

Uruguayan Cryptography: Printed Book Covers

Juan José Cabezas
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
jcabezas@fing.edu.uy

Francisco Castro
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
fcr@adinet.com.uy

Joachim von zur Gathen
B-IT, Universität Bonn
Germany
gathen@bit.uni-bonn.de

Jorge Tiscornia
Presidencia Uruguay
Montevideo, Uruguay
jtiscornia@
presidencia.gub.uy

Alfredo Viola
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
viola@fing.edu.uy

Abstract

During the military dictatorship in Uruguay in the 1970s and 1980s, three encrypted documents were sent between members of the urban guerrilla *MLN Tupamaros*. In this work we present the second of these documents, whose text was encrypted by an array of colored circles on the covers of a book.

We introduce some of the history and the processes of image recognition and decryption of this document.

1 Introduction

In Uruguay, a small buffer country between Argentina and Brazil, an urban guerrilla *MLN Tupamaros* emerged in the 1960s and 1970s under the influence of the Cuban Revolution. After years of fame, it was finally defeated in the mid-seventies and a large part of its members were imprisoned in EMR Nº1 (Libertad Prison) and EMR Nº2 (Punta de Rieles Prison), for men and women respectively. The rest sought political asylum in European countries, or died in combat or torture. Prisoners in EMR Nº1 and some exiles played a leading role in this story in the early 1980s involving Uruguay, Germany, and Sweden.

When nine members of their leadership were held hostages (“rehenes políticos”) in different barracks in September 1973, the Tupamaros

left in EMR Nº1 focussed on self-criticism inside the prison and felt the need to make their political positions known to their comrades in exile. The elaboration of self-criticism took them three years (1977-1980) in severe prison conditions on the second floor of the EMR Nº1 where the military kept the prisoners that they considered “dangerous”. The discussions necessary to reach a certain consensus, more or less representative, went through several stages and led to the final draft of the document and a detailed revision among many of them. Two prisoners were charged with the task of elaborating the key and deciding on the vehicle that would carry the message, as well as looking for ways to get the key to family members, so that they could transmit it into exile. A cryptographic *tapestry* (called *carpet* in (Cabezas et al., 2018)) emerged as a medium for clandestinely passing their elaborations and proposals to the outside.

They agreed on an 18-letter alphabet, reduced from the standard 27 letters of Spanish, and its representation in six columns and three rows:

<i>m</i>	<i>a</i>	<i>r</i>	<i>k</i>	<i>o</i>	<i>s</i>
<i>d</i>	<i>i</i>	<i>n</i>	<i>t</i>	<i>e</i>	<i>l</i>
<i>j</i>	<i>u</i>	<i>p</i>	<i>v</i>	<i>h</i>	<i>f</i>

Figure 1: Encryption key used in the *tapestry*.

A total of six colors was used. Each column

represents one of the six colors, and each row one of three among them. The encryption of a letter is the pair of colors at the intersection of its column and row. It was implemented as a *tapestry* elaborated with threads of wool of six different colors and made in the "cross stitch" technique on burlap. This was a craft accepted by the military and many inmates sent similar items outside, passing censorship.

It took many months to make the *tapestry*. First, the proportion and the size of the fabric, and then a reading direction were defined. The agreed text was written in the new alphabet, obtaining the pairs of colors for each letter of the alphabet. Then a *tapestry* design was made on common squared paper, and finally letter by letter, stitch by stitch, the text was transferred from the squared paper to the burlap.

In spaced and controlled visits, the prisoners tried to pass the code to the outside, with the logical difficulties implied in committing family members to things unknown to them. An inmate who was soon to be released and then to be expelled from the country was chosen to remember the key and transmit it to Germany. In exile, the key and the *tapestry* should converge to allow reading of the message. It was also intended that a message with news from the outside be returned and that the same key be used.

What happened to the key, to the alphabet? It came out of prison on thin cigarette rolling paper wrapped in nylon, forming a tiny *pill* that the released prisoner carried in his mouth, as a backup to his memory. But since life is always complex and much more so for those who in the midst of a dictatorship are "expelled" from their country, in a moment of tension he swallowed the pill containing the alphabet. The *tapestry* remained in Uruguay and never reached its intended recipients in exile.

