La neuvième couronne

pictor.free.fr – 31 mai 2007

⁂

Cette étude est consacrée au déchiffrement du message crypté figurant au chapitre 9 du roman[1] de Didier Müller, *Les 9 couronnes*.

Le roman est intéressant : il présente plusieurs techniques cryptographiques classiques, où codage et décodage peuvent se faire à la main. Il offre aussi un témoignage original sur la Russie, en alternance avec les explications techniques.

Le dernier chapitre se conclut sur un message chiffré à l'intention du lecteur. Le code utilisé est l'ADFGVX, mais aucune stratégie d'attaque n'est proposée. Le message est rappelé dans l'appendice de ce document.

Le message :

La première chose à noter est que le message n'est pas une implémentation rigoureuse de l'ADFGVX. Il possède en effet 405 caractères, alors que l'on s'attend à en trouver un nombre pair[2].

La raison de cet écart apparaît lorsque l'auteur décrit la transposition (p. 64) :

> « On remplit ensuite le tableau ligne par ligne, et, s'il manque des lettres, on rajoute ce qu'on appelle des nulles : ce sont des lettres sans signification choisies parmi A, D, F, G, V et X. »

Cela ne correspond pas à la réalité historique : pendant la Première Guerre mondiale, les messages n'étaient pas allongés au moyen de nulles, dans le but de compléter le tableau. Cela conduisit donc en particulier à l'existence de « colonnes courtes » dans les messages, sans que cela ne présente de gêne particulière pour les utilisateurs légitimes. Cela étant dit, ce petit arrangement avec l'histoire n'a pas d'effet sur le décryptement qui nous concerne.

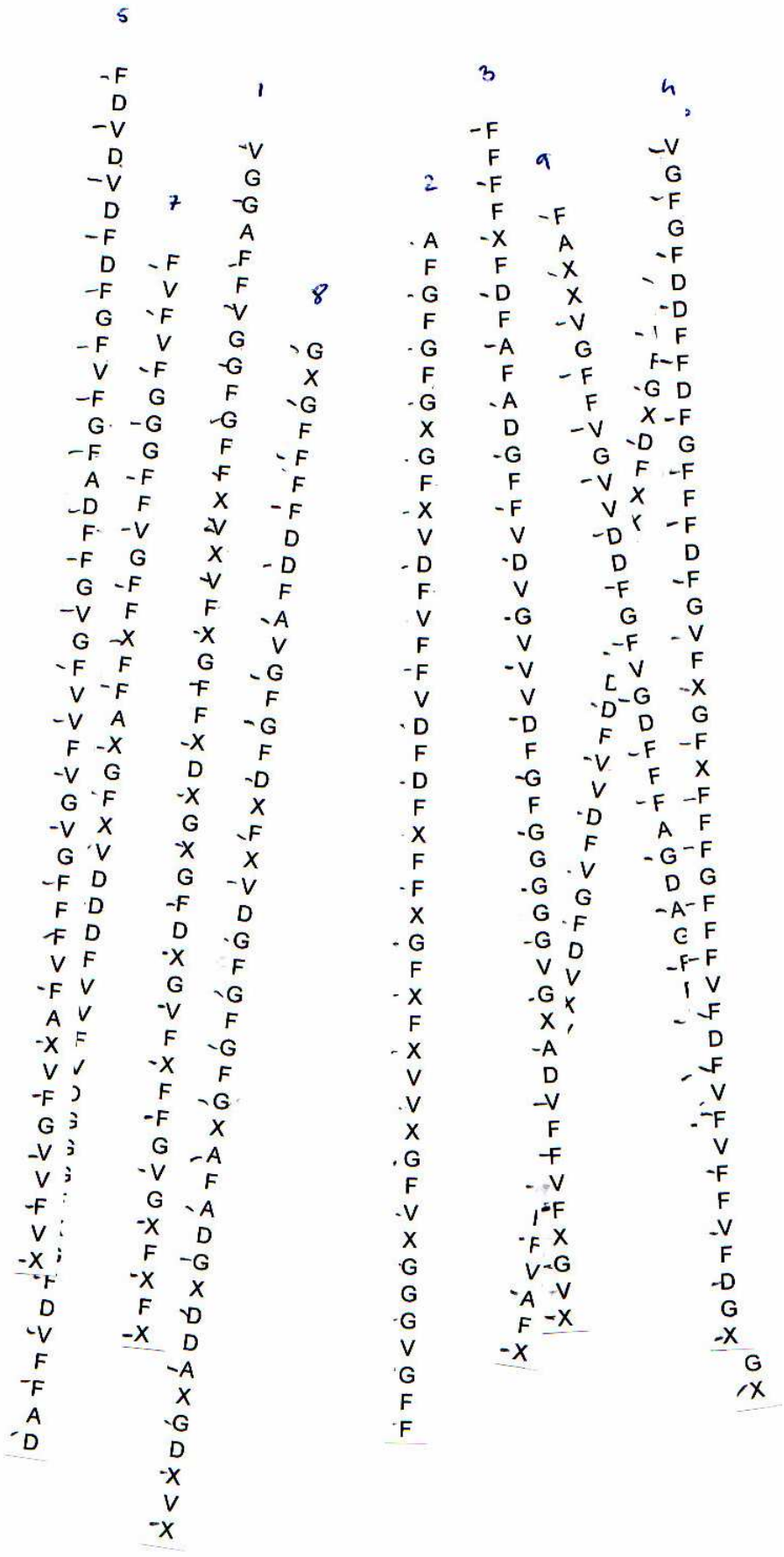
Autre fait notable, l'auteur insiste fortement dans son texte sur la longueur de la clé de transposition. Il ne fait aucun doute que celle-ci comporte 9 caractères.

