} & \text{for } i = 0, \ldots l-1, \\ c_i * a_{i-l}^{-1} & \text{for } i = l, \ldots r-1. \end{cases}$$

A Perl program is `autokey.pl` in the web directory `http://www.staff.uni-mainz.de/pommeren/Cryptology/Classic/Perl/`.

Approaches to Cryptanalysis

The four most promising approaches are:

- Exhaustion for small l.
- Interpretation as running-key cipher from position l, in case of a key word or phrase from the plaintext language even from the beginning of the ciphertext:

  - Probable word and zigzag exhaustion
  - Frequent word fragments
  - Frequency analysis
  - Frequent letter combinations

  The repetition of the plaintext in the key makes the task considerably easier.

- Similarity with the TRITHEMIUS-BELLASO cipher, see Section 8 below
- Algebraic cryptanalysis (for known plaintext): Solving equations. We describe this for a commutative group, the group operation written as addition, that is, we consider Σ, Σ^r, and Σ^{r+l} as $\mathbb{Z}$-modules.

We interpret the encryption formula as a system of linear equations with an $r \times (r+l)$ coefficient matrix:

$$\begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{l-1} \\ c_l \\ \vdots \\ c_{r-1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 1 & & & & \\ & 1 & 0 & \dots & 1 & & & \\ & & \ddots & \ddots & & & \ddots & \\ & & & 1 & 0 & \dots & & 1 \end{pmatrix} \begin{pmatrix} k_0 \\ k_1 \\ \vdots \\ k_{l-1} \\ a_0 \\ \vdots \\ a_{r-1} \end{pmatrix}$$

This is a system of r linear equations with the $r+l$ unknowns (the components of) $k \in \Sigma^l$ and $a \in \Sigma^r$. "In general" such a system is solvable as soon as l of the unknowns are guessed, that means known plaintext of length l (not necessarily connected). Since the involved $\mathbb{Z}$-modules are (in most interesting cases) not vector spaces, solving linear equations is a bit intricate but feasible. This is comprehensively treated in the next chapter.

Ciphertext Autokey

Using ciphertext instead of plaintext as extension of the l-letter key is a useless variant, but also proposed by VIGENÈRE. We only describe it by an example:

Example, $\Sigma = \{\texttt{A}, \dots, \texttt{Z}\}$, $l = 2$, $k = \texttt{XY}$:

```
P L A I N T E X T
X Y M J M R Z K D
-----------------
M J M R Z K D H W
```

Exercise. Give a formal description of this cipher. Why is cryptanalysis almost trivial? Work out an algorithm for cryptanalysis.

Exercise. Apply your algorithm to the cryptogram

```
IHTYE VNQEW KOGIV MZVPM WRIXD OSDIX FKJRM HZBVR TLKMS FEUKE
VSIVK GZNUX KMWEP OQEDV RARBX NUJJX BTMQB ZT
```

Remark: Using a nonstandard alphabet makes this cipher a bit stronger.

7 Example: Cryptanalysis of an Autokey Cipher

The Cryptogram

Suppose we got the ciphertext

```
LUSIT FSATM TZJIZ SYDZM PMFIZ REWLR ZEKLS RQXCA TFENE YBVOI
WAHIE LLXFK VXOKZ OVQIP TAUNX ARZCX IZYHQ LNSYM FWUEQ TELFH
QTELQ IAXXV ZPYTL LGAVP ARTKL IPTXX CIHYE UQR
```

The context suggests that the plaintext language is French.

Here are some statistics. The letter count

A	B	C	D	E	F	G	H	I	J	K	L	M
8	1	3	1	9	6	1	4	10	1	4	11	4
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
3	3	5	7	6	5	10	4	5	3	9	6	9

as well as the coincidence index 0.0437 suggest a polyalphabetic cipher, the autocincidence spectrum shows no meaningful period. The frequency distribution of the single letters hints at a running-key or autokey cipher that uses the standard alphabet (= Trithemius table).