Memory is deceitful and elusive. The exiled prisoner tried to reconstruct the code in order to send a message with news from exile, but only came up with one somewhat inspired by the *tapestry*'s. The released prisoner's new life, the health care given in Germany, the reunions with family and colleagues, the commitment to denounce the living conditions of

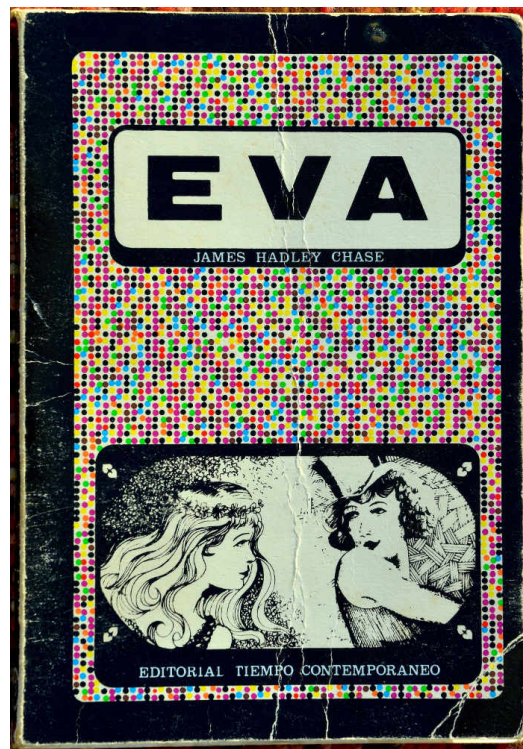


Figure 2: Front cover of the book "Eva".

prisoners in the Libertad Prison, the eagerness to know what had happened in the organization and in the world in all those years, undoubtedly conspired to hinder the recovery of the key. This inconvenience became the greatest obstacle for the successful deciphering effort almost forty years later ((Tiscornia and Cabezas, 2015), also (Cabezas et al., 2018)). That is the first part of this story.

The exiled Tupamaros, who did not even know of the *tapestry*'s existence, sent a message from Sweden to Uruguay. It was encrypted by colored dots on the covers of a little book entitled "Eva" (Figures 2 and 3) whose contents is irrelevant to our story. Two copies were printed in Sweden. One arrived in Uruguay at the time, but did not pass prison censorship. The coding used a method vaguely similar to that of the *tapestry*, but with some changes that complicated its recent analysis, which is the subject of the present work. This is the second part of the story.

Its third part is a *notebook* (*cuadernola* in Spanish) having on its covers the same plaintext as "Eva", but encrypted differently, with a key similar to that of the *tapestry*. The

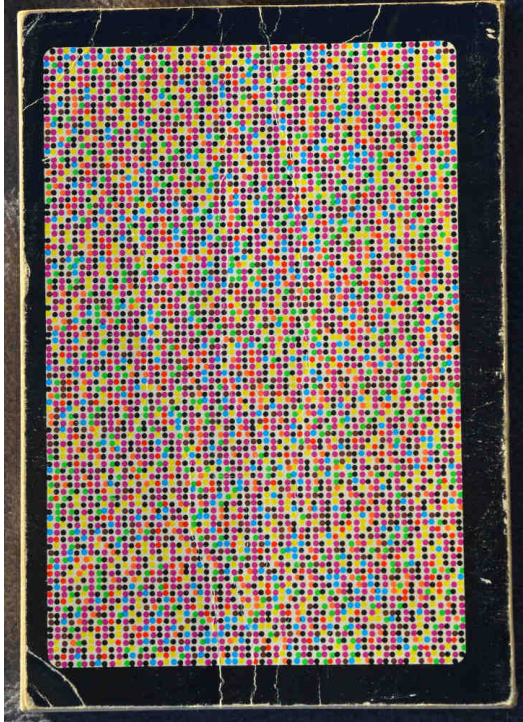


Figure 3: Back cover of the book “*Eva*”.

three key words MARKOS-DINTEL-JUPVHF were identical, but with the rows and columns encoded in “*Eva*”, using the run-length of black and magenta dots as the row indicator. It was printed in Sweden and arrived safely in the Libertad prison and had to be deciphered without any technical help. This demanded a great deal of effort due to alterations, as mentioned above, in the *tapestry*’s encryption method. Enclosed in their cells, against all odds and dedicating weeks of permanent work, eventually they managed to do this. It completed the whole route, but we do not have the *notebook* with us because it was destroyed inside the prison for security reasons as soon as its contents was copied. And this is the third part of the story.