Le message s'organise donc en 9 colonnes de 45 lignes, que j'ai matérialisées sous la forme d'autant de bandelettes de papier, susceptibles d'être déplacées à volonté.

---

[1] http://www.apprendre-en-ligne.net/crypto/menu/index.html

[2] en ADFGVX, la longueur du message codé est toujours le double de celle du message clair

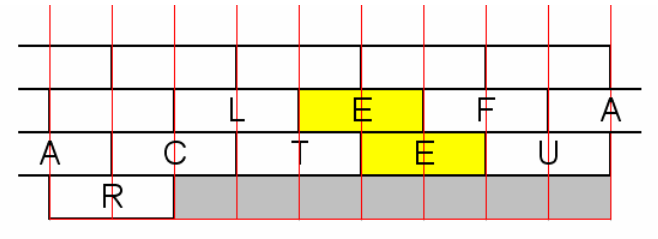


Première approche :

Les messages décryptés des autres chapitres, à l'exception du premier, se terminant par les mots « *le facteur* », il était intéressant de voir si une fin semblable était envisageable pour le dernier message. Au vu des remarques de l'auteur sur :

- l'ajout de nulles,
- la longueur de la clé de transposition,

on pouvait supposer que la structure du message était la suivante :



Dans la figure ci-dessus, la dernière ligne est composée de 2 lettres symbolisant un « R » (dernière lettre du mot *facteur*) et de 7 X « nuls » destinés à compléter le rectangle.

Par ailleurs, la présence de digrammes identiques représentant les deux E (de *le* et de *facteur*), situés à une ligne et une colonne d'intervalle, est une contrainte forte : seuls les arrangements de colonnes 391 ou 396 sont envisageables[3] (avec comme conséquence que la lettre E est codée par le digramme GF).

Dans ce cas, l'ordre des colonne aurait alors été du type : 27--391-- / 27--396-- / 72--391-- / 72--396--.
Un tel espace de recherche représente 4.4! (=96) cas à traiter et laisse encore beaucoup d'incertitudes. Les tentatives d'identifier les autres lettres de *le facteur* dans certains de ces cas se sont soldées par un échec.