A Probable Word

Since the message probably originated from the french embassy at Berlin in 1870 we may assume that the plaintext contains the word “allemand”. Moving this word along the ciphertext and subtracting the probable word—with the help of the Perl script `probwd.pl`—we get 4 good matches (plus some weak ones):

```
000: LJHEHFFX     015: SNSVAPZC     030: ZTZHGRDU
001: UHXPTSNQ     016: YSOIDMSF     031: EZAOFQKZ
002: SXIBGAGJ     017: DOBLAFVW     032: KAHNEXPX
003: IIUOOTZQ     018: ZBEITIMO <-- 033: LHGMLCNQ
004: TUHWHMGW     019: MEBBWZEB     034: SGFTQAGC
005: FHPPATMG     020: PBUENRRT     035: RFMYOTSB
006: SPIIHZWF     021: MUXVFEJI     036: QMRWHFRK
007: AIBPNJVW     022: FXONSWYO     037: XRPPTEAB
008: TBIVXIMP     023: IOGAKLEW     038: CPIBSNRV
009: MIOFWZFV     024: ZGTSZRMB     039: AIUABELY <--
010: TOYENSLA <== 025: RTLHFZRH     040: TUTJSYOS
011: ZYXVGYQW     026: ELANNEXI <== 041: FTCAMBIL <--
012: JXOOMDMJ     027: WAGVSKYP     042: ECTUPVBF
013: IOHURZZM     028: LGOAYLFO     043: NTNXJOVT
014: ZHNZNMCJ     029: ROTGZSEN     044: ENQRCIJX
```

```
045: YQKKWWNE     060: VMDGNOIN     075: AGOYLIMV <--
046: BKDEKAUF     061: XDZVCVDF     076: RORTWZLE
047: VDXSOHVB     062: OZOKJQVM     077: ZRMENYUN
048: OXLWVIRI     063: KODREICQ     078: CMXVMHDI
049: ILPDWEYI     064: ZDKMWPGX     079: XXOUVQYK
050: WPWESLYU     065: OKFEDTNR     080: IONDELAP <==
051: AWXAZLKC     066: VFXLHAHK     081: ZNWMZNFV
052: HXTHZXSH     067: QXEPOUAU     082: YWFHBSLJ
053: ITAHLFXS     068: IEIWINKX     083: HFAJGYZC
054: EAATTKIU     069: PIPQBXNO     084: QACOMMST
055: LAMBYVKL     070: TPJJLAEW     085: LCHUAFJR
056: LMUGJXBH     071: AJCTORMZ     086: NHNITWHB
057: XUZRLOXW     072: UCMWFZPU     087: SNBBKURN
058: FZKTCKML     073: NMPNNCKF     088: YBUSIEDQ
059: KKMKYZBS     074: XPGVQXVW     089: MULQSQGB

090: FLJAETRI     105: IPMTJZCV     120: AGIGZICQ
091: WJTMHEYC     106: AMMRNPLQ     121: RIZHWPGU
092: UTFPSLSE     107: XMKVDYGI     122: TZAEDTKU
093: EFIAZFUN     108: XKOLMTYI     123: KAXLHXKZ
094: QITHTHDQ     109: VOEUHLYD     124: LXEPLXPF
095: TTABVQGB     110: ZENPZLTX     125: IEITLCVE
096: EAUDETRI <== 111: PNIHZGNS     126: PIMTQIUV
097: LUWMHEYN     112: YIAHUAIM     127: TMMYWHLB
098: FWFPSLDF     113: TAACOVCX     128: XMREVYRR
099: HFIAZQVX     114: LAVWJPNO     129: XRXDMEHN
100: QITHEINU     115: LVPRDAEQ
101: TTAMWAKU     116: GPKLORGH
102: EAFEOXKS     117: AKEWFTXI
103: LFXWLXIW     118: VEPNHKYF
104: QXPTLVMM     119: PPGPYLVM
```

Four good matches

The first good match occurs at position 10:

```
            1
01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ
            ALLEM AND
            TOYEN SLA
```