In 2015, thirty-five years later, someone coming from exile in Sweden brings a small book entitled “*Eva*”, a copy of the original that had not passed censorship. Slightly embarrassed, he informs the ex-prisoner that its front and back covers contain the message that came to the Libertad prison in the *notebook*. They consist of regular arrangements of a myriad of colored dots. Whoever has been able to see the *tapestry* and “*Eva*” together under-

stands that the strategy has been the same. The little book is given to an ex-prisoner who has been working on the *Memory of the Libertad Prison*, not knowing the story we have been telling so far. So this researcher started from scratch, consulting his comrades. No prisoner had ever seen “*Eva*”, but some knew about the existence of the *tapestry*, and others had held it in their hands; step by step he advanced until finding the author of the *tapestry*. Although this person had never seen “*Eva*”, he recognized the colored dots. In the first discussion with him, the surprise was great when the researcher found the already moth-eaten *tapestry*. And then, in successive meetings, 35 years later, began a process of retrieving memories; trying to recover the key and the whole process of its creation.

The researcher, besides being surprised, tried to put a rigorous logic to the events in order to help memory, on the one hand, and on the other hand to construct an understandable story. Thus, first questions arose, and after the answers, certainties of what had happened, in order to arrive at the following conclusions. Only one person saw, no matter at what time, the *tapestry*, the *notebook*, and “*Eva*”. This is the author of the *tapestry*. Thirty-five years later, when shown “*Eva*”, of whose existence he had no prior knowledge, a circle is closed. The long-awaited message contained in the *tapestry* never left the home of the author’s family. Miraculously, thirty-five years later it was still there. After so much effort, so many dangers, and so much inventiveness and just out of EMR Nº1, the *tapestry* ran aground in his family’s house.

Almost forty years later, after a deciphering effort at INCO¹, someone unrelated to the authors of the message was able to read it (Tiscornia and Cabezas, 2015). The *tapestry* ended its journey and revealed its content and the thoughts of those imprisoned Tupamaros.

For deciphering “*Eva*”, we have in our hands the *tapestry* and its alphabet and key, and the almost certainty “*Eva*”’s code should be the same. We also know that the *notebook* is a backup of the message sent as “*Eva*”, which never reached the hands of the prisoners be-

¹Computing Science Department at the Universidad de la República, Montevideo, Uruguay

cause the censorship did not allow it. In other words, the *notebook* was a second attempt of a message from exile - this time successful - with many months, or years, between the two.

Time made more memory come to light in that expelled prisoner and with it the key to the *notebook*, in spite of small differences with the *tapestry*'s code. Eventually, the *notebook* could be deciphered. However, this did not help to solve the problem of “*Eva*”, but rather complicated it. Deciphering the *tapestry* was considered more complex due to the digitization itself, the discoloration of the wool, the structure of the fabric, the waves that are generated, and the moth eaten areas. These are the reasons why we started with the *tapestry*. Once again, chance, memory and oblivion intersect. In the course of “*Eva*”'s decipherment, it was detected that the encoding method was not the same as the one used in the *tapestry* and the complications multiplied. But the task was solved (Castro, 2019). All these steps were finished by confirming the correctness of the discovered plaintext with those who received the *notebook* and verifying with the “forgetful” ex-prisoner that the new alphabet was the one in Figure 10.

2 Previous Knowledge

For decoding “*Eva*”, the following previous information was known, or at least could be assumed as a starting point for decrypting the book covers:

- The text was written in Spanish without spaces and using an 18-character alphabet.
- The encoding method of the *tapestry*, namely a substitution cipher with each character encoded as two colors, one for the row and another for the column. (This assumption turned out to be wrong.)
- The key of the *tapestry* (forming a 3×6 matrix): MARKOS-DINTEL-JUPVHF (Figure 1).
- The encoding method of the book covers and particularly the key should not be too different from the one on the *tapestry* since it was decoded by hand.

A few days before final success, an additional hint was provided: the author of the book cover remembered that there was a word in Latin somewhere. Which later proved to be true, as MORTIS was found (by computer) as part of the book cover's key.

