數學傳播　30卷2期, pp. 55-76

傳統密碼之旅(下)

沈淵源

八. 希爾密碼(Hill Ciphers)

上面介紹過的每一個密碼系統中, 改變明文中的一個字母, 在密文中也有一個字母隨著改變。在位移、仿射與代換 (下一節) 密碼系統中, 密文中的一個字母來自明文中唯一的一個字母。如此一來, 頻率分析法在這些系統中尋找鑰匙時就無往不利了。在維吉內爾密碼中, 由於使用了與鑰匙等長的字母區塊, 以致直接頻率分析困難重重; 然而一旦算出鑰匙長度之後, 再分頭用頻率分析予以各個擊破, 仍有機會破解, 此乃因爲在每一個字母區塊當中的字母彼此之間並沒有任何的互動。雷斯特·希爾 (Lester Hill) 在 1929 年所發明的密碼法 [4] 當中, 巧妙的運用了線性代數的技巧讓字母彼此之間開始互動起來, 藉以增加系統的安全等級, 因而得到更好更實用的一個密碼系統。

首先, 我們選取一個正整數 n, 如 $n = 3$。鑰匙是一個在模 26 之下的 n 階方陣 M。例如, 令

$$M = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 7 & 11 \end{pmatrix}。$$

信息可寫成一長度爲 n 的列向量序列。如信息爲 abc, 先轉換成列向量 $(0, 1, 2)$。怎麼加密呢? 只要在列向量 $(0, 1, 2)$ 的右邊乘上方陣 M 並在模 26 之下縮簡即可, 如下:

$$(0, 1, 2)M = (0, 1, 2)\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 7 & 11 \end{pmatrix} \equiv (4, 19, 2) \pmod{26},$$

所以得到密文就是 ETC。

```
In[52]:= M={{1,2,3},{4,5,6},{0,7,11}};Mod[{0,1,2}.M,26]
Out[52]= {4,19,2}
```

爲了解密, 我們需要 M 的行列式值與 26 互質。在上例中, $\det(M) = 9$, 所以 M 的逆方陣就是

$$\frac{1}{9}\begin{pmatrix} 13 & -1 & -3 \\ -44 & 11 & 6 \\ 28 & -7 & -3 \end{pmatrix}。$$

因爲在模 26 之下 3 是 9 的乘法反元素, 所以將 $\frac{1}{9}$ 用 3 來取代, 乘開並在模 26 之下縮簡得

$$M' = M^{-1} = \begin{pmatrix} 13 & 23 & 17 \\ 24 & 7 & 18 \\ 6 & 5 & 17 \end{pmatrix}。$$

在 MATHEMATICA 中, 上面兩個步驟可合併如下; 但需注意必須使用指令 `PolynomialMod` 才行得通。

```
In[53]:= M'=PolynomialMod[Inverse[M],26]

Out[53]= {{13,23,17},{24,7,18},{6,5,17}}
```

解密時, 可在密文的右邊乘上 M 的逆方陣, 也就是它的乘法反元素 $M' = M^{-1}$, 並在模 26 之下縮簡即可, 如下:

$$(4,19,2)M' = (4,19,2)\begin{pmatrix} 13 & 23 & 17 \\ 24 & 7 & 18 \\ 6 & 5 & 17 \end{pmatrix} \equiv (0,1,2) \pmod{26}。$$

```
In[54]:= Mod[{4,19,2}.M',26]

Out[54]= {0,1,2}
```

一般而言, 將明文分割成含 n 個字的區塊並用 $a = 0$, $b = 1$, $\cdots$, $z = 25$ 轉換成長度爲 n 的列向量。如上例的方陣 M, 假設我們的明文 `ma` 是

`hihappybirthdaytoyou`

在此信息附加一個 `x` 使得長度變成 $n = 3$ 的倍數, 以符號 `mx` 表示之。所以區塊變成列向量, 而信息 `mx` 則變成一個 $n \times k$ 的矩陣。加密時, 在明文矩陣 m 右方乘加密矩陣 M, 可得密文矩陣 c 如下:

```
In[55]:= abc="abcdefghijklmnopqrstuvwxyz"; num="000102030405060708091011121314151617181920212223242 5";
         digitalize=Table[StringTake[abc,{i}]->StringTake[num,{2*i-1,2*i}],{i,1,26}];
         alphabetize=Table[StringTake[num,{2*i-1,2*i}]->StringTake[abc,{i}],{i,1,26}];
         Q0[plaintext_]:=StringReplace[plaintext,digitalize];
         A[digit_]:=StringReplace[digit,alphabetize];
In[60]:= ma="hihappybirthdaytoyou"; mx=ma<>"x";
         m0=Table[StringTake[Q0[mx],{i,i+1}],{i,1,2StringLength[mx],2}]//ToExpression;
         m=Table[{m0[[i]],m0[[i+1]],m0[[i+2]]},{i,1,StringLength[mx],3}]
Out[60]= hihappybirthdaytoyoux
Out[62]= {{7,8,7}, {0,15,15}, {24,1,8}, {17,19,7},{3,0,24},{19,14,24},{14,20,23}}hihappybirthdaytoyoux
In[63]:= c=Mod[m.M,26]
Out[63]= {{13,25,16},{8,24,21},{2,5,10},{15,22,8},{3,18,13},{23,16,15},{16,3,25}}
```

最後, 將每一列向量轉換回到字母區塊, 再將區塊合併之可得密文

NZQIYVCFKPWIDSNNQPQDZ

```
In[64]:= cf=Flatten[c]; cs=Table[If[cf[[i]]<10,"0"<>ToString[cf[[i]]],ToString[cf[[i]]]],{i,StringLength[mx]}]
Out[64]= {13,25,16,08,24,21,02,05,10,15,22,08,03,18,13,23,16,15,16,03,25}
In[65]:= ca=A[cs]//StringJoin//ToUpperCase
Out[65]= NZQIYVCFKPWIDSNXQPQDZ
```

解密時, 先逆向轉換成密文矩陣 c, 然後在 c 右邊乘以 $M' = M^{-1} \pmod{26}$, 如此得回明文矩陣 m, 最後再轉換爲原信息。

```
In[66]:= cl=ToLowerCase[ca]; cm=Table[StringTake[cl,{i}],{i,StringLength[cl]}]; d=Q0[cm]//ToExpression;
         Table[{d[[i]],d[[i+1]],d[[i+2]]},{i,1,StringLength[cl],3}]
Out[67]= {{13,25,16},{8,24,21},{2,5,10},{15,22,8},{3,18,13},{23,16,15},{16,3,25}}
In[68]:= %==c
Out[68]= True
In[69]:= mf=Mod[c.M',26]//Flatten;
         ms=Table[If[mf[[i]]<10,"0"<>ToString[mf[[i]]],ToString[mf[[i]]]],{i,StringLength[cl]}]//A//StringJoin
Out[70]= hihappybirthdaytoyoux
In[71]:= StringDrop[ms,-1]==ma
Out[71]= True
```

很顯然的, 變更明文中的一個字母, 通常會導致密文中 n 個字母的改變。例如, 若將 block $= (1, 11, 14, \ldots)$ 改成 clock $= (2, 11, 14, \ldots)$, 則密文中前三個字母由 TZP 變爲 UBS。

$$\text{blo} = (1, 11, 14) \longmapsto (1, 11, 14) \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 7 & 11 \end{pmatrix} = (19, 25, 15) = \text{TZP}$$

$$\text{clo} = (2, 11, 14) \longmapsto (2, 11, 14) \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 7 & 11 \end{pmatrix} = (20, 1, 18) = \text{UBS}$$

這使得頻率分析的效率降低許多。雖然如此, 對小的 n 值而言, 用頻率分析來破解並非不可能。二字串及三字串已經有人計算而且統計出來其出現之頻率。三字串以上則因其可能數目變得太大了, 且若沒有提供大量的密文, 各種字串的數目會低到很難從其中獲取有意義的資訊。如此一來, 頻率分析更是英雄無用武之地。

下面我們一起來思考那四種攻擊法如何進行。

1. 密文攻擊法:用密文攻擊法要破解希爾密碼是困難的, 但希爾密碼卻俯首稱臣於其他幾個攻擊法之下。
2. 已知明文攻擊法:如果我們不知道 n, 那麼就試幾個不同的 n 值, 直等到找到正確的爲止。所以假設 n 爲已知。如果我們有 n 個長度爲 n 的明文區塊, 則我們可使用明文及其對應的密文得到一個關於 M (或是它的乘法反元素, 這可能還更有用) 的矩陣方程式。例如, 假設我們現在已經知道 $n = 2$ 而且我們有明文 howareyoutoday $=$

$$7 \;\; 14 \qquad 22 \;\; 0 \qquad 17 \;\; 4 \qquad 24 \;\; 14 \qquad 20 \;\; 19 \qquad 14 \;\; 3 \qquad 0 \;\; 24$$