A plausible completion to the left could be `CITOYENS`, giving

```
            1
01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ
         RE ALLEM AND
         CI TOYEN SLA
```

The second good match occurs at position 26:

```
            1           2           3
01234 56789 01234 56789 01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ REWLR ZEKLS RQXCA
                               ALLE MAND
                               ELAN NEXI
```

A plausible completion to the right could be `LANNEXIONDE` (“l’annexion de”), so we get

```
            1           2           3
01234 56789 01234 56789 01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ REWLR ZEKLS RQXCA
                               ALLE MANDE ENT
                               ELAN NEXIO NDE
```

The third good match occurs at position 80:

```
7           8           9
01234 56789 01234 56789 01234 56789
TAUNX ARZCX IZYHQ LNSYM FWUEQ TELFH
            ALLEM AND
            IONDE LAP
```

The previous letter could be `T` (“. . . tion de la p. . . ”), providing not much help:

```
5           6           7           8           9
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
WAHIE LLXFK VXOKZ OVQIP TAUNX ARZCX IZYHQ LNSYM FWUEQ TELFH
                                  E ALLEM AND
                                  T IONDE LAP
```

And the fourth good match at position 96 also is not helpful:

```
8           9          10          11
01234 56789 01234 56789 01234 56789 01234 56789
IZYHQ LNSYM FWUEQ TELFH QTELQ IAXXV ZPYTL LGAVP
                   ALLE MAND
                   EAUD ETRI
```

Zig-Zag Exhaustion

The four good matches occur as two pairs whose positions differ by 16. This is a bit of evidence for an autokey cipher with a 16 letter key.

This is easily tested: If we really have an autokey cipher, then the fragments should match at another position too, preferably 16 positions apart. Let's try the longest one, `ELANNEXIONDE`, at position 26. We expect exactly one match beside the one we already know, at position $26 - 16 = 10$, or $26 + 16 = 42$. And we get

```
000: HJSVGBVSFZQV   026: ALLEMANDEENT <===
001: QHIGSODLYGWF   027: SARMRGOKDDUY
002: OXTSFWWEFMGE   028: HGZRXHVJCKZW
003: EIFFNPPLLWFV   029: NOEXYOUIJPXP
004: PUSNGIWRVVWO   030: VTKYFNTPONQB
005: BHAGZPCBUMPU   031: AZLFEMAUMGCA
006: OPTZGVMALFVZ   032: GASEDTFSFSBJ
007: WIMGMFLRELAV   033: HHRDKYDLRRKA
008: PBTMWECKKQWI   034: OGQKPWWXQABU
009: IIZWVVVQPMJL   035: NFXPNPIWZRVX
010: POJVMOBVLZMI   036: MMCNGBHFQLYR
011: VYIMFUGRYCJB   037: TRAGSAQWKOSK
012: FXZFLZCEBZCE   038: YPTSRJHQNILE
013: EOSLQVPHYSFV   039: WIFRAABTHBFS
014: VHYQMISERVWN   040: PUEARUENAVTW
015: ONDMZLPXUMOA   041: BTNRLXYGUJXD
016: USZZCIIALEBS   042: ACELORRAINEE <===
017: ZOMCZBLRDRTH   043: JTYOIKLOMUFA
018: VBPZSECJQJIN   044: ANBIBEZSTVBH
019: IEMSVVUWIYOV   045: UQVBVSDZURIH
020: LBFVMNHOXEWA   046: XKOVJWKAQYIT
021: IUIMEAZDDMBG   047: RDIJNDLWXYUB
022: BXZERSOJLRHH   048: KXWNUEHDXKCG
023: EORRJHURQXIO   049: ELAUVAODJSHR
024: VGEJYNCWWYPN   050: SPHVRHOPRXST
025: NTWYEVHCXFOM   ...  ...
```

a perfect accord with our expectations. This gives

```
3           4           5           6           7
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
ZEKLS RQXCA TFENE YBVOI WAHIE LLXFK VXOKZ OVQIP TAUNX ARZCX
              ELA NNEXI ONDE
              ACE LORRA INEE
```

and suggests "Alsace-Lorraine". We complete the middle row that seems to be the keytext:

```
3           4           5           6           7
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
ZEKLS RQXCA TFENE YBVOI WAHIE LLXFK VXOKZ OVQIP TAUNX ARZCX
          A INELA NNEXI ONDE
          A LSACE LORRA INEE
```