3 Image Recognition

Compared to the *tapestry*, the book was found in good condition (Tiscornia and Cabezas, 2015). The inks used in the cover were sufficiently clear and well-preserved to easily distinguish the colors from each other and from the white background. This allowed us to use simple algorithms for the recognition.

The back cover consists of a 61×91 array of colored circles, missing one per corner, totaling 5547 circles. On the front cover there is a rectangular space lacking circles where an image is located instead. That is why the analysis began with the back cover only.

Two difficulties during the recognition process are illustrated in Figure 4:

- Minor cracks, probably originating from bending the cover, causing thin yellow lines (from the acid paper) to appear behind a few circles.
- Each of the eight inks was manually printed from a different plate, and the plates were not perfectly aligned. The amount of misalignment was only problematic on the front cover.

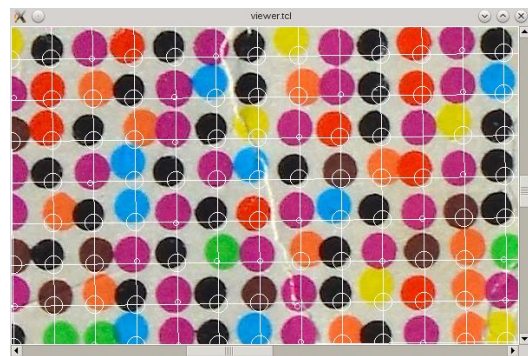


Figure 4: Snapshot of the recognition tool on the back cover. Smaller overlaid white circular marks show which circles were recognized.

A tool was created to assist in the recognition process.

1. First, the pixel coordinates for the centers of the circles located at the corners are specified, along with the number of rows and columns.

With this information the program is able to overlay a grid where an approximation to the center of each circle is calculated. This is done using bilinear interpolation.

2. When the application is run, it shows a list of the circles that it couldn't recognize. For each one of them the color at their center point is also shown (including the RGB values, each in the 0...255 range) at the center point.

The user may then specify the RGB values of recognized colors with their names, like `color 30 192 44 g` for green.

A circle is recognized when a pixel of a valid color is located in the area given by the center point ($\pm 5, \pm 5$) pixel area. Valid colors are those whose RGB triplets differs less than 10 in the 2-norm of a recognized color. In case of ambiguity the point nearest to the center is used.

The front cover, being significantly shorter, was recognized and transcribed by hand once the back was decoded and the key was known.









It was then found that in fact the front cover is the continuation of the back, with the bottom half being the repetition of the beginning of the text.

4 Encoding deduction

4.1 Frequency count

After the image was recognized we only had a text file with a character per circle, and the knowledge that it was encoded with a method similar to the one used in the *tapestry*.

There are eight colors with the following number of occurrences:

Code	Count	Samples for each color
k	1501	 Black
m	1497	 Magenta
y	622	 Yellow
g	466	 Green
c	424	 Cyan
n	394	 Brown
o	350	 Orange
r	293	 Red

The first difference from the *tapestry* is that 8 colors were used instead of 6. Black and magenta are the two most common colors, yet no two consecutive black nor magenta circles can be found.

4.2 Conditional probabilities

When we started, we did not know the reading direction. In order to determine it, Figure 5 exhibits for a circle with color 'y', the probability to be immediately followed by a circle of color 'x'. Here, $P(x)$ is the overall probability for color x , say $P(\mathbf{k}) = 1501/5547$. Additionally, this table can also be seen as the transition table of a first order Markovian model.

The background hue of each cell in Figures 5 and 6 indicates the distance from a memoryless process; the more colorful, the more distant it is. Thus the stronger colors in Figure 5 say that reading is more likely to be horizontal than vertical. Formally the hue is calculated as $h = c_1 \cdot \text{sigmoid}(c_2 \cdot (P(yx|y) - P(x)))$, where green indicates positive and red negative h .²

By building the same table from the transposed array of circles, which would be in effect equivalent to reading the circles vertically, as done in Figure 6, we can see that the transition probabilities are much more similar to a memoryless model.

From this we can infer that the text is encoded horizontally.

4.3 Text direction via compression

Another hint that the text is encoded horizontally can be obtained by compressing the text file with the arrays of circles using a pure variant of LZ-77 (Ziv and Lempel, 1977)³, resulting in 2064 bytes compressed horizontally and 3014 bytes vertically.