其對應的密文爲 ZWSENIUSPLJVEU =

$$25\ \ 22 \qquad 18\ \ 4 \qquad 13\ \ 8 \qquad 20\ \ 18 \qquad 15\ \ 11 \qquad 9\ \ 21 \qquad 4\ \ 20$$

前兩個區塊得到矩陣方程式

$$\begin{pmatrix} 7 & 14 \\ 22 & 0 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \equiv \begin{pmatrix} 25 & 22 \\ 18 & 4 \end{pmatrix} \pmod{26}。$$

可惜矩陣 $\begin{pmatrix} 7 & 14 \\ 22 & 0 \end{pmatrix}$ 的行列式值爲 -308, 此數在模 26 之下沒有乘法反元素 (雖然這個矩陣可用來大大的縮減可能的加密矩陣之數目)。因此我們將方程式最後一列取代爲, 譬如說, 第五區塊而得到

$$\begin{pmatrix} 7 & 14 \\ 20 & 19 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \equiv \begin{pmatrix} 25 & 22 \\ 15 & 11 \end{pmatrix} \pmod{26}。$$

這一次, 矩陣 $\begin{pmatrix} 7 & 14 \\ 20 & 19 \end{pmatrix}$ 在模 26 之下是可逆的:

$$\begin{pmatrix} 7 & 14 \\ 20 & 19 \end{pmatrix}^{-1} \equiv \begin{pmatrix} 5 & 10 \\ 18 & 21 \end{pmatrix} \pmod{26}。$$

```
In[72]:= PolynomialMod[Inverse[{{7,14},{20,19}}],26]
Out[72]= {{5,10},{18,21}}
```

我們得到

$$M \equiv \begin{pmatrix} 5 & 10 \\ 18 & 21 \end{pmatrix} \begin{pmatrix} 25 & 22 \\ 15 & 11 \end{pmatrix} \equiv \begin{pmatrix} 15 & 12 \\ 11 & 3 \end{pmatrix} \pmod{26}。$$

```
In[73]:= Mod[{{5,10},{18,21}}.{{25,22},{15,11}},26]
Out[73]= {{15,12},{11,3}}
```

因希爾密碼難於防守此種攻擊, 所以就不能看成是非常強的密碼。

3. 選擇明文攻擊法: 選擇明文攻擊法可採取相同的策略來進行, 不過會快一些。再一次地, 如果你不知道 n, 試幾個不同的 n 值, 直到行的通爲止。所以假設 n 爲已知。選擇明文的第一區塊爲 $\mathrm{baaa}\cdots = 1000\cdots$, 第二區塊爲 $\mathrm{abaa}\cdots = 0100\cdots$, 如此繼續至第 n 區塊爲 $\cdots\mathrm{aaab} = \cdots 0001$。密文區塊就是加密矩陣 M 的列向量。
4. 選擇密文攻擊法: 至於選擇密文攻擊法, 則採用跟選擇明文攻擊法完全一樣的策略來進行, 但明文與密文的角色對調。如此所得到的明文區塊將會是加密矩陣 M 之反元素 (即解密矩陣) 的列向量。

九. 代換密碼(Substitution Ciphers)

代換密碼是更大衆化的密碼系統之一。常見於週末報紙的謎題專欄。其原理甚簡單: 將每一個字母用另一個 (可能會同一個) 字母來取代, 但不重複。更明確的說, 選取 26 個英文字母的一個排列然後施之於明文即得密文。字與字中間的空白在謎題專欄中通常被保留, 這對解謎題者當然是一大助益; 因爲字的結構是猜測的一大提示 (電腦習題第三題是很好的一個例子)。所以, 爲了增加安全性, 將這些空白省略會比較好。

位移密碼與仿射密碼都是代換密碼的簡單例子。維吉內爾密碼與希爾密碼則不是, 因爲他們是將一個區塊的字母排列成另一個區塊的字母而非一次一個字母。衆所周知, 代換密碼可由頻率分析來破解。然其過程遠比我們所想像的還複雜許多。從下面的例子即可窺見一二。

湯姆士·傑弗遜[8] (Thomas Jefferson) 有一潛在性叛逆的信息要傳遞給班傑明·富蘭克林[9] (Benjamin Franklin)。很明顯地, 他不要英國人截取此信息並知悉其內容, 所以他就用代換密碼器將此信息加密。還好, 富蘭克林知道所用的排列, 所以他僅需逆向推導回去即可得到原信息 (當然, 班傑明是相當聰明的, 所以可能他不用事先知道鑰匙就已經將信息解密了)。

現在假設我們在 1776 年時任職於英格蘭的政府代碼與密碼學校而且上級交辦下來破解此一截取到的信息。

```
In[74]:= lwn="LWNSOZBNWVWBAYBNVBSQWVWOHWDIZWRBBNPBPOOUWRPAWXAWPBWZWMYPOBNPBBNWJPAWWRZSLWZQJBNWIAXAWPBSALIBNXWA\
          BPIRYRPOIWRPQOWAIENBVBNPBPUSREBNWVWPAWOIHWOIQWABJPRZBNWFYAVYIBSHNPFFIRWVVBNPBBSVWXYAWBNWVWAIENBV\
          ESDWARUWRBVPAWIRVBIBYBWZPUSREUWRZWAIDIREBNWIATYVBFSLWAVHASUBNWXSRVWRBSHBNWESDWARWZBNPBLNWRWDWAPR\
          JHSAUSHESDWARUWRBQWXSUWVZWVBAYXBIDWSHBNWVWWRZVIBIVBNWAIENBSHBNWFWSFOWBSPOBWASABSPQSOIVNIBPRZBSIR\
          VBIBYBWRWLESDWARUWRBOPJIREIBVHSYRZPBISRSRVYXNFAIRXIFOWVPRZSAEPRIKIREIBVFSLWAVIRVYXNHSAUPVBSVWWUU\
          SVBOICWOJBSWHHWXBBNWIAVPHWBJPRZNPFFIRWVV";
```

數算各字母出現的次數如下 (總共 520 個字):

W	B	R	S	I	V	A	P	N	O	···
76	64	39	36	36	35	34	32	30	16	···

```
In[75]:= cap="ABCDEFGHIJKLMNOPQRSTUVWXYZ";
         frequency[txt_]:= Array[Function[i,{Count[Characters[txt],Characters[cap][[i]]],Characters[cap][[i]]}],26];
         StringLength[lwn]
         frequency[lwn]//Sort
Out[77]= 520
Out[78]= {{0,G},{1,C},{1,K},{1,M},{1,T},{6,Q},{7,J},{7,L},{8,D},{11,F},{11,X},{13,E},{13,U},{13,Y},{14,H},
          {15,Z},{16,O},{30,N},{32,P},{34,A},{35,V},{36,I},{36,S},{39,R},{64,B},{76,W}}
```

英文字母的頻率在上一章已介紹過, 我們再次把最常出現的九個列在此:

---

[8]傑弗遜(1743-1826) 美國政治家。1801 年被選爲第三任總統, 1804 年連任。曾起草獨立宣言, 主張民權自由, 尊重各州自治, 禁止奴隸買賣。總統卸任後, 致力於教育工作, 並創立維吉尼亞大學。

[9]富蘭克林(1706-1790) 美國科學家、發明家、政治家、出版家、哲學家、音樂家及經濟學家。他設立了北美第一個公共圖書館, 創辦賓夕凡尼亞大學。五十歲以後, 致力於獨立活動, 在獨立戰爭中, 曾當選爲第二屆大陸會議代表, 參與起草獨立宣言。

英文字母最常出現的前九個

e	t	a	o	i	n	s	h	r
.127	.091	.082	.075	.070	.067	.063	.061	.060

因爲次高頻率的 B 與第三高頻率的 R 差距挺大的, 這提供了一個合乎我們理性的信心, 據此來猜測 W, B 有可能就是 e, t。但其他的字母又如何呢? 接下來頻率差距不大的七個 R, S, I, V, A, P, N 除了一個或兩個外, 在某種程度上我們會猜測可能就是明文字母 a, o, i, n, s, h, r。但要判斷那一個對應那一個, 就有些困難了。所以一個陽春的單一字母頻率數算, 不足以成就大事。因此我們必須作二字串的頻率分析。