Deuxième approche :

Dans le cryptage ADFGVX, les colonnes transposées sont de deux familles différentes :

- lorsque la longueur de la clé de transposition est paire (2N), N colonnes sont de la famille « lettre initiale » et N colonnes sont de la famille « lettre finale ». Les colonnes du premier groupe sont aux positions 1, 3, 5, etc. celles du dernier aux positions 2, 4, 6, etc.
- lorsque la longueur de la clé de transposition est impaire (2N+1), les colonnes se répartissent également en deux familles. Mais cette fois, il y a N+1 colonnes « lettre initiale aux lignes impaires et lettres finales aux lignes paires » et N colonnes « lettre finale aux lignes impaires et lettre initiale aux lignes paires ». Les colonnes de la première famille sont aux positions 1, 3, 5, etc., celles de la deuxième sont aux positions 2, 4, etc.

[3] aux positions 5, 6 et 7.

Dans les deux cas (longueur paire ou impaire de la transposition), les deux familles de colonnes ont leurs propres distributions en fréquence. Il est alors possible, par analyse de ces fréquences, de regrouper les colonnes par similarité de leurs distributions.

J'ai procédé à une simple analyse pour chaque colonne. A noter que dans la figure ci-dessous, chaque colonne est séparée en deux : une marque dans la moitié gauche (resp. droite) de la colonne signifie que la lettre a été trouvée en position impaire (resp. paire)[4].

La figure ci-dessous résume ce qui a été trouvé : un + (resp. -) signifie que la lettre désignée est fréquente aux lignes impaires (resp. paires) de la colonne donnée. Les tendances les plus significatives ont été mises en évidence en vert.

	1	2	3	4	5	6	7	8	9
A			+		-	+	-	+	
D	-	+		-	-	+	-	-	-
F	-	-	-	+	+	-	+	-	+
G	-	+	+	-	-		-	+	
V	+		-			+	+		
X	+					-	+	-	

Groupes détectés	
368	57
145789	26
12368	4579
1457	238
167	3
17	68

Cet arrangement permet de distinguer deux familles de colonnes :

- 1, 4, 5, 7, 9 d'une part (« G fréquent et F infréquent aux lignes paires »), et
- 2, 3, 6, 8, d'autre part. (« G fréquent et F infréquent aux lignes impaires »)

[4] l'analyse de fréquence est facilitée par les marques visibles sur les bandelettes

Du fait que le premier groupe possède un membre de plus que le deuxième, on peut affirmer que :

- les colonnes 1, 4, 5, 7, 9 se trouvent aux positions impaires,
- les colonnes 2, 3, 6, 8 se trouvent aux positions paires.

L'étape suivante consiste à reconstituer des paires de colonnes. La méthode est inspirée du *Cours de cryptographie* du Général Givierge.

L'opération consiste à accoler une colonne « position impaire » à une colonne « position paire ». Il s'agit d'établir une nouvelle analyse des fréquences, cette fois portant sur les digrammes, en espérant trouver une distribution proche de la langue française.

La figure suivante représente l'analyse des fréquences pour tous les appareillages possibles : colonne 1 et colonne 2, colonne 1 et colonne 3, colonne 1 et colonne 6, etc.

Le but de cette analyse est de valider les paires de colonnes pour lesquelles la distribution des fréquences des digrammes est comparable à celle des lettres de l'alphabet dans un texte français. En particulier, on souhaiterait détecter :

- un pic marqué et unique (la lettre E)
- un contraste relativement élevé (une distribution trop uniforme signalerait des données aléatoires)
- de nombreuses cases vides (correspondant aux lettres rares de l'alphabet et aux chiffres).