If we repeat the fragment from row 3 in row 2 at position $55 = 39 + 16$ we see the very plausible text "l'annexion de l'Alsace-Lorraine", and fill up the rows:

```
3           4           5           6           7
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
ZEKLS RQXCA TFENE YBVOI WAHIE LLXFK VXOKZ OVQIP TAUNX ARZCX
          A INELA NNEXI ONDEL ALSAC ELORR AINEE
          A LSACE LORRA INEET LAFFI RMATI ONDEL
```

To find the key we go backwards in zig-zag:

```
            1           2           3           4
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ REWLR ZEKLS RQXCA TFENE YBVOI
                           IR EALLE MANDE ENTRA INELA NNEXI
                           AI NELAN NEXIO NDELA LSACE LORRA
            1           2           3           4
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ REWLR ZEKLS RQXCA TFENE YBVOI
        SCI TOYEN SLAVI CTOIR EALLE MANDE ENTRA INELA NNEXI
        IRE ALLEM ANDEE NTRAI NELAN NEXIO NDELA LSACE LORRA
            1           2           3           4
01234 56789 01234 56789 01234 56789 01234 56789 01234 56789
LUSIT FSATM TZJIZ SYDZM PMFIZ REWLR ZEKLS RQXCA TFENE YBVOI
AUXAR MESCI TOYEN SLAVI CTOIR EALLE MANDE ENTRA INELA NNEXI
LAVIC TOIRE ALLEM ANDEE NTRAI NELAN NEXIO NDELA LSACE LORRA
```

Now it's certain that we have an autokey cipher and the key is "Aux armes, citoyens"—a line from the "Marseillaise". Using the key we easily decipher the complete plaintext:

> La victoire allemande entraîne l'annexion de l'Alsace-Lorraine et l'affirmation de la puissance allemande en Europe au détriment de l'Autriche-Hongrie et de la France.
>
> [Consequences of the German victory are the annexation of Alsace-Lorraine and the affirmation of the German power at the expense of Austria-Hungary and France.]

8 Similarity of Ciphers

Let Σ be an alphabet, $M \subseteq \Sigma^*$ a language, and K a finite set (to be used as keyspace).

Definition [Shannon 1949]. Let $F = (f_k)_{k\in K}$ and $F' = (f'_k)_{k\in K}$ be ciphers on M with encryption functions

$$f_k, f'_k \colon M \longrightarrow \Sigma^* \quad \text{for all } k \in K.$$

Let $\tilde{F}$ and $\tilde{F'}$ be the corresponding sets of encryption functions. Then F is called **reducible** to F' if there is a bijection $A \colon \Sigma^* \longrightarrow \Sigma^*$ such that

$$A \circ f \in \tilde{F'} \quad \text{for all } f \in \tilde{F}.$$

That is, for each $k \in K$ there is a $k' \in K$ with $A \circ f_k = f'_{k'}$, see the diagram below.

F and F' are called **similar** if F is reducible to F', and F' is reducible to F.

$$\begin{tikzcd} & M \arrow[dl, "f_k"'] \arrow[dr, "f'_{k'}"] & \\ \Sigma^* \arrow[rr, "A"'] & & \Sigma^* \end{tikzcd}$$

Application. Similar ciphers F and F' are cryptanalytically equivalent—provided that the transformation $f \mapsto f'$ is efficiently computable. That means an attacker can break F if and only if she can break F'.