我們從密文中最常出現的前九個字母, 數算所組合而成之二字串出現的次數, 列表如下:

	W	B	R	S	I	V	A	P	N
W	3	4	12	2	4	10	14	3	1
B	4	4	0	11	5	5	2	4	20
R	5	5	0	1	1	5	0	3	0
S	1	0	5	0	1	3	5	2	0
I	1	8	10	1	0	2	3	0	0
V	8	10	0	0	2	2	0	3	1
A	7	3	4	2	5	4	0	1	0
P	0	8	6	0	1	1	4	0	0
N	14	3	0	1	1	1	0	7	0

位於 W 列 N 行之項目 1 表示二字串 WN 在內文中出現 1 次。位於 N 列 W 行之項目 14 表示二字串 NW 在內文中出現 14 次。

二字串出現頻率的排序, 在上一章也介紹過了, 我們再一次地把最常出現的前 15 個列在此:

- th he in er an re ed on es st en at to nt ha

從這兩個表, 我們馬上得到一個結論:BN 極有可能就是 th。若與上面的猜測 W, B 有可能就是 e, t 合起來考慮, 我們有 $W = e$, $B = t$, $N = h$。

如果我們將上上表延伸至包括低頻率的字母, 我們會看到 W 也和許多其他的字母接觸, 此乃 e 的另一個特性。這幫助我們確認上述的猜測。另外, 透過第二高票的 he 所對應之 NW

在此例中與 WA 並列亞軍的事實; 一方面可確認 W = e, N = h, 另一方面也可猜測 A 很有可能就是 r。到目前爲止, 我們也有

$$\{\mathrm{R},\mathrm{S},\mathrm{I},\mathrm{V},\mathrm{P}\} = \{\mathrm{a},\mathrm{o},\mathrm{i},\mathrm{n},\mathrm{s}\}。$$

母音 a, o, i 傾向於彼此互相排斥。我們若看看 R 列, 我們看到 R 並不常出現在 S, I, A, N 之前。但瞄一下 R 行顯示出 R 頗常出現在 S, I, A 之後。所以, 我們懷疑 $\mathrm{R} \notin \{\mathrm{a},\mathrm{o},\mathrm{i}\}$。因爲 V 在 W=e 之前的次數不少, 但 $\{\mathrm{a},\mathrm{o},\mathrm{i}\}$ 在 e 之前的次數不多, 甚至落在 30 名之外; 所以我們也同樣懷疑 $\mathrm{V} \notin \{\mathrm{a},\mathrm{o},\mathrm{i}\}$。繼續下去, 我們看到最有可能的是

$$\{\mathrm{S},\mathrm{I},\mathrm{P}\} = \{\mathrm{a},\mathrm{o},\mathrm{i}\} \quad 與 \quad \{\mathrm{R},\mathrm{V}\} = \{\mathrm{n},\mathrm{s}\}。$$

字母 n 的特性是, 大約有 80% 在 n 之前的字母爲母音。看看前 30 名之內的二字串中 in, an, on, en 分別位居第 3, 5, 8, 11 名, 即可感受一二。目前我們已經認出 W, S, I, P 是母音, 所以滿足上述性質最有可能的字就屬 R 與 A 了, 但上面提到過 A 很有可能是 r, 如此一來 R 就是 n 的最佳候選人。因此之故, 我們也附帶得到了 V=s, 因爲 $\{\mathrm{R},\mathrm{V}\} = \{\mathrm{n},\mathrm{s}\}$。

最後, $\{\mathrm{S},\mathrm{I},\mathrm{P}\} = \{\mathrm{a},\mathrm{o},\mathrm{i}\}$ 當中如何分辨那一個是那一個呢? 先看 o, 我們知道 to 遠比 ot 多得多。比較 BS, BI, BP 及 SB, IB, PB 出現的次數, 不難看出 S=o。再看 i, 我們知道 in 遠比 ni 多得多。比較 IR, PR 及 RI, RP 出現的次數, 得知 I=i, 因而 P=a。所以, 經過合理猜測而決定的前 9 個字母分別如下:

$$\mathrm{W} = \mathrm{e},\ \mathrm{B} = \mathrm{t},\ \mathrm{R} = \mathrm{n},\ \mathrm{S} = \mathrm{o},\ \mathrm{I} = \mathrm{i},\ \mathrm{V} = \mathrm{s},\ \mathrm{A} = \mathrm{r},\ \mathrm{P} = \mathrm{a},\ \mathrm{N} = \mathrm{h}。$$

這些字母在內文 520 個字當中佔有 382 個。

```
LehoOZ these trYths to QeseOHeDiZent that aOOUen
```

在這個時候, 關於這個語言的知識、中間頻率的字母 (l, d, $\cdots$) 及有根據的猜測可用來填滿剩餘的字母。例如, 在第一行的前面一小段中, 一看即知 Y=u 是一好的猜測, 因爲眞理 truth 出現在眼前。當然, 仍有許多猜測的工作以及各式各樣的假設需要測試, 直到一切都沒問題才行。

到此爲止, 我們已將此方法的精神傳達清楚, 所以就略過剩下的細節。解密之後的信息將空格恢復, 如下所示:

We hold these truths to be self-evident, that all men are created equal, that they are endowed by their Creator with certain unalienable Rights, that among these are Life, Liberty, and the pursuit of Happiness.

That to secure these rights, Governments are instituted among Men, deriving their just powers from the consent of the governed.

That whenever any Form of Government becomes destructive of these ends, it is the Right of the People to alter or to abolish it, and to institute new Government, laying its foundation on such principles and organizing its powers in such form, as to seem most likely to effect their Safety and Happiness.

來自美國獨立宣言 [5] (the Declaration of Independence) 內文中間的一段。

十. 福爾摩斯與跳舞的人

密碼與文學

密碼術在多處文學的舞台上出現過她的芳蹤。這個舞台當然不會是武俠小說, 因爲武俠小說中的人物都太厲害了, 密碼術派不上場。在居勒·凡爾納[10] (Jules Verne) 的地心之旅 (Voyage au centre de la Terre, 1864), 引發這一趟偉大旅程的就是一張滿是古冰島文字的羊皮紙被破解的結果。在英國, 最有名的偵探小說家亞瑟·柯南道爾 (Sir Arthur Conan Doyle) 所寫的福爾摩斯探案全集, 其中「歸來記」裡的「跳舞的人」就是一篇以密碼爲主題的短篇小說。在大西洋彼岸, 美國大文豪艾德格·愛倫坡 (Edgar Allan Poe) 也對密碼分析學產生興趣。他寫了一篇與密碼有關的短篇小說金甲蟲 (The Gold Bug)。這篇小說廣受密碼專家的禮讚, 稱之爲最佳密碼創作文學。另外還有威廉·契可瑞 (William Thackeray) 的亨利·艾斯曼的歷史 (The History of Henry Esmond) 及阿嘎他·克里斯堤 (Agatha Christie) 的四嫌疑犯 (The Four Suspects) 也值得一看。

現在我們扼要地來看一下福爾摩斯在「跳舞的人」中破解一密碼系統所展現的智慧與才華。此處僅提及與密碼術有關的情節, 欲知詳情, 請上網 [2] 閱讀其全文, 約有 20 頁, 花兩個小時即可欣賞完畢。也可觀賞由 Simon & Schuster 公司所拍成的影片, 片長才 52 分鐘 [10]。

跳舞的人

一年前才與艾爾西·帕翠克 (Elsie Patrick) 結爲連理的希爾頓·丘比特 (Hilton Cubitt) 寫了一封信給福爾摩斯, 信內附了一張紙條, 在紙上橫著畫了些在跳舞的奇形怪狀的小人, 這是丘比特在他的花園日晷儀上找到的, 如圖 1 所示。

