自動校正碼理論簡介

楊重駿・楊照崑

一、導 言

近代資訊科學的快速發展，得力於電子計算機的普及與多頻道通訊網的形成，二者相輔相成，缺一不可。若少了快速通訊網，則每個電子計算機必須由鍵盤、磁帶、或磁碟輸入資料，大大的減低了快速的功能，但是若沒有計算機在通訊網的兩端，則快速的通信無法及時消化，亦不能發揮其最大的功能。可惜目前尚未能完全保證通訊中不發生錯誤。由於發生錯誤的原因很多，有自然因素如雷電，有人爲因素如強電流及別的通訊信號所帶來的干擾。本文所要討論的就是如何發現或自動更正信號在通訊中所發生的錯誤。

電子計算機加上通訊網，其所處理的資料之多，信號傳遞之快，已不是人肉眼校對可以勝任得了的，我們必須在計算機通訊的本身加上檢錯的裝置。一個淺而易見的檢錯法就是把要傳送的信號重覆一次，如兩者不合則必定有錯。譬如今有某個被重覆的信號是某人存了一萬元及某人存了一千元，二者不符，至少有一個錯，也許你會說，如果兩次傳遞都發生了同樣的錯，把一萬誤傳成了一千，不就查不出來了嗎？不錯，但且看兩次同時出錯的可能性是多少？假定一次出錯的或然率是 p，則二次同時出錯的或然率是 p^2（註一）。這兒的 p 必須是很小的數量，否則信號老是出錯，改不勝改，則此種機器尚不宜問市或要修理。設 p 是十萬分之一，即 10^{-5}，則 $p^2=10^{-10}$ 即十億分之一的或然率，碰到了這種同時出錯的事，眞是天亡我也。

有時知道有錯還不夠，因一個信號被送兩次而不符合，必是錯了，但却不知誰對誰錯，只好請對方重傳一次，這往往是一個耗時費事的程序，發方必須打斷原有的傳遞，重新找到對方所說傳錯的位置，重新再傳一次。若是「知錯能改」，並不是「善莫大焉」的事嗎？何不每個信號傳三次，取其多數？由或然率看來，每個信號傳三次，是極具有自動更正的功能，但代價是否太大？一個信號傳三次，也許太浪費了吧？有沒有更經濟的方法？

爲了使我們的回答更合於電子計算機通訊，我們就採用它們之間通信的符號，即以數碼 0，1 所組成的數字事來代表信號，像現在電腦中英文通訊所用的ASCII電碼是由七位數碼組成的。例如 a 是 1100001，b 是 1100010，1 是 0110001。因每位數有二個可能性，所以一共可以排出 $2^7=128$ 個不同的電碼，足夠表示英文大小寫字母、標點符號、數字、希

臘字母了。爲了說明簡潔起見，我們今用二個數碼來表示4個信號，即如表一。

表一

信號	表示數碼
a	00
b	01
c	10
d	11

現在看，若是a爲00誤傳成了01，我們會以爲信號是b，無法確定會有錯誤，若我們每個信號傳二次，即a以0000傳，b以0101傳，等等。則如果a的0000錯傳成了0100，則我們就知道出毛病了，但却不知它是由0000或者0101而來，由於二者都只含一個錯誤。當然0100也可能是由1111誤傳而來，但因一個錯誤比多個錯誤的機會小得多，我們最好的選擇是假定只有一個錯誤，這是引用了統計學上常用的最大可能原則（maximum likelihood principle）。它也是我們日常推測事情眞相的原則。譬如我們在地上拾到了一筆錢，我們總會推測這是有人遺失的（要交給警察）而不會認爲是有人故意放一筆錢在地上做好事送給人。但因爲後者的可能性太小，在沒有其它的證據之下，我們不會做後者的推測。

再仔細研究一下，我們之所以知道0100信號有錯，是因爲0100不在對方輸出的碼中，因對方只可以輸出0000，0101，1010，及1111的其中之一，可見若要能檢錯，輸出碼不可以佔用了全部可能的碼字，爲了統一起見，我們作以下的定義。

定義1 凡是用n位0或1作爲一個單元（或數碼）輸出入信號者，稱爲一n位碼（輸送）系統（或簡稱爲碼）。其中任何一個信號，若含有n個數碼，稱之爲一個碼字（注意文中有時將“字”省去）。

定義2 在n位碼系統中，其前m（$m\leq n$）位稱之爲信息碼（或原碼），它是我們所有要傳的信號，而後$t=n-m$位稱爲檢定碼，，是用來檢定或改正錯誤而用。

下文中所謂檢定k位是指說可以知道至多有k位數碼錯了，而校正k位是指說把至多有k位錯的數碼的位置找出來。

以表一及重覆傳送爲例，其前兩位是信息碼，後兩位是檢定碼（見表二），其中$n=4$，$m=2$，$t=2$。

表二

信號	信息碼	輸送碼	檢定碼
a	00	0000	00
b	01	0101	01
c	10	1010	10
d	11	1111	11

用表二的輸送碼，我們已知他們可以檢錯一位，但不能更正錯誤，對檢錯一位的要求而言，我們又不知道這種重覆輸送是否是最經濟的辦法（答案是否定的，我們在第三節會談到），現在，我們可以把我們想解決的命題寫下來了：