4.4 Text direction via common substrings

A conclusive method to ensure the text was indeed written horizontally is to find the longest common repeated substrings in both directions. Figure 7 shows a text `mkmnkomkgn-gnmkmykycmkrmgc` of 26 circles appearing

² $c_1 = 32, c_2 = 10, \text{RGB color} = [224 - h, 224 + h, 224]$.

³Variant with a sliding window of 4kB coding in 9 bits each uncompressed byte and in 17 bits (1 bit for the type, 12 for offset and 4 for length) the matches of length 3 to 18.

P($yx y$)	$x=k$	$x=m$	$x=y$	$x=g$	$x=c$	$x=n$	$x=o$	$x=r$
$y=k$	0.00	44.40	15.53	10.67	8.73	7.07	6.53	7.07
$y=m$	45.16	0.00	15.76	9.95	7.82	8.08	8.15	5.08
$y=y$	31.99	31.19	1.93	2.89	8.04	10.61	7.40	5.95
$y=g$	30.69	30.26	4.08	1.29	7.30	15.45	5.58	5.36
$y=c$	29.48	33.49	8.96	14.39	3.54	4.01	1.89	4.25
$y=n$	34.26	32.74	5.84	3.30	7.87	1.02	9.90	5.08
$y=o$	35.14	31.71	8.57	12.57	6.86	2.29	0.29	2.57
$y=r$	33.79	38.91	10.58	5.12	7.51	0.00	3.41	0.68

Figure 5: Conditional probabilities from the text read horizontally.

P($yx y$)	$x=k$	$x=m$	$x=y$	$x=g$	$x=c$	$x=n$	$x=o$	$x=r$
$y=k$	25.13	27.87	11.73	8.40	7.13	7.33	5.93	6.47
$y=m$	29.39	27.59	10.89	7.82	7.35	6.81	6.01	4.14
$y=y$	26.37	26.53	13.18	8.68	7.88	6.11	6.75	4.50
$y=g$	28.76	25.75	10.52	8.37	6.87	7.51	6.44	5.79
$y=c$	28.54	23.82	12.74	5.42	8.02	8.49	6.13	6.84
$y=n$	25.63	26.14	7.36	11.42	10.66	7.61	7.61	3.55
$y=o$	26.86	26.00	9.43	10.29	6.86	7.43	7.71	5.43
$y=r$	23.89	29.01	12.29	8.87	8.87	5.80	5.46	5.80

Figure 6: Conditional probabilities from the text read vertically.

nkmckcmkomykymkymyookmknmkycmrkmgcckomk~~mn~~omkgngnmkykycmkr
 gmgcroygkmykomkynymk~~mn~~omkgngnmkykycmkr~~gmgc~~cmknkomkrymykmykmk
 rcmkymrkmykyomkygmrokmcckomymk~~mn~~omkgngnmkykycmkr~~gmgc~~kmrykmk

Figure 7: Example showing the longest repeated string.

thrice in the text, with a newline breaking the first two lines which are contiguous in the original.

4.5 Kasiski test

The Kasiski test is commonly used to obtain the key length for Vigenère cryptosystems, and even though we assume from the start that Vigenère was not used, this test can provide additional information.

In the Figure 8 we present the number of pairs of circles of the same color (*#matches*) with the same distance *offset* between them, *factors* shows the prime factorization of the offset.

#matches	offset	factors
1112	173	173
1112	1842	2,3,307
1115	1071	3,3,7,17
1119	782	2,17,23
1121	2679	3,19,47
1122	2294	2,31,37
1126	289	289
1130	2148	2,2,3,179
1130	2538	2,3,3,3,47
1166	6	2,3
1175	530	2,5,53
1562	3	3

Figure 8: Top values of the Kasiski test ($\bar{x} = 1015.80$, $\sigma = 41.47$)

Since there are only 8 colors to encode an alphabet of 18 characters (assuming it was the same alphabet as in the *tapestry*), we have a multi-symbol cryptosystem, where more than one symbol is needed to encode each charac-

ter. The Kasiski test could then be used to obtain a clue about the number of circles per characters needed, in the case that such fixed value existed.