---

[10]凡爾納(1828-1905) 法國十九世紀後半葉的著名小說家, 寫過許多科幻小說, 如從地球到月球、海底兩萬里、神秘島以及環遊世界八十天等。

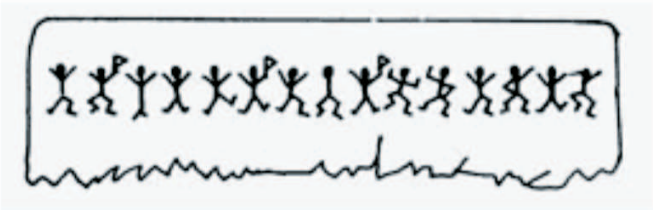


圖1. 艾爾西一看上面這張紙條, 立刻昏倒了。

艾爾西一看上面這張紙條, 立刻昏倒了。之後她就像在做夢一樣, 精神恍惚, 眼睛裡一直充滿了恐懼。就在那個時候, 丘比特寫了上面那一封信給福爾摩斯並於次日到倫敦拜訪福爾摩斯。在這之前一星期左右, 丘比特第一次發現在一個窗臺上畫了一些跳舞的滑稽小人, 跟那張紙上的一模一樣, 是粉筆畫的, 可惜沒有保留就擦掉了。跟福爾摩斯會面後的隔天早上, 發現另一系列跳舞的小人用粉筆寫在工具房門上, 如圖 2 所示。

圖2. 用粉筆寫在工具房門上的另一系列跳舞的小人。

過了兩個早上, 又出現了新的, 如圖 3 所示。

圖3. 出現過三次的第三系列跳舞的小人。

三天後, 在日晷儀上找到一張紙條, 很潦草地畫了一行小人, 跟上一次的完全一樣。那夜在工具房門上又有人畫了一行跳舞的人, 排列跟前兩次的完全相同。隔天早上那扇門除了已經有過的那行小人外, 又添了幾個新畫的, 如圖 4 所示。

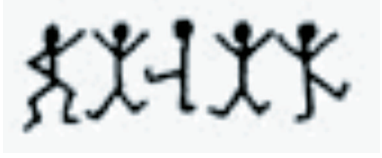


圖4. 在工具房門上除已有的那行小人外, 又添了幾個新畫的。

圖 2 到圖 4 是第二次拜訪福爾摩斯時提供的。當然福爾摩斯心裡是十分興奮。丘比特的背影一消失, 他就急急忙忙跑到桌邊, 把所有的紙條都擺在自己面前, 開始進行繁瑣的分析。一連兩個小時他把畫著小人和寫上字母的紙條, 一張張來回掉換。他全神貫注在這工作上, 完全忘了華生就在旁邊。順手的時候, 便一會兒吹口哨, 一會兒哼著小調; 有時給難住了, 就好一陣子皺著眉頭、兩眼發呆。最後, 他滿意地叫了一聲, 從椅子上跳起在屋裡走來走去, 不住地搓著兩隻手。顯然已有重大的突破。後來, 他打了一通很長的電報給某人, 然後就等著回電。但遲遲不見回電, 如此耐著性子等了兩天。在這兩天裡, 只要門鈴一響, 福爾摩斯就側著耳朵聽。第二天晚上, 來了一封信, 丘比特說他家裡平靜無事, 只是那天清早又看到一長行跳舞的人畫在日晷儀上。他臨摹了一張, 附在信裡寄了來, 如圖 5 所示。

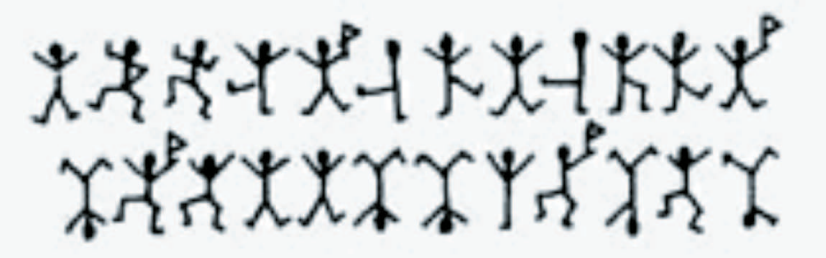


圖5. 福爾摩斯伏在桌上, 對著這張怪誕的圖案看了幾分鐘, 猛然站起來, 發出一聲驚異、沮喪的喊叫。焦急使他臉色憔悴。

福爾摩斯伏在桌上, 對著這張怪誕的圖案看了幾分鐘, 猛然站起來, 發出一聲驚異、沮喪的喊叫。焦急使他臉色憔悴。接著對華生表示他們應該盡快趕到馬場村莊園 (Riding Thorpe Manor)。過沒多久他所盼著的電報來了, 看完後並表示急需讓丘比特知道目前的情況, 多耽誤一分鐘都不應該, 因爲這位諾福克的糊塗紳士已陷入了危險的羅網。隔天一早, 他與華生抵達馬場村莊園時, 發現警察已在那兒了。丘比特先生已中彈身亡, 而他太太艾爾西也中彈且情況相當危險。福爾摩斯問了一些問題後, 就叫人送了一則短訊息給附近艾爾裡奇斯 (Elriges) 農場的阿貝·斯蘭尼 (Abe Slaney) 先生。

處理完這一切, 他們等著犯人前來的空檔, 福爾摩斯就向華生與警長解釋他如何破解那幾張畫著滑稽小人的紙條。他說道: 在我面前擺著的就是這些罕見的作品, 要不是它們成了這麼一場悲劇的前兆, 那麼誰見了也會一笑置之。我比較熟悉各種形式的秘密文字, 也寫過一篇關於這個問題的粗淺論文, 其中分析了一百六十種不同的密碼。但是這一種我還是第一次見到。想出這一套方法的人, 顯然是爲了使別人以爲它是隨手塗抹的兒童畫, 看不出這些符號傳達的資訊。然而, 只要看出這些符號是代表字母, 再應用秘密文字的規律來分析, 就不難找到答案。在交給我的第一張紙條上那句話很短, 我只能稍有把握假定圖 6 代表 E。你們也知道, 在英文字母中 E 最常見, 它出現的次數多到即使在一個短的句子中也是最常見的。第一張紙條上的十五個符

號, 其中有四個完全一樣, 因此把它看成 E 是合理的。這些圖形中, 有的還帶一面小旗, 有的沒有小旗。從小旗的分佈來看, 帶旗的圖形可能是用來把這個句子分成一個個的單詞。我把這看作一個可以接受的假設, 同時記下 E 是用圖 6 來代表的。

圖6. 這就是 E

可是, 現在最難的問題來了。因爲, 除了 E 以外, 英文字母出現次數的順序並不很清楚。這種順序, 在平常印出的一頁文字裡和一個短句裡, 有可能正好相反。大致說來, 按出現次數其順序爲

T, A, O, I, N, S, H, R, D, L;

但是 T, A, O, I 出現的次數幾乎不相上下。要是把每一種組合都試一遍, 直到得出一個意思來, 那會是一項沒完沒了的工作。所以, 只好等新材料來了再說。丘比特先生第二次來訪的時候, 果眞給了我另外兩個短句和似乎只有一個單詞的話, 就是這幾個不帶小旗的符號。在這個由五個符號組合的單字中, 我找出了第二和第四個都是 E。這個單詞可能是 sever(切斷), 也可能是 lever(槓桿), 或者 never(決不)。毫無疑問, 使用末了這個詞來回答一項請求的可能性極大, 而且種種情況都表明這是丘比特太太寫的答復。假如這個判斷正確, 那現在就多了三個符號分別代表 N, V 和 R。然而此時我的困難仍很大, 但一個很妙的想法使我知道了另外幾個字母。我想如果這些懇求是來自一個在丘比特太太年輕時就跟她親近的人, 那麼一個兩頭是 E, 當中有三個別的字母的組合很可能就是 ELSIE 這個名字。我一檢查, 發現這個組合曾經三次構成一句話的結尾。這樣的一句話肯定是對 ELSIE 提出的懇求。這一來我就找出了 L, S 和 I。可是, 究竟懇求什麼呢? 在 ELSIE 前面的一個詞, 只有四個字母, 末了是 E。這個詞必定是 COME(來)。我試過其他各種以 E 結尾的四個字母, 都不符合情況。這樣我就找出了 C, O 和 M, 而現在我可以回頭再分析第一句話, 把它分成單詞, 還不知道的字母就用點代替。經過這樣的處理, 這句話就變成:

· M · ERE ·· ESLNE ·。

現在, 第一個字母只能是 A。這是最有幫助的發現, 因爲在這短句中出現三次。第二個詞開頭是 H 也是顯而易見的。這句話現在變成了:

AMHEREA· ESLANEY。

再把名字中所缺的字母添上:

AMHEREABESLANEY。(我來了, 阿貝·斯蘭尼。)

現在有了這麼多字母, 我能夠很有把握地解釋第二句話了。這一句讀出來是這樣的:

A· ELRI · ES。

我看這一句中, 我只能在缺字母的地方加上 T 和 G 才有意義 (意爲住在艾爾裡奇斯 Elriges), 並且假定這名字是寫信人住的地方或旅店。馬丁警長和華生很有興趣的聽著福爾摩斯詳細講他如何找到答案的經過, 這解答了他們所有的疑問。「後來你怎麼辦, 先生?」警長問。「我有充分理由猜想阿貝·斯蘭尼是美國人, 因爲阿貝是美國式的編寫, 而這些麻煩的起因又是從美國來的一封信。我也有充分理由認爲這件事帶有犯罪的內情。女主人說的那些暗示她過去的話和她拒絕把實情告訴她丈夫, 都使我往這方面去想。所以我才給紐約警察局一個叫威爾遜·哈格裡夫的朋友發了一個電報, 問他是否知道阿貝·斯蘭尼這個名字。他的回電說: 此人是芝加哥最危險的騙子。就在我接到回電的那天晚上, 丘比特寄來了阿貝·斯蘭尼最後畫的一行小人。用已知道的這些字母譯出來就成了這樣的一句話:

ELSIE· RE · ARETOMEETTHYGO ·。

再添上 P 和 D, 這句話就完整了 (意爲, 艾爾西準備見上帝), 這說明了這個流氓已從勸誘改爲恐嚇。對芝加哥的那幫歹徒我很瞭解, 所以我想他可能會很快把恐嚇的話付諸行動。於是和華生醫生立刻趕來諾福克, 但不幸的是, 我們趕到這裡的時候, 最壞的情況已經發生了。

福爾摩斯一解釋完, 警長就急著要帶人去艾爾裡奇斯 (Elriges) 農場即刻展開逮捕斯蘭尼的行動。福爾摩斯說沒有這個必要, 斯蘭尼很快就會自己送上門來。果眞如福爾摩斯所說的, 斯蘭尼來了, 但一進門馬上就被警長帶上手銬。在等待被押走的時候, 斯蘭尼坦承認罪 (但聲稱他開槍乃是自衛), 並說到這種秘密文字是艾爾西的父親發明給他們在芝加哥的幫派『聯幫』所使用的。斯蘭尼已經和艾爾西訂過婚, 但艾爾西無法容忍他們幫派世界的行當, 於是她就趁他們都不防備的時候溜走逃到倫敦。斯蘭尼最後終於找到了艾爾西的住處, 並送出秘密訊息。奇怪的是爲何斯蘭尼會走進福爾摩斯所設好的圈套呢? 福爾摩斯所寫的信息如圖 7 所示。從前面所推導出的字母, 你會發現它的意思不過是馬上到這裡來 (COME HERE AT ONCE)。斯蘭尼當然會確信這必定是從艾爾西來的, 因爲在他們幫派之外不會有人知道這種秘密文字的書寫。因此, 他就來了並走進這個羅網。

圖7. 爲什麼斯蘭尼會走進福爾摩斯所設好的圈套呢?

評論

福爾摩斯雖然以非常少的資料就可完成任務, 但他所做的其實就是破解一簡單的代換密碼系統而已。跟多數此類的密碼系統一樣, 頻率分析及對該語言的知識兩者都非常有用。好運氣當然也不錯, 不管是幸運的猜測或是具有良好的字母分佈都好。注意到 E 是如何勢不可當的成爲最常出現的字母。實際上, 在前四個信息共 38 個字母中, E 就佔了 11 席之多。這給了福爾摩斯一個好的開始。

認證在密碼術中是相當重要的一環。假如五爺破解了三毛的密碼系統, 那麼五爺就能經常僞裝成三毛來與四郎通訊。所以採取預防措施是重要無比的。法官帶給了斯蘭尼許許多多的時間足以來好好思考這方面的議題。

聰明而又機敏的你, 或許已注意到在解密的過程中, 我們有一點在作弊。同一符號代表 NEVER 中的 V 也代表 PREPARE 中的 P。據推測這有可能是誤印且發生在每一個印刷過的版本當中, 甚至可追溯到該小說於 1903 年的第一次印刷。倘若這錯誤發生在福爾摩斯身上, 那麼他的解密過程就會更加艱辛, 而且有可能馬上得到一個結論:『聯幫』需要錯誤更正的技術來傳達他們的信息。實際上, 某種型態的錯誤更正技術應該與大部分的密碼協定同時使用才行。

十一. 二進位數與ASCII

衆所週知, 在電腦的世界中, 用 0 與 1 的字串來表示資料遠比用英文字母及數字來得自然些。數字可以轉換爲二進位數。標準方式是以 10 爲底來表示一個數。例如, 711 表示 $7 \times 10^2 + 1 \times 10^1 + 1 \times 10^0$。二進制用 2 來代替 10 且僅需數字 0 與 1。例如,1011000111 表示 $1 \times 2^9 + 0 \times 2^8 + 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$ (這等於十進制中的 711)。每一個 0 或 1 就稱爲一個二進位數字 (binary digits), 簡稱位元 (bit)。用到八個位元來表示的數就稱爲一個八位元數 (8-bit number), 或一個位元組 (byte)。最大的八位元數爲 255, 而最大的十六位元數爲 65535。

我們經常要處理的東西不僅僅是數字而已。在這種情形之下, 所有的符號、字母與數字就得先轉換成二進制數字。有許多不同的方式可以來完成這件工作, 其中之一就是所謂的美國標準資訊交換碼 (American Standard Code for Information Interchange), 簡稱 ASCII(念成 ass-key)。每一個符號用一個 7 位元的數表示之, 這一來就有 128 個可能的符號。但電腦所普遍使用的是 8 位元, 因此之故, 一個符號經常就用 8 位元來表示。第 8 個位元可用來當成傳輸時的錯誤更正碼, 或經常拿來擴張表列之符號包括像 $\ddot{u}$ 與 $\grave{e}$ 等之符號。

這裡就是 ASCII 符號表。我們不會用到, 之所以放在此處, 只是想讓你看一下文字是怎麼編碼成爲一個 0 與 1 所組成的序列。

```
  0 NUL   1 SOH   2 STX   3 ETX   4 EOT   5 ENQ   6 ACK   7 BEL
  8 BS    9 HT   10 NL   11 VT   12 NP   13 CR   14 SO   15 SI
 16 DLE  17 DC1  18 DC2  19 DC3  20 DC4  21 NAK  22 SYN  23 ETB
 24 CAN  25 EM   26 SUB  27 ESC  28 FS   29 GS   30 RS   31 US
 32 SP   33  !   34  "   35  #   36  $   37  %   38  &   39  '
 40  (   41  )   42  *   43  +   44  ,   45  -   46  .   47  /
 48  0   49  1   50  2   51  3   52  4   53  5   54  6   55  7
 56  8   57  9   58  :   59  ;   60  <   61  =   62  >   63  ?
 64  @   65  A   66  B   67  C   68  D   69  E   70  F   71  G
 72  H   73  I   74  J   75  K   76  L   77  M   78  N   79  O
 80  P   81  Q   82  R   83  S   84  T   85  U   86  V   87  W
 88  X   89  Y   90  Z   91  [   92  \   93  ]   94  ^   95  _
 96  `   97  a   98  b   99  c  100  d  101  e  102  f  103  g
104  h  105  i  106  j  107  k  108  l  109  m  110  n  111  o
112  p  113  q  114  r  115  s  116  t  117  u  118  v  119  w
120  x  121  y  122  z  123  {  124   125  }  126  ~  127 DEL
```

十二. 單次鑰匙簿密碼(One -Time Pads)

這個無法破解的密碼系統在 1919 年由吉伯特·先得福·維念 (Gilbert Sandford Vernam) 取得美專利權 1310719 號 [13]。將明文信息透過二進制或 ASCII 表示成由 0 與 1 所成的序列。但信息也有可能是一個數位化影像或是一個聲音訊號。

鑰匙是一由 0 與 1 所構成的隨機序列, 其長度與明文信息相等, 用過隨即丟棄且絕對不再使用。加密運算就是將鑰匙加到信息上, 一個位元接著一個位元的在模 2 之下進行。此過程通常稱之爲 XOR。換句話說, 我們所使用的規則如下:$0+0=0,\quad 0+1=1+0=1,\quad 1+1=0$。例如明文信息是 0010010101 而鑰匙爲 1001001010, 則得到密文如下:

$$\begin{array}{ll}\text{(明文)} & 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\\ \text{(鑰匙)} & 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\\ \text{(密文)} & 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 1\end{array}$$