A cette fin, les indicateurs suivants ont été crées:

- « E » : fréquence du (des) digramme(s) le(s) plus fréquent (supposé représenter E)
- « cases occupées » : l'« étalement » de la distribution (ramassée, c'est mieux)
- « singletons » : le nombre de digrammes uniques
- « doublons » : le nombre de digrammes existant deux fois
- « s+d » : la somme des deux précédents

Ils figurent sous la forme de notes au dessous du tableau.

Les flèches au-dessus du tableau représentent des couples (et une colonne seule) que j'ai considérés comme intéressants[5].

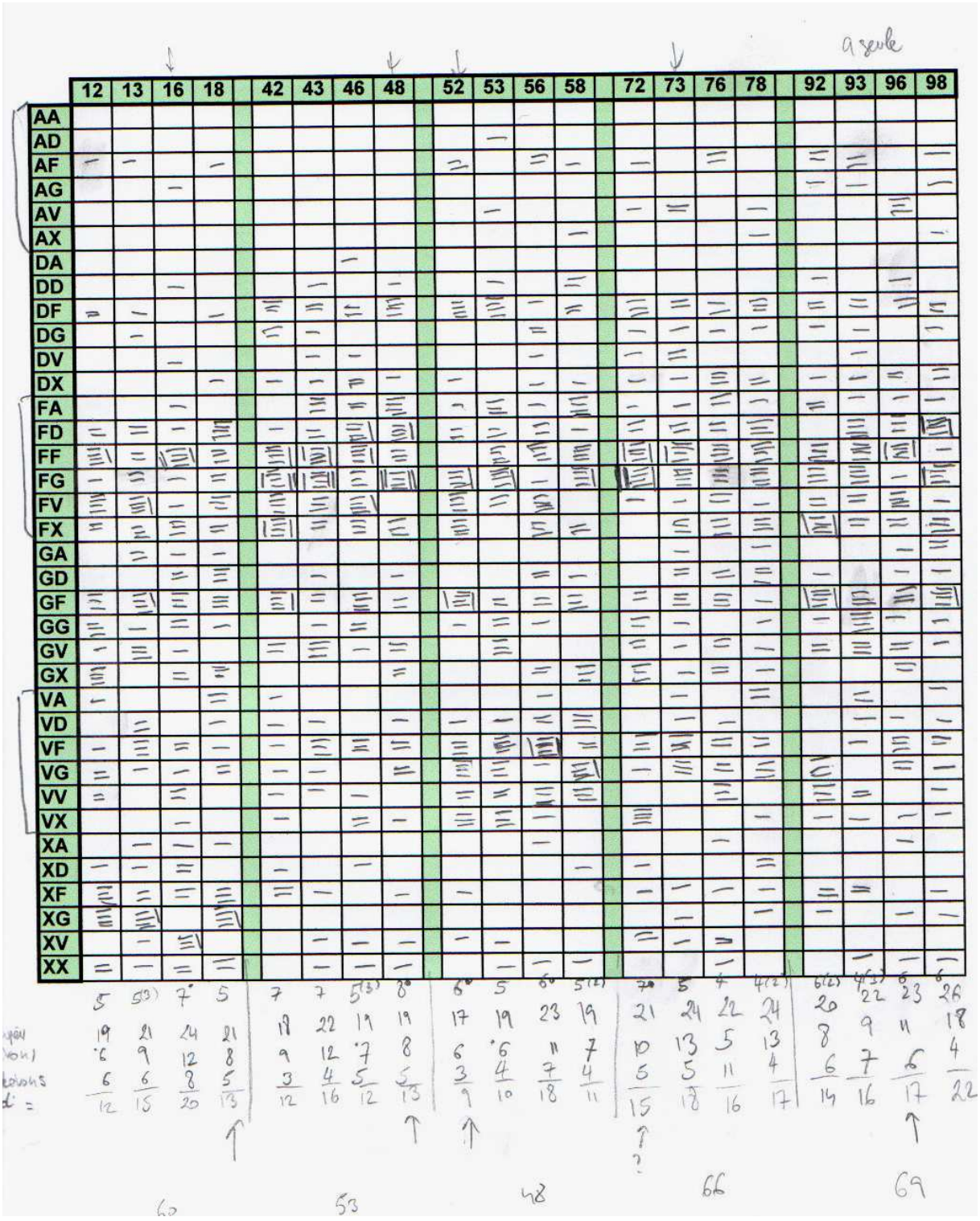
Si, à ce point, les combinaisons trouvées sont correctes, les opérations de transposition et de substitution ont alors été démêlées : il devient possible d'attaquer chaque problème par sa solution classique respective, ce qui représente une simplification considérable.