Even though a higher number of coincidences were found at offsets 3 and 6 (vs 1, 2, 4 or 5), the number of matches for offsets congruent with i modulo 3 show a difference small enough to discard trilateral or trinomic cryptosystems (US Army, 1990):

offset	\bar{x}	σ
$\cong 0_{(3)}$	1015.08	37.63
$\cong 1_{(3)}$	1015.23	47.14
$\cong 2_{(3)}$	1017.10	38.97

4.6 Black and magenta runs

For any subset of two or more colors, we consider the substream of those circles whose colors belong to the subset. When we compress these substreams with LZ77, we obtain a significant result: black and magenta circles reach the maximum compression ratio. This occurs because they are interspersed (**kmkmmkmm**...).

If we treated black and magenta as identical colors, then the ciphertext would still have the same meaning. We observed that the run lengths for consecutive black and magenta circles follow a pattern: There are 638 runs of length 1, 694 of length 2, 324 of length 3, but none of length 4 or higher.

At this point we correctly suspected that the run length was pointing out the row number of the 3×6 key, with the other 6 colors referencing the column. This was one of the key observations leading to our cryptanalysis.

Example 1 *The first few circles of the back cover can then be decoded as:*

- **kmyc**: in row 2: yellow and cyan,
- **ky**: in row 1: yellow,
- **mkgr**: in row 2: green and red,
- **mo**: in row 1: orange,
- **kmkn**: in row 3: brown,
- **moy**: in row 1: orange and yellow,
- **kmo**: in row 2: orange,
- **ko**: in row 1: orange,
- **mkmr**: in row 3: red.

The sequence of black and magenta run lengths starts with (21213121321232312121-2112121232...).

Few numbers are repeated consecutively. This suggests that in most places when there are two or more consecutive letters from the same row, the row number is indicated only once.

A few repetitions can be seen in the first part of the back cover. However from about midpage on there are no more consecutive runs with the same length. This can be observed in Figure 9. Maybe they realized during the production process that repeated run lengths could be optimized out.



Figure 9: Positions where consecutive repeated black and magenta run lengths are found on the back cover.

Assuming that our hypotheses are correct at this point, the “*Eva*” code is **not** a simple substitution. The alphabet is split into three parts (rows) of six letters each. In each part, a letter is indicated by one of six colors. Two special colors (black and magenta) indicate the part to be used, but not by colors, rather by the black/magenta run length. We are not aware of any historical cipher that employs such a two-step encryption.

However, we can rewrite it as a simple substitution. The first entry **kmyc** in Example 1 then becomes $((2, y), (2, c))$, and we now work with this simple substitution. But it required a major effort to discover this unusual and clever step in the encryption.

5 Simple Substitution

Once we have reduced the encoding to a simple substitution, we need to know which letter corresponds to each of the 18 entries in this 3×6 table. This table forms the secret key of the resulting cipher.

Traditionally such ciphers are broken by frequency analysis, a process which begins by counting the number of times each character appears in a corpus and in the encrypted text. Since the ratio depends mainly on the language used and applying the substitution does not change the ratios, it is expected to have

a similar order both in the corpus as in the encrypted text.

Instead of using just the frequencies for the individual characters, digrams are commonly used as well (Clark and Dawson, 1998).

In our case we used the simple hill climbing ILS greedy algorithm (iterative improvement local search) which starts from a random solution *key* and keeps looking for the best neighboring one until a local maximum is reached (Lourenço et al., 2003):

```

procedure ILS(key)
  best ← maxn ∈ neighbours(key) {fitness(n)}
  if fitness(best) > fitness(key) then
    return ILS(best)
  else
    return key
  end if
end procedure

```

A key is a permutation of the 18 letters in our alphabet \mathcal{A} , and two keys are neighbors if one can be obtained by swapping two entries of the other.

5.1 Fitness function

For the criterion used to prefer one key over another we use a “fitness” function. Such functions return higher values whenever the plain text obtained (by using a “better” key) has some property closer to a given corpus, like the relative frequencies of letters or polygrams.

A first approach to a fitness function based on a dictionary is just breaking the text in words, so that the sum of the lengths of the words that can also be found in a Spanish dictionary is maximized.

Furthermore we can power the length of the words to a positive constant α to increase the weight of longer words. The longer the word, the lower its probability is to appear inside garbled text.