Examples

1. **Reverse** Caesar. This is a monoalphabetic substitution with a cyclically shifted exemplar of the reverse alphabet `Z Y ... B A`, for example

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

We have $K = \Sigma = \mathbb{Z}/n\mathbb{Z}$. Let $\rho(s) := n - s$ the reversion of the alphabet. Then encryption is defined by

$$f_k(s) := k - s \quad \text{for all } k \in K.$$

This encryption function is involutory: $f_k \circ f_k(s) = k - (k - s) = s$. The ordinary Caesar encryption is

$$f'_k(s) := k + s \quad \text{for all } k \in K.$$

Then

$$\rho \circ f_k(s) = \rho(k - s) = n + s - k = (n - k) + s = f'_{n-k}(s),$$

whence $\rho \circ f_k = f'_{\rho(k)}$. Because also the corresponding converse equation holds CAESAR *and Reverse* CAESAR *are similar.*

2. **The** BEAUFORT **cipher** [SESTRI 1710]. This is a periodic polyalphabetic substitution with a key $k = (k_0, \ldots, k_{l-1}) \in \Sigma^l$ (periodically continued):

$$f_k(a_0, \ldots, a_{r-1}) := (k_0 - a_0, k_1 - a_1, \ldots, k_{r-1} - a_{r-1}).$$

Like Reverse CAESAR it is involutory. The alphabet table over the alphabet $\Sigma = \{\mathtt{A}, \ldots, \mathtt{Z}\}$ is in Figure 1. Compare this with TRITHEMIUS-BELLASO encryption:

$$f'_k(a_0, \ldots, a_{r-1}) := (k_0 + a_0, k_1 + a_1, \ldots, k_{r-1} + a_{r-1}).$$

Then as with Reverse CAESAR we have $\rho \circ f_k = f'_{\rho(k)}$, and in the same way we conclude: *The* BEAUFORT *sipher is similar with the* TRITHEMIUS-BELLASO *cipher.*

3. **The Autokey cipher.** As alphabet we take $\Sigma = \mathbb{Z}/n\mathbb{Z}$. We write the encryption scheme as:

$$\begin{array}{ccc|ccc|ccc}
c_0 & = & a_0 + k_0 & & & & & & \\
c_1 & = & a_1 + k_1 & & & & & & \\
\vdots & & & & & & & & \\
c_l & = & a_l + a_0 & c_l - c_0 & = & a_l - k_0 & & & \\
\vdots & & & & & & & & \\
c_{2l} & = & a_{2l} + a_l & c_{2l} - c_l & = & a_{2l} - a_0 & c_{2l} - c_l + c_0 & = & a_{2l} + k_0 \\
\vdots & & & & & & & &
\end{array}$$

Let

$$A(c_0, \ldots, c_i, \ldots, c_{r-1}) = (\ldots, c_i - c_{i-l} + c_{i-2l} - \ldots, \ldots).$$

In explicit form the i-th component of the image vector looks like:

$$\sum_{j=0}^{\lfloor i \rfloor} (-1)^j \cdot c_{i-jl}.$$

and as a matrix A looks like

$$\begin{pmatrix} 1 & & -1 & & 1 & \\ & \ddots & & \ddots & & \ddots \\ & & 1 & & -1 & \\ & & & \ddots & & \ddots \\ & & & & 1 & \\ & & & & & \ddots \end{pmatrix}$$

Then

$$A \circ f_k(a) = f'_{(k,-k)}(a),$$

where $f'_{(k,-k)}$ is the TRITHEMIUS-BELLASO cipher with key $(k_0, \ldots, k_{l-1}, -k_0, \ldots, -k_{l-1}) \in \Sigma^{2l}$. Hence *the Autokey cipher is reducible to the* TRITHEMIUS-BELASO *cipher with period* twice *the key length.* [FRIEDMAN und SHANNON] The converse is not true, the ciphers are not similar: This follows from the special form of the BELLASO key of an autokey cipher.

Note that A depends only on l. The reduction of the autokey cipher to the TRITHEMIUS-BELASO cipher is noteworthy but practically useless: The encryption algorithm and the cryptanalysis are both more complicated when using this reduction. And the reduction is possible only after the keylength l is known.

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

Figure 1: The alphabet table of the SESTRI-BEAUFORT cipher