解密動作與加密完全一樣, 用同一個鑰匙加到密文即可, 如下:

$$\begin{array}{ll}\text{(密文)} & 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 1\\ \text{(鑰匙)} & 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\\ \text{(明文)} & 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\end{array}$$

當然明文及鑰匙也可回到原先的字母所構成的序列。此時的單次鑰匙簿密碼其實就是一種維吉內爾密碼, 其鑰匙長度與明文信息相等。不同的是此鑰匙的選擇是完全隨機的, 因而密文也保有此隨機性, 如此才能成就其無法破解的特性。

對密文攻擊法而言, 這個加密的方法是完全無法破解的。例如若密文爲 LPRASQOBDQRCZXKE, 則明文可能是 iloveyouverymuch, 也有可能是 wonderfulweekend; 只要是相同長度的信息都有相同的可能性是明文。因此, 除了長度外, 從密文看不出任何明文的蛛絲馬跡。這在解藍恩 (Shannon) 的 Entropy 理論中可以講得更明確。

如果我們有明文中的片段, 那麼就可以找出此片段中所對應的鑰匙, 但對其他片段的鑰匙卻毫無用處可言。在大多數的情況中, 不管是選擇明文攻擊或是選擇密文攻擊都不太可能。即使可能也只透露出所知道的那個片段的鑰匙而已, 這對破解信息毫無助益, 除非此片段鑰匙又被再用一次。

那麼我們該如何使用這麼樣的一個系統呢? 而且在何種情況之下使用呢? 鑰匙當然可以事先產生, 但問題是如何製造出眞正隨機的 0 與 1 之序列呢? 你可叫阿貓阿狗一起來丟銅板, 並且邊哼著『丟丟銅子』的民謠, 免得太無聊; 然而這種方法在大部分的場合是太慢了, 而且不實用。我們也可請出蓋革計數器, 並計算在一小週期內喀嚓了幾聲; 若是偶數就記成 0, 若是奇數就記成 1。有一些其他實際可行的方法, 速度較快但隨機性就差一些; 但不難看出, 要快速產生一把好的鑰匙是困難的。一旦鑰匙出爐, 經由信得過的密使交給接收方。這樣一來, 有需要的時候, 就可以傳送信息了。據報導, 在冷戰 (Cold War) 期間, 華府與克里姆林宮頭頭之間的『熱線 (Hot Line)』就是採用單次鑰匙簿密碼系統來互通信息。

單次鑰匙簿密碼的缺點是它需要一很長的鑰匙, 不僅製造昂貴而且傳送也昂貴。一旦鑰匙用完了, 若再使用於第二個信息, 那就危險萬分; 因爲任何由第一個信息所得到的知識, 都會反映到第二個信息當中。所以在大部分的情況裡, 有好幾個方法只需少許的輸入即可用來產生一合理的 0 與 1 之隨機序列, 因而可看成是一近似的單次鑰匙簿。如此一來, 信差特使所攜帶的資訊量就比原先要送出去的信息少了好多。下面我們描述一個速度快, 但不怎麼安全的這一類的方法。

十三. 線性回饋位移暫存器序列

在涉及加密的許多場合中, 有一介於速度與安全性之間的平衡交易。若你要求很高水準的安全性, 那就得犧牲速度, 反之亦然。例如在有線電視方面, 許多的資訊要傳達, 所以速度是重要的; 至於安全性就沒那麼重要, 因爲不值得投下昂貴的設備來攻擊此一系統。

在這裡我們描述一個速度比安全性重要的方法。下面的序列 s

$$0100001001011001111100011011101010000100101100111111$$

可由其起始值 $k = \{0, 1, 0, 0, 0\}$

$$x_1 = 0, \quad x_2 = 1, \quad x_3 = 0, \quad x_4 = 0, \quad x_5 = 0$$

及下面的線性遞迴關係式得到:

$$x_{n+5} = x_n + x_{n+2} \pmod 2。$$

這個序列在第 31 項後開始重複。

更一般地, 考慮一長度爲 m 的線性遞迴關係式

$$x_{n+m} \equiv c_0 x_n + c_1 x_{n+1} + c_2 x_{n+2} + \cdots + c_{m-1} x_{n+m-1} \pmod 2,$$

此處係數 $c = \{c_0, c_1, \ldots, c_{m-1}\}$ 爲 0 或 1。若我們指定起始值爲

$$k = \{x_1,\ x_2,\ x_3,\ \cdots,\ x_m\}$$

則所有接下去的 x_n 值可由遞迴關係式求出。我們定義如下指令來執行製造此等所謂的 LFSR 序列 (理由待會兒說明) 之任務:

- `lfsr[c,k,n]` 將係數爲 `c` $= \{c_0, c_1, \ldots, c_{m-1}\}$ 之遞迴關係式, 在起始值 `k` $= \{x_1, x_2, x_3, \ldots, x_m\}$ 之下所生成的序列, 輸出其前面 n 項。

上述指令之程式如下:

```
In[79]:=lfsr[c_,k_,n_]:=Module[{z},z=k;Do[AppendTo[z,Mod[Array[Function[i,z[[j-Length[k]-1+i]]],Length[k]].c,2]],
                                                                                {j,Length[k]+1,n}];z];
```

用此指令來驗證上述序列 s 如下:

```
In[80]:= c5={1,0,1,0,0};k5={0,1,0,0,0};
         s={0,1,0,0,0,0,1,0,0,1,0,1,1,0,0,1,1,1,1,1,0,0,0,1,1,0,0,1,0,1,1,1,1,0,0,0,1,1,0,1,0,1,0,0,0,0,1,0,0,1,0,1,1,0,0,1,1,1,1,1};
         lfsr[c5,k5,Length[s]] == s
Out[82]= True
```

如此所得到的由 0 與 1 組成的序列可用來當加密之用的鑰匙。將明文寫成由 0 與 1 組成的序列, 然後在模 2 之下把鑰匙中適當的位元, 一個位元接著一個位元地加在明文上。例如明文信息是 1011001110001111 而鑰匙序列爲上面的序列, 我們有

$$\begin{array}{lcccccccccccccccc} \text{(明文)} & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \text{(鑰匙)} & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ \text{(密文)} & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \end{array}$$

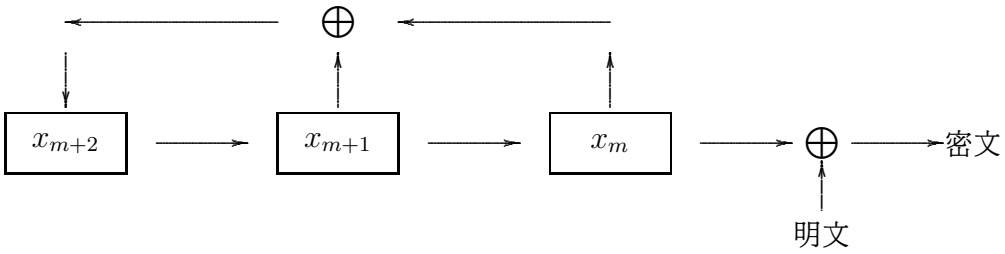
解密則與加密完全一樣, 將鑰匙序列加在密文上即可完成。

此法的一個優點是大的鑰匙週期僅需用到很少的資訊。長週期對維吉內爾密碼是一項改良, 因爲短週期一下子就會被找到鑰匙。在上例, 指定起始向量 $\{0, 1, 0, 0, 0\}$ 及係數 $\{1, 0, 1, 0, 0\}$ 得到一週期爲 31 的序列。所以 10 個位元可用來製造 31 個位元的序列。同樣地, 可證明遞迴關係式

$$x_{n+31} = x_n + x_{n+3}$$

及任何非零起始向量產生一序列, 其週期爲 $2^{31}-1=2147483647$。因此 62 個位元可用來製造超過 20 億位元的鑰匙。這是凌駕於單次鑰匙簿的一大優點, 因爲單次鑰匙簿必須事先傳送 20 億位元的鑰匙。

這個方法在硬體方面非常容易付諸行動, 就是所謂的線性回饋位移暫存器 (Linear Feedback Shift Register), 簡稱爲 LFSR, 而且速度非常快。這也是爲什麼我們將此等序列稱之爲 LFSR 序列的緣由。考慮下圖:



對一暫存器的下一階段, 在每個框框中的位元按箭頭的方向移動至另一框框中, 如圖示。其中 $\bigoplus$ 表示模 2 之下的加法, 圖中指的是 x_m+x_{m+1}。輸出之位元 x_m 與下一個明文的位元相加產生密文。上圖代表遞迴關係式 $x_{n+3}=x_n+x_{n+1}$, 一旦給定起始值 $\{x_1,x_2,x_3\}$ 此機器就非常有效率的產生接下去的位元。

不妙的是, 上述的加密法很容易就會被已知明文攻擊法破解。更明確的說, 若我們只知道幾個連續的明文及其對應的密文, 則我們可用此來決定其遞迴關係式, 因而算出此鑰匙接下去的位元。

將明文從密文中減掉 (或加上, 在模 2 之下是一樣的), 即得鑰匙之位元。因此在下面的討論, 我們不管密文及明文是什麼; 僅專注在鑰匙並且假設鑰匙序列的片段已被發現。

例題01: 已知週期爲 15 之序列 $t=011010111100010011010111\cdots$, 其起始片段爲 011010111100 且已知是從一個線性遞迴關係式所產生的。如何決定此遞迴關係式的係數? 我們不見得知道其長度, 所以就從長度 2 開始 (長度 1 爲常數序列)。假設遞迴關係式爲 $x_{n+2}=c_0x_n+c_1x_{n+1}$。令 $n=1,2$, 用已知數值 $x_1=0,x_2=1,x_3=1,x_4=0$ 得方程組並寫成矩陣的形式如下:

$$\begin{pmatrix}0&1\\1&1\end{pmatrix}\begin{pmatrix}c_0\\c_1\end{pmatrix}=\begin{pmatrix}1\\0\end{pmatrix}。$$

解之得 $c_0=1$ 與 $c_1=1$, 所以猜測遞迴關係式爲 $x_{n+2}=x_n+x_{n+1}$。很遺憾的是, 這

個猜測是不正確的, 由於 $x_6 \neq x_4 + x_5$。因此我們就試長度等於 3, 得到矩陣方程式

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}。$$

這個係數矩陣的行列式值爲 $0 \pmod 2$; 實際上, 此方程式無解。這也可透過觀察就可看出其無解的性質, 因爲左邊矩陣各行元素和爲 0 但右邊向量則否。現在考慮長度等於 4, 得到矩陣方程式

$$\begin{pmatrix} 0\,1\,1\,0 \\ 1\,1\,0\,1 \\ 1\,0\,1\,0 \\ 0\,1\,0\,1 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix},$$

解之得到 $c_0 = 1, c_1 = 1, c_2 = 0, c_3 = 0$。

```
In[83]:= A={{0,1,1,0},{1,1,0,1},{1,0,1,0},{0,1,0,1}}; B={{1}, {0}, {1}, {1}};
         PolynomialMod[Inverse[A].B, 2]
Out[84]= {{1}, {1}, {0}, {0}}
```

所以我們猜測可能的遞迴關係式爲 $x_{n+4} = x_n + x_{n+1}$; 又此遞迴關係式生成已知鑰匙片段的其餘元素, 因而這是從現有資訊所得到之遞迴關係式的最佳猜測。實際上, 很快的算一下, 可以確知這的確就是了。所以我們已經找到所要的遞迴關係式。用指令 `lfsr` 來驗證序列 t 如下:

```
In[85]:= c4={1,1,0,0}; k4={0,1,1,0}; t={0,1,1,0,1,0,1,1,1,1,0,0,0,1,0,0,1,1,0,1,0,1,1,1,1};
         lfsr[c4, k4, Length[t]] == t
Out[86]= True
```

在一般的情況如下, 若要求出一個長度爲 m 的遞迴關係式, 我們得先知道前 $2m$ 項 $x_1, x_2, x_3, \ldots, x_{2m}$。如上得到矩陣方程式

$$\begin{pmatrix} x_1 & x_2 & \cdots & x_m \\ x_2 & x_3 & \cdots & x_{m+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_m & x_{m+1} & \cdots & x_{2m-1} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{m-1} \end{pmatrix} = \begin{pmatrix} x_{m+1} \\ x_{m+2} \\ \vdots \\ x_{2m} \end{pmatrix}。$$

此矩陣方程式的係數矩陣、右翼行向量及其解可經由下列三指令分別計算之, 此處 `v` 就是所涉及的 LFSR 序列。對應之程式如下:

```
In[87]:= lfsrmatx[v_,m_]:=Array[Function[{i,j},v[[i+j-1]]],{m,m}]
         lfsrrhs[v_, m_]:=Array[Function[i,v[[i+m]]],m]
         lfsrsoln[v_,m_]:=PolynomialMod[Inverse[lfsrmatx[v,m]].lfsrrhs[v,m],2]
```

驗證上述的 LFSR 序列 t, 當 $m = 4$ 對應的係數矩陣爲 A (在In[20]之中) 及其解如下:

```
In[90]:= {lfsrmatx[t, 4] == A, lfsrsoln[t, 4]}
Out[90]= {True, {1, 1, 0, 0}}
```

下面我們將予以證明這個係數矩陣在模 2 之下是可逆的充分而且必要條件爲 $x_1, x_2, x_3, \ldots, x_{2m-1}$ 不滿足長度小於 m 的任何線性遞迴關係式。這意味著其線性遞迴關係式的長度至少是 m。

因此尋找線性遞迴關係式係數的一個策略清清楚楚的展現在你我眼前。假設我們知道鑰匙序列的前面 100 位元。對 $m = 2, 3, 4, \ldots$, 形成一 $m \times m$ 的矩陣如上並計算其行列式值。若連續有好幾個 m 值, 其對應之矩陣的行列式值爲 0, 那就停止。最後一個具有非零行列式值的 m, 有可能就是遞迴關係式的長度。解此矩陣方程式得到係數 $c_0, c_1, \cdots, c_{m-1}$, 然後再檢查有這個線性遞迴關係式所生成的序列是否與原鑰匙序列中已知的那些位元吻合。若否, 則試大一些的 m 值。利用上述指令 `lfsrmatx[v,m]`, 定義指令 `lfsrlength[v,m]` 來計算前 m 個行列式值, 其程式如下:

```
In[91]:= lfsrlength[v_,m_]:=Table[Mod[Det[lfsrmatx[v,k]],2],{k,Min[m,1+Floor[Dimensions[v][[1]]/2]]}]
```

假設我們不知道鑰匙序列起始的 100 位元, 但卻知道在某處連續的 100 個位元。利用這些位元爲起始點, 執行上述的步驟。實際上, 一旦我們找到遞迴關係式, 我們也可逆向追溯起始點之前的位元。

例題02: 假設我們有 100 位元的 LFSR 序列 u 如下:

```
In[92]:=u={1,0,0,1,1,0,0,1,0,0,1,1,1,0,0,0,1,1,0,0,0,1,0,1,0,0,0,1,1,1,1,1,0,1,1,0,0,1,1,1,1,1,0,1,0,1,0,1,0,0,1,
          0,1,1,0,1,1,0,1,0,1,1,0,0,0,0,1,1,0,1,1,1,0,0,1,0,1,0,1,1,1,1,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,0};
```

從 $m = 1$ 開始, 前 20 個行列式值爲

```
In[93]:= lfsrlength[u, 20]
Out[93]= {1,0,1,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0}
```

一個合理的猜測是 $m = 8$ 乃是具非零行列式值的最後一個。解矩陣方程式, 得到係數爲

```
In[94]:= lfsrsoln[u, 8]
Out[94]= {1, 1, 0, 0, 1, 0, 0, 0}
```

所以我們猜測其遞迴關係式爲

$$x_{n+8} = x_n + x_{n+1} + x_{n+4}$$

這個遞迴關係式生成原序列所有的 100 項, 所以至少根據我們現有的知識, 這是正確的答案。

假設我們知道在某序列中間連續的 100 個位元, 且我們想知道在這之前的位元。例如序列從 x_{17} 開始, 所以 $x_{17} = 1, x_{18} = 0, x_{19} = 0, \ldots$。將遞迴關係式寫成

$$x_n = x_{n+1} + x_{n+4} + x_{n+8}$$

令 $n = 16$ 得到

$$x_{16} = x_{17} + x_{20} + x_{24} = 1 + 1 + 1 = 1$$

如此繼續下去, 我們可依序決定 $x_{15}, x_{14}, \cdots, x_1$。

我們現在證明上面所承諾要證明的結果。

定理: 令矩陣 M 爲

$$M = \begin{pmatrix} x_1 & x_2 & \cdots & x_m \\ x_2 & x_3 & \cdots & x_{m+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_m & x_{m+1} & \cdots & x_{2m-1} \end{pmatrix}$$