La substitution simple est un problème facile à résoudre par analyse de fréquences. Une fois celle-ci réalisée, la transposition se résout par recherche d'anagrammes.
En théorie c'est simple, en pratique cela l'est moins. D'autant que les combinaisons de colonnes retenues ci-dessus ne sont pas absolument certaines.

Dans le cas qui nous intéresse, c'est la colonne 9 que l'on a estimée comme solitaire. Elle se trouve donc en position 9. Il reste alors retrouver l'ordre des 4 paires de colonnes les unes par

[5] vous pouvez être amenés à des conclusions différentes sur la base des même données

rapport aux autres, ce qui implique d'examiner un nombre de transpositions égal à 4! = 24 cas, une fois la clé de substitution retrouvée.



Là encore, le chemin s'annonçait trop long pour tâtonner à la main. C'est ainsi que s'est terminée ma deuxième approche.

Note du 11 juin 2007 :

Dans ce qui précède, les paires de colonnes ont été formées après étude des « indicateurs » cités. Une approche plus rigoureuse peut être réalisée : seront retenues les paires pour lesquelles l'indice de coïncidence global sera maximum.

Partant du principe que la colonne 9 est isolée, les indices de coïncidence de chacune des paires ont été calculés. Ils sont le suivant (en millièmes) :

	1	**4**	**5**	**7**
2	43	69	58	58
3	46	52	46	35
6	43	50	40	31
8	39	62	49	33

Exemple de lecture : la distribution des lettres (de AA à XX) obtenue en appareillant les colonnes 1 et 2 correspond à un indice de coïncidence de 0,043.

Il y a 24 combinaisons de paires possibles. Pour chaque combinaison (trois premières lignes), le tableau ci-dessous indique, la somme des quatre indices, en millièmes (dernière ligne).

12						13						16						18					
43		46		48		42		46		48		42		43		48		42		43		46	
56	58	53	58	53	56	56	58	52	58	52	56	53	58	52	58	52	53	53	56	52	56	52	53
78	76	78	73	76	73	78	76	78	72	76	72	78	73	78	72	73	72	76	73	76	72	73	72
168	175	172	177	182	180	188	195	187	203	197	206	191	196	186	202	198	209	185	183	180	189	182	193

Troisième approche :

Le calcul de l'indice de coïncidence est ma troisième approche.

A ce sujet, l'auteur précise dans son texte (p. 65) :

> « (...) Je pourrais essayer d'écrire un programme qui teste toutes les transpositions possibles et qui élimine les transpositions qui manifestement ne sont pas plausibles, parce que les fréquences des bigrammes sont trop éloignées de la théorie. Mais je ne sais pas si c'est facile - j'en doute un peu. Inutile de dire que cette méthode ne marchera jamais si la clef est trop longue...
>
> - Mais c'est l'horreur ! Même si l'ordinateur ne garde que cent transpositions plausibles, il te faudra encore un temps monstrueux pour trouver la bonne. »

L'idée est bonne : une clé de transposition de longueur 9 est suffisamment courte pour qu'une analyse simple puisse être menée sur les 362 880 cas de figure. Il est également vrai que cette méthode est inapplicable pour des clés de d'une longueur de 20 à 25 caractères, longueur typiquement utilisée pendant la Première guerre mondiale.