對一個原爲m位的信息碼，我們最少要加上多少位的檢定碼t，使之可以自動檢定或更正k位的錯誤？ (1)

在上面的命題中，m與k爲已知數，而t爲待定的數。

要解決這個問題，我們首先把這些碼字用向量的符號表示，並定義兩碼字的距離，然後我們證明當可以輸出之碼字之間都有相當大的距離時，某些位的錯誤就可以檢定或校正。

看本文所需要的預備知識是矩陣的一些基本運算，最後在結論中，我們提出了一些較難的問題，供有興趣的讀者思考或做更深入的研究。

二、如何表示碼字與它們之間的關係

我們延用定義1，2中的符號m，n，t，如果以Z表｛0，1｝之集合，則m位信息碼可以Z^m表示之，即

$$Z^m=\{(x_1, x_2, \cdots, x_m),\ x_i=0 \text{ 或 } 1\} \tag{2}$$

其中一共有2^m個元素。同樣我們用Z^n表示所有n位碼字之集合，則製碼法就是把一個Z^m中的元素鑲到Z^n中去的函數，以表二爲例，重傳就是把Z^2中的元素(a_1, a_2)鑲成Z^4中的元素(a_1, a_2, a_1, a_2)，以一般函數符號表示，則

$$f=Z^m \rightarrow Z^n \tag{3}$$

表示一種製碼法。令A爲f之區域，即

$$A=f(Z^m)=\{f(x) \mid x\in Z^m\}$$

則很顯然的$A\subset Z^n$，我們又稱A爲輸出碼表，而且在我們表示向量$(x_1, x_2, \cdots, x_m)$，有時不加逗點，即在表二中的b，我們可以寫成(0, 1, 0, 1)或(0101)。

定義3 在(3)中的製碼法如果在收到的n位碼字中有k位或k位以下的錯誤時，能知道收到的碼字有錯，則此種製碼法稱之爲有檢定k位錯誤的功能，若其能校正此k位錯誤，則稱之有校正k位錯誤之功能。

例如表二中的輸送碼，有檢定一位錯誤之功能，但沒有校正一位之功能。

定義4 （漢明（Hamming）距離），設X，Y爲Z^n中的二元素，則X與Y之間的距離定爲$H(X, Y)=X$，Y中不相同的數碼之數目。

例如：$X=(1010)$，$Y=(1001)$，則因X與Y有二位（第三、四位）不相同，故$H(X, Y)=2$，若令$0=(0, 0, \cdots 0)\in Z^n$，即全含0的元素，則

$$H(X, 0)=X\text{中所含1的數目}$$

，並以$|X|$表示之。

定義5 令$X=(x_1, x_2, \cdots, x_n)$

$Y=(y_1, y_2, \cdots, y_n)$

則定向量之加減，

$$X\pm Y=(x_1\pm y_1, x_2\pm y_2, \cdots, x_n\pm y_n) \tag{4}$$

且在上式中，$1\pm 1=0\pm 0=0$，

$1\pm 0=0\pm 1=1$。

也就是說，如果x_i與y_i相同，則它們的差爲0，否則爲1。

由此很容易看出X，Y的距離

$$H(X, Y)=|X-Y| \tag{5}$$

一些不熟悉抽象數學的讀者也許會覺得$1+1=0$很怪，但如我們抽象地把0，1分別相應“偶”、“奇”兩個概念，就不會驚訝了。且我們只有0與1兩個數，因此$1+1$只有等於0或1兩個選擇，若令$1+1=1$，則會造成$1=0$的結果，二元變成了一元，一些主要的演算特性都消失了，因此只好定義$1+1=0$，而熟悉抽象數學的人知道這樣的定義加上一般的乘法

$$0\cdot 1=1\cdot 0=0\cdot 0=0,\quad 1\cdot 1=1 \tag{6}$$

使得｛0，1｝成爲一個域，這樣上面定的向量集合就自然形成一個向量空間了。

由上面的運算規則，很容易可以到及瞭解下面一些定理及定義。

定理1：令X，Y，Z爲Z^n中之任意三元素，則

(i) $H(X,Y)=|X+Y|$
$=|X-Y|$
$=H(Y,X)$

(ii) $X+X=X-X=0$
注意由此可得$X=-X$的一重要事實。

(iii) 若$X\neq Y$，則$X\pm Y\neq 0$

(iv) $H(X,Y)\leqslant H(X,Z)+H(Y,Z)$
(三角不等式，(註2)) (7)

在結束本節之前，我們定義 Z^n 一個子集之離散度。

定義6：令 $A\subset Z^n$ ，則A之離散度定爲

$$d(A)=\min_{\substack{X\in A\\Y\in A\\X\neq Y}}|X-Y|=\min_{\substack{X\in A\\Y\in A\\X\neq Y}}|X+Y| \tag{8}$$