若序列 $x_1, x_2, x_3, \ldots, x_{2m-1}$ 滿足一長度小於 m 的線性遞迴關係式, 則 $\det(M) = 0$。反之, 若序列 $x_1, x_2, x_3, \ldots, x_{2m-1}$ 滿足一長度等於 m 的線性遞迴關係式且 $\det(M) = 0$, 則此序列也會滿足一長度小於 m 的線性遞迴關係式。

注意: 首先說明一下定理敍述中關於線性遞迴關係式長度的問題。一個序列可能滿足一長度爲 3 的遞迴關係式, 如 $x_{n+3} = x_{n+2}$。顯然地, 此序列也滿足一長度更短的遞迴關係式, 如 $x_{n+1} = x_n$(至少對 $m \geq 2$)。然而有些序列可能會滿足一長度比所期待者小的遞迴關係式, 如 $x_{n+4} = x_{n+3} + x_{n+1} + x_n$。假設起始值爲 $1, 1, 0, 1$, 則利用遞迴關係式可算出接下去的 12 項爲 $1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, \ldots$。容易看出此序列滿足遞迴關係式 $x_{n+2} = x_n + x_{n+1}$。

證明: 若有一長度小於 m 的線性遞迴關係式, 則矩陣 M 中的一列是其他幾列的線性組合。如遞迴關係式爲 $x_{n+3} = x_n + x_{n+2}$, 則第四列爲第一列與第三列的和。因此行列式值爲 $0 \pmod 2$。

反之, 若行列式值爲 $0 \pmod 2$。則存在一非零向量 $\vec{b} = (b_0, \ldots, b_{m-1})$ 使得 $\vec{b}M = \vec{0}$。這給了我們一遞迴關係式, 但無法馬上看出此遞迴關係式可一路延伸到 x_{2m-1}。如 M 的前面兩列的和等於第三列, 即 $x_{n+2} = x_n + x_{n+1}$ $\forall n = 1, 2, \ldots, m$。我們需將此一路延伸到 $n = 2m - 3$ 得到 $x_{2m-1} = x_{2m-3} + x_{2m-2}$。記得嗎我們也假設有一長度爲 m 的線性遞迴關係式 $x_{n+m} = c_0 x_n + \cdots + c_{m-1} x_{n+m-1}$, $1 \leq n \leq m$ 用此遞迴關係式來定義

$x_{2m},\ x_{2m+1},\ x_{2m+2},\ldots$ 如此繼續延伸此一序列 (當然, 若此序列已有這些項 $x_n,\ n \geq 2m$, 那可能與我們在證明當中所暫時使用的值不一樣)。我們有

$$M_n \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{m-1} \end{pmatrix} = \begin{pmatrix} x_{n+1} & x_{n+2} & \cdots & x_{n+m} \\ x_{n+2} & x_{n+3} & \cdots & x_{n+m+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n+m} & x_{n+m+1} & \cdots & x_{n+2m-1} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{m-1} \end{pmatrix} = \begin{pmatrix} x_{n+m+1} \\ x_{n+m+2} \\ \vdots \\ x_{n+2m} \end{pmatrix}$$

(此處 M_n 就是在式子中間的 $m \times m$ 矩陣)。當 $n = 0$, 這就是原來的矩陣方程式。對較大的 n, 這僅僅表示我們的遞迴關係式。記得向量 $\vec{b}$ 滿足 $\vec{b}M = 0$。在上面方程式中, 令 $n = 0$ 並在兩邊的左方乘上向量 $\vec{b}$。因爲 $\vec{b}M_0 = \vec{b}M = 0$, 左式 $= 0$, 故右式也 $= 0$。這意味著 $\vec{b} \cdot (x_{m+1}, \ldots, x_{2m}) = 0$。

現在考慮 $n = 1$ 的情形。上面已經證明的是 $\vec{b}$ 乘上 M_1 的最後一行等於 0。但是 M_1 的其他行來自 M_0 的行向量, 所以也會被 $\vec{b}$ 所消化掉。因此 $\vec{b}M_1 = 0$。如上, 我們得到 $\vec{b}$ 乘上 M_2 的最後一行等於 0。如此這般, 我們看出 $\vec{b}M_n = 0\ \forall n$。

顯而易見, 這成就了一長度小於 m 的遞迴關係式, 也同時完成了整個定理的證明。

最後, 我們對序列的週期做一些評論。假設一遞迴關係式的長度爲 m。任何此序列中連續的 m 項決定所有後面的元素, 而且將遞迴關係式逆向書寫, 則也可求出所有前面的值。顯然地, 若我們連續有 m 個 0, 則在它之前之後都是 0。因此我們不考慮此種情況。長度爲 m 的不全爲 0 的 0 與 1 所構成的數串總共有 $2^m - 1$ 個。所以, 只要超過 $2^m - 1$ 項, 某一個長度爲 m 的數串必定會出現兩次, 因而此序列會開始重複。其週期頂多是 $2^m - 1$。

伴隨著一個遞迴關係式 $x_{n+m} = c_0x_n + c_1x_{n+1} + c_2x_{n+2} + \cdots + c_{m-1}x_{n+m-1}$, 我們有一多項式定義爲

$$f(T) = T^m - c_{m-1}T^{m-1} - \cdots - c_0。$$

若 $f(T)$ 在模 2 之下是不可分解的, 則可證明其週期整除 $2^m - 1$。一個有趣的情形是當 $2^m - 1$ 爲質數 (亦即梅仙尼質數) 的時候。如果週期不是 1 (亦即不是常數序列) 時, 則其週期一定就是極大值 $2^m - 1$。上面的例子週期爲 $2^{31} - 1$ 就是此種類型。

線性回饋位移暫存器序列已廣泛地被研究過。例如, 可參考所羅門·郭倫 (Solomon W. Golomb) 的位移暫存器序列 [3] (Shift Register Sequences) 或珍·凡德路比 (Jan. C. A. van der Lubbe) 的密碼術的基本方法 [12]。

阻撓上述攻擊的一個方法是採用非線性遞迴關係式, 如

$$x_{n+3} = x_{n+2}x_n + x_{n+1} \pmod 2$$

一般而言, 要破解這些非線性系統稍困難些。不過, 我們不會在此討論。

參考文獻

1. Becker H./Piper, F., *Cipher Systems: The Protection of Communication*, Northwood Books, London, 1982.
2. Doyle, Arthur Conan, *The Adventures of Sherlock Holmes*, The Dancing Men, 1903. http://www.bookhome.net/zhentan/knde/glj-twdr.html
3. Golomb, Solomon, W., *Shift Register Sequences*, Aegean Park Press, Revised Edition, 1982.
4. Hill, Lester S., Cryptography in an Algebraic Alphabet, The American Mathematical Monthly 36, June-July 1929, pp.306-312. http://en.wikipedia.org/wiki/Hill_cipher
5. 美國獨立宣言 http://usinfo.org/docs/deceng.htm
6. Kahn, David, The Codebreakers, The Story of Secret Writing, Scribner, Revised and Updated, 1996.
7. Kerckhoffs, Auguste, La cryptographie militaire, Journal des sciences militaires, vol. IX, pp.5-83, Jan. 1883, pp.161-191, Feb. 1883.
   http://www.petitcolas.net/fabien/kerckhoffs/la_cryptographie_militaire_i.htm
8. Kessler, Gary C.: *Hiding Data in Data*, Windows & .NET Magazine, April 2002.
   http://www.garykessler.net/library/steganography.html
9. Saeednia, Shahrokh, How to Make the Hill Cipher Secure, Cryptologia, 24(4), October 2000, pp.353-360.
10. Simon & Schuster, Inc., The Adventures of Sherlock Holmes, The Dancing Men, Simon & Schuster Video, 1986.
11. Singh, Simon, The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography, 1999.
12. van der Lubbe, Jan C. A., *Basic Methods of Cryptography*, Cambridge University Press, 1998.
13. Vernam, Gilbert S., Cipher Printing Telegraph Systems For Secret Wire and Radio Telegraphic Communications, *Journal of the IEEE*, Vol.55, pp.109-115 (1926).
    http://en.wikipedia.org/wiki/Gilbert_Vernam
14. Wright, Ernest Vincent: Gadsby, A Story of Over 50,000 Words Without Using the Letter "E", 1937. http://spinelessbooks.org/gadsby/

—本文作者任教於東海大學數學系—