Par contre, la deuxième remarque est moins compréhensible (« *il faudra un temps monstrueux* »). On verra plus loin que la vérité est tout autre : la transposition réellement utilisée est immédiatement identifiable parmi 362 880.

Une lecture du forum[6] consacré au message indique que plusieurs personnes ont réussi à le décrypter en calculant l'indice de coïncidence[7] pour chacune des 9! combinaisons de colonnes. Cette méthode permet d'évaluer au cas par cas la pertinence de la distribution fréquentielle des digrammes.

Je me suis attelé à cette tâche. Ce fut finalement la manière la plus rapide, la moins laborieuse, et, surtout, celle qui fournit la solution du problème.

J'ai conçu un programme en C qui calcule l'indice de coïncidence du message généré par l'arrangement des 9 colonnes, pour chaque arrangement possible.

Le code permettant de balayer les arrangements de n objets a été emprunté au cours de statistiques informatisées *Stat 606* de Kerby Shedden[8]. La boucle principale de mon code est le `while{}` de sa fonction `test_perm_key()`.

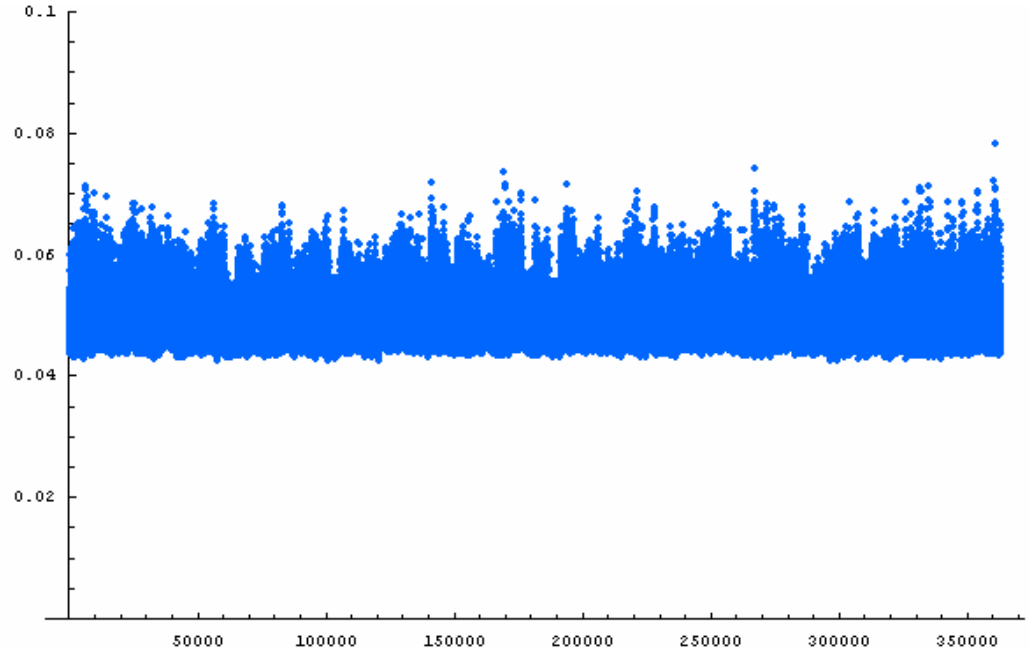
Voici un extrait des informations générées par le programme :

```
key:       0 8,7,6,5,4,3,2,1,0 IC = 0.051721
key:       1 7,8,6,5,4,3,2,1,0 IC = 0.047239
key:       2 8,6,7,5,4,3,2,1,0 IC = 0.049997
key:       3 6,8,7,5,4,3,2,1,0 IC = 0.046500
key:       4 7,6,8,5,4,3,2,1,0 IC = 0.047140
key:       5 6,7,8,5,4,3,2,1,0 IC = 0.048519
key:       6 8,7,5,6,4,3,2,1,0 IC = 0.046598
key:       7 7,8,5,6,4,3,2,1,0 IC = 0.046155
key:       8 8,5,7,6,4,3,2,1,0 IC = 0.047042
```