Example 2 Given the text HELLONE and an English dictionary, $\alpha = 1.8$ would prefer a higher coverage selecting “hell”+“one”, while $\alpha = 2.2$ would prefer a longer match “hello”.

α	‘hello’	‘he’, ‘lone’	‘hell’ ‘one’
1.8	18.1	15.6	19.4
2.0	25	20	25
2.2	34.5	25.7	32.23

Formally, our fitness function $\text{fitness} : \mathcal{A}^* \mapsto$

\mathbb{R} is defined as:

$$\begin{aligned} \text{fitness}(z) &= \max\{\text{fitness}(\phi) + |\omega|^\alpha \\ &\quad \forall (\phi || \sigma_1 || \omega || \sigma_2) = z, \omega \in \mathcal{D}\}, \\ \text{fitness}(\varepsilon) &= 0. \end{aligned}$$

where:

- $\phi, \sigma_1, \omega, \sigma_2 \in \mathcal{A}^*$,
- ε is the empty string,
- ‘||’ is the string concatenation operator,
- $|\omega|$ is the length of the string ω ,
- α is a constant applied to the length, and
- $\mathcal{D} \subset \mathcal{A}^*$ is the Spanish dictionary whose characters were projected to the alphabet \mathcal{A} (without accents or diresis).

The α parameter can be calibrated to make the function have stronger preference to finding longer words vs. finding more words (a higher percentage of the text covered by recognized words). A value of 1.8 was used.

Trying all possible partitions of a text has exponential cost. By remembering (dynamic programming) the fitness result of the first i characters and keeping the dictionary in a trie, we can implement the fitness function with $\mathcal{O}(m)$ memory usage and $\mathcal{O}(n \cdot m)$ CPU usage, where n is the text length and m is the length of the longest word in the dictionary.

5.2 Result

After several executions of the algorithm, the correct key is obtained when the maximum fitness value is obtained:

	r	c	y	n	o	g
1	<i>m</i>	<i>o</i>	<i>r</i>	<i>t</i>	<i>i</i>	<i>s</i>
2	<i>k</i>	<i>l</i>	<i>a</i>	<i>n</i>	<i>d</i>	<i>e</i>
3	<i>j</i>	<i>u</i>	<i>p</i>	<i>v</i>	<i>h</i>	<i>f</i>

Figure 10: The key of “Eva”.

The *tapestry* and the book cover share the third row of the key. However instead of “markos”/“dintel” here we have “mortis” as Latin for death and “klante” as short of “clandestino” (Spanish for clandestine).

Obtaining or not the correct key in a series of independent executions can be modeled as a Bernoulli process of a given probability p .

Since this key was obtained in 643 of 10000 independent executions, we know that the probability of *not* obtaining a correct key in 500 independent executions given $p > 0.05$ has a trivial upper bound of $(1 - p)^{500} < 7 \cdot 10^{-12}$.

On the other hand, given $p \leq 0.05$ the probability of finding the correct key at least 643 times in 10000 independent executions would be less than $1 - \text{bin}(k = 643, n = 10000, p = 0.05) \approx 1.8 \cdot 10^{-16}$, where bin , the binomial cumulative distribution function, equals $\sum_{i=0}^{\lfloor k \rfloor} \binom{n}{i} p^i (1 - p)^{n-i}$.

Thus running 500 independent executions of this algorithm implies testing an average of 1.15 million keys, with the probability of *not* finding the correct key upper-bounded by 10^{-10} . That said, if we were to do a 1.15 million naïve random searches for the key, the probability of *finding* it would be less than $1.8 \cdot 10^{-10}$.

6 Parts of the document's cleartext.

We present the initial part of the cleartext. The opinions and political points of view expressed in this document do not, in any way, reflect necessarily those of the authors or their institutions.

Al recibir digan: Magdalena llegó.

Colores marcan renglones.

Cuando salí no revisaron boca.

[...]

Hay agrupamiento tupas en base a acuerdos, con discrepancias llamado proceso, o simposio. Coincido bastante con ellos, todavía embrión, falta mucho.

Fracasos anteriores, recelo y desesperanza, exilio jode gente.

Va a ser largo y difícil, derrota muy gruesa.

Viejas desviaciones, Cros en Chile, Argentina, Cuba.