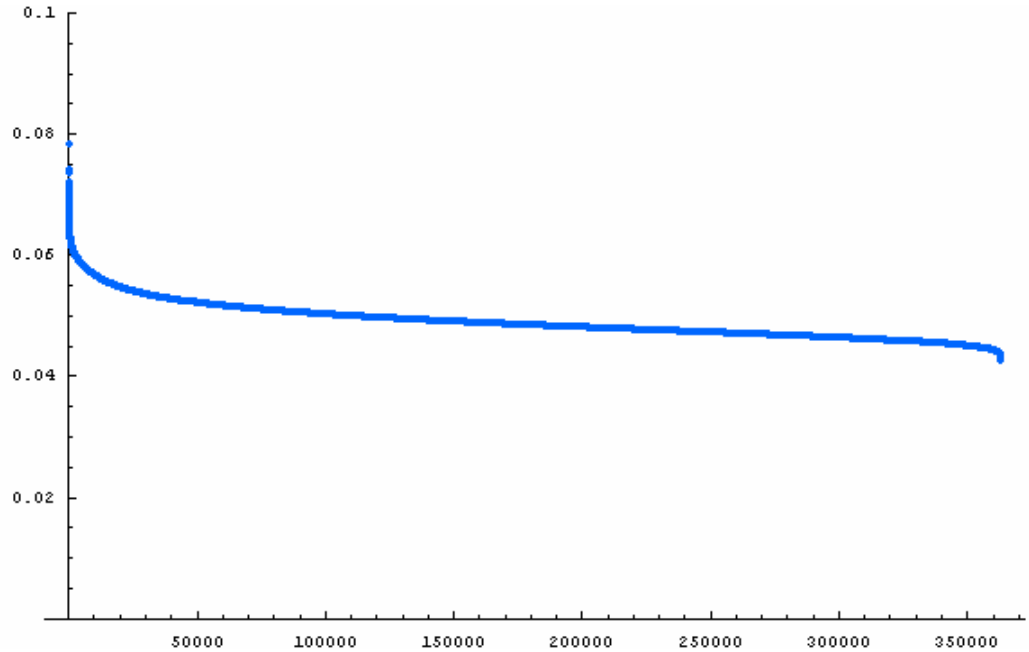
Le programme affiche dans l'ordre : le numéro d'ordre de la permutation (il commence à 0), l'ordre des colonnes[9], et l'indice de coïncidence[10] (IC). La sortie du programme est redirigé vers un fichier, qui fait environ 16 Mo, difficilement manipulable avec les outils de bureautique standard[11].

Les indices de coïncidence peuvent être présentées graphiquement. D'abord dans l'ordre où ils ont été calculés.

[6] http://www.apprendre-en-ligne.net/forums/viewforum.php?f=5
[7] une définition se trouve sur http://fr.wikipedia.org/wiki/Indice_de_co%C3%AFncidence
[8] http://www.stat.lsa.umich.edu/~kshedden/Courses/Stat606/C-Code/enumerate.c
[9] dans le programme, la colonne 1 est notée 0, la colonne 2 est notée 1, etc.
[10] pour un « alphabet » de 36 caractères
[11] sous Windows, le logiciel Notepad++ permet de visualiser le fichier sans difficulté



Ils peuvent aussi être présentés par ordre décroissant :



Les trois plus grandes valeurs de l'indice sont, dans l'ordre décroissant[12] : 0.0784, 0.0743 et 0.0737. Autrement dit, la valeur maximale se détache très nettement des autres.

Le message correspondant à la permutation pour laquelle l'indice de coïncidence est maximum est ensuite étudié. Si le message suit le schéma initié au message précédent, on s'attend à trouver la phrase « *Une belle muse*… ». On se focalisera en particulier sur la recherche d'un motif du type ABBA correspondant à la fin du mot *belle*.

[12] sous Windows, le logiciel CMsort permet de trier rapidement des fichiers volumineux

La conception, l'écriture et le test du programme a pris environ cinq heures.
L'exécution du programme dure environ 20 secondes.
L'identification du motif clé est instantanée.
La recherche de la clé de substitution prend dix minutes, en prenant le temps de la savourer.

Epilogue :

La recherche de la solution de ce message a été longue, comme probablement beaucoup d'autres en cryptanalyse. En plus de permettre de lire le dernier message, que j'ai trouvé émouvant, cette étude a l'intérêt de montrer le potentiel de l'utilisation combinée de plusieurs approches pour les messages dont la longueur de la clé de transposition ne permettrait pas une balayage exhaustif des transpositions.

En effet, pour les cas où un tel balayage serait inenvisageable, la méthode présentée lors de la deuxième approche, qu'il est possible de réaliser à la main, permet de « casser la factorielle » impactant la recherche de l'indice de coïncidence maximum.

Prenons par exemple le cas d'un message dont la clé de transposition a une longueur de 19. Il y a 19! (environ10^{17}) indices à calculer. Si, par la technique présentée à la deuxième approche, on arrive à séparer les colonnes en deux groupes, le nombre d'indices à calculer passe à 9! 10! (environ10^{12}).
Si par ailleurs, on arrive à reconstituer les paires de colonnes, le nombre de cas à tester est de 9! (9 paires de colonnes dont il faut retrouver l'ordre, sachant que la colonne solitaire se trouve à droite).

Appendice :

Le message codé est :

```
VGGAF FVGGF GFFXV XVFXG FFXDX GXGFD XGVFX FFGVG XFXFX AFGFG FGXGF
XVDFV FFVDF DFXFF XGFXF XVVXG FVXGG GVGFF FFFFX FDFAF ADGFF VDVGV
VVDFG FGGGG GVGXA DVFFV FXGVX VGFGF DDFFD FGFFF DFGVF XGFXF FFGFF
FVFDF VFVFF VFDGX FDVDV DFDFG FVFGF ADFFG VGFVV FVGVG FFFVF AXVFG
VVFVX FVAGG XVFGX DFXXX FGFVD DFVVD FVGFD VXVFD FAXFF FVAFX FVFVF
GGGFF VGFFX FFAXG FXVDD DFVVF VDGGG FXGFD VFFAD GXGFF FFDDF AVGFG
FDXFX VDGFG FGFGX AFADG XDDAX GDXVX FAXXV GFFVG VVDDF GFVGD FFFAG
DAGFF FVGFF FGFGD VFFGX
```

Les colonnes sont :

```
Col 1 : VGGAFFVGGFGFFXVXVFXGFFXDXGXGFDXGVFXFFGVGXFXFX
Col 2 : AFGFGFGXGFXVDFVFFVDFDFXFFXGFXFXVVXGFVXGGGVGFF
Col 3 : FFFFXFDFAFADGFFVDVGVVVDFGFGGGGGVGXADVFFVFXGVX
Col 4 : VGFGFDDFFDFGFFFDFGVFXGFXFFFGFFFVFDFVFVFFVFDGX
Col 5 : FDVDVDFDFGFVFGFADFFGVGFVVFVGVGFFFVFAXVFGVVFVX
Col 6 : FVAGGXVFGXDFXXXFGFVDDFVVDFVGFDVXVFDFAXFFFVAFX
Col 7 : FVFVFGGGFFVGFFXFFAXGFXVDDDFVVFVDGGGFXGFDVFFAD
Col 8 : GXGFFFFDDFAVGFGFDXFXVDGFGFGFGXAFADGXDDAXGDXVX
Col 9 : FAXXVGFFVGVVDDFGFVGDFFFAGDAGFFFVGFFFGFGDVFFGX
```