Europa: fui bienvenido por todos, hablo con todos grupos, homogéneos y sueltos, no pude hablar (con) seis puntos.

Empezamos a organizar trabajos prácticos, necesarios para MLN, algunos quieren arrimarse América.

Hay unos pocos en Salvador, Nicaragua, Cuba, Colombia.

Intento solidaridad, reorganización, impulso tarea, me queda enorme.

Gran respeto pensamiento (del segundo. Estamos mas frescos que exilio. PUMA (Pautas Unificadas Mínimas y Amplias) muy bien recibido, coincidencia. Olvidé partes, saquen de nuevo. Importante sigan trabajando ustedes.⁴

7 Conclusions

None of the Uruguayan Tupamaros had any in-depth knowledge of cryptography, but still they succeeded in encrypting three fairly long messages in different ways: the *tapestry*, the *notebook*, and the small book “*Eva*”. The *notebook* was deciphered and then destroyed on purpose in its time, the 1980s, but the other two items survive. Both are esthetically pleasing. Their contents were deciphered only in recent years.

For the *tapestry* and “*Eva*”, special-purpose image recognition software was designed in order to translate the sequences of colors into machine-readable text. Understanding the *tapestry* encryption gave several hints for

⁴When you receive this, notify by saying “Magdalena has arrived”. Colors mark rows. When I left [the prison], they did not check my mouth.

There are groups of Tupamaros based on agreements, with discrepancies called process or symposium. I basically agree with them, still in embryonic state, that much is missing. Previous defeats, retreat, and despair, exile fucks up people. It will be a long and arduous way, after a big defeat.

Old disagreements, comrades from Chile, Argentina, Cuba. In Europe, it was welcomed by everybody, I talk to all groups, homogeneous and unattached, I could not talk (to) six groups. We start organizing practical work, necessary for MLN, some want to go with America. There are a few in Salvador, Nicaragua, Cuba, Colombia.

I go for solidarity and reorganization, push towards the goal, but much remains to be done. Great respect for the thinking on the second floor [where the “dangerous” prisoners were housed]. We are fresher [our political thoughts here in prison are clearer] than those in exile. PUMA (minimal and broad unifying guidelines) well received, a coincidence. I forget some parts [of PUMA], please send them again. The important thing is for you to keep working.

“*Eva*”. Some of them proved useful, for example the arrangement of 18 letters in a 3×6 array. Others did not. The arrangement of the 18 row/column pairs was done just by colors in the *tapestry*, but no such arrangement worked for “*Eva*”.

Statistical experiments led to the insight that the run length of black and magenta circles indicates the row number, and any color(s) following such a run indicate the column(s) to be encoded. Once this unusual method was understood, a hill climbing algorithm with an appropriate fitness function broke the resulting simple substitution of “*Eva*”.

Overall, we have unique encryption systems designed by amateurs. They embedded sufficient steganographic cleverness to pass critical inspection by prison guards, were hand-coded with substantial efforts, and the coding method is sufficiently strong to resist attacks by equal adversaries, namely by hand.

References

- Juan José Cabezas, Joachim von zur Gathen, and Jorge Tiscornia. 2018. Uruguayan cryptographic carpet. *Proceedings HistoCrypt 2018*, 21-27, <http://www.ep.liu.se/ecp/contents.asp?issue=149>.
- Francisco Castro. 2019. Printed book cover cryptanalysis: A formal approach. Master’s thesis, Facultad de Ingeniera, Universidad de la Republica. In progress.
- Andrew Clark and Ed Dawson. 1998. Optimization heuristics for the automated cryptanalysis of classical ciphers. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 28:63–86.
- Helena Ramalinho Lourenço, Olivier C. Martin, and Thomas Stützle, 2003. *Iterated Local Search*. In: *Handbook of Metaheuristics*, pages 320–353. Springer US, Boston, MA.
- Jorge Tiscornia and Juan José Cabezas. 2015. El código secreto del tapiz del MLN hecho en el Penal de Libertad. *Reportes Técnicos 15-10*.
- US Army. 1990. *Basic Cryptanalysis. Chapter 5: Monoalphabetic Multilateral Substitution Ciphers*. Headquarters Department of the Army. Washington DC.
- Jacob Ziv and Abraham Lempel. 1977. A universal algorithm for sequential data compression. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 23(3):337–343.