即$d(A)$爲A中不同元素間距離的最小值。

定義7：令 $A\subset Z^n$ ，若$X\in A$，$Y\in A$可以導至$X+Y\in A$，則A稱爲 Z^n 的一個封閉子集。

定理2：若A爲一封閉子集，則

$$d(A)=\min_{\substack{X\in A\\X\neq 0}}|X|$$

證明很顯然，因(8)式中$X+Y$可以A中之元素表示之。例如表二中的輸送碼爲一封閉子集，而其離散度爲2，再回到表二，表二的製碼法可以用矩陣的乘法表示，我們若以 X_m 表二位的信息碼，而X表四位的輸送碼，則

$$X=X_m\begin{bmatrix}1&0&1&0\\0&1&0&1\end{bmatrix}$$

$$=(x_1,x_2)\begin{bmatrix}1&0&1&0\\0&1&0&1\end{bmatrix}$$

當然一般的製碼法(3)並一定可以以這種乘法表示，但可以用這種乘法表的製碼法至少有兩個好處，第一是規則簡單，第二是它所形成區域$A=f(Z^m)$是Z^n中的一個封閉子集，這個特性我們會在第四節中用到。

三、檢錯碼

定理3：設A爲一碼表，則它能檢驗出至多k位錯誤的充要條件是$d(A)\geqslant k+1$，即A之離散度必須大於或等於$k+1$。

證明：充分性：設$d(A)\geqslant k+1$，$\bar{X}$爲X之誤傳，因$|\bar{X}-X|\leqslant k$，而$d(A)\geqslant k+1$，故$\bar{X}\notin A$，必定有錯。

必要性：若$d(A)\leqslant k$，則存在兩X，Y，$|X-Y|\leqslant k$。因我們允許k個錯誤，故Y可以爲X之誤，但因$Y\in A$，我們不知可能的錯誤，與假設要求不合。

在這兒要注意一點的,所謂可以檢出k個錯誤是指定義3所說的,如果X有k個錯，我們只知道它至多有k個數碼錯了，但並不知道錯在何處。對數碼爲0，1的碼字而言，一旦知道錯在何處，就等於可以改正了，以表二爲例，因$d(A)=2$，$k=1$，因此，如果有一位誤傳，我們可以知錯(但不知錯在何處)若有兩位發生錯誤，例如0101傳成0000，則我們就發現不出錯誤了，這要$d(A)\geqslant 3$才行。

由定理3可知若要檢定輸送的碼字是否有一個或一個以下的錯誤，我們得要有$d(A)=2$，若不加檢定碼，因A的元素皆爲信息碼，於是$d(A)=1$，不可能檢定錯誤。因此我們至少要加一位檢定數碼。現在我們要證明只要加一位檢定數碼就是以檢定一位的錯誤，因此表二中的重覆輸送是不必要的。令$0(X)$表X的奇偶數(parity)，即

$$0(X)=\begin{cases}1 & \text{如果}X\text{含有奇數個}1\\0 & \text{如果}X\text{含有偶數個}1\end{cases}$$

則很容易證明下面的定理。

定理4：若$X \nsim Y$，$0(X)=0(Y)$
則$H(X,Y)\geqslant 2$

系理：若A中之元素均有相同的奇偶數，則若在輸送時出了一位錯，我們可以檢出。

要使得所有碼表中的元素都有相同的奇偶數是很容易的事，只要在信息碼後加上一位奇偶檢定碼使之成爲一個含有偶數個1的輸送碼就可以了（註二），因此對表一的四個信號而言，可以檢定一個錯的編碼法是令$a=000$，$b=011$，$c=101$，$d=110$。

奇偶檢定碼與原碼字長度m無關，前面談到的ASCII碼都是七位再加一位奇偶檢定碼。因此a實際是八位的11000011。

若要使表一中的信息碼有檢定二個錯的功能，必須要$d(A)\geqslant 3$，字碼表⑩是不行的，因$d(A)=2$，再加一位檢定碼行不行？答案是不行，但這不是一眼可以看出來的，加兩位尾數的方法很多，但我們可以用窮舉法證明。若是我們令a爲$00a_1a_2$，b爲$01b_1b_2$，c爲$10c_1c_2$，則不可能使得它們之間的距離都不小於3，加上三位如何？這就更難驗算了，我們把這個問題與下節的自動校正碼合併，我們將證明可以檢二位錯的檢錯碼，就有自動改正一位錯誤的功能。

在結束本節之前，我們要順便提一下，當我們加碼來檢定錯誤，也會因碼位加多而增加了一個碼字發生錯誤的或然率，但在出錯或然p很小時，仍爲“合算”，讀者很容易算出增加碼位後發生某種錯誤的或然率，因非本文主旨，我們不再詳談。

四、校正碼

定理5：一個輸出碼表A能校正至多k位錯碼的充要條件是$d(A)\geqslant 2k+1$。

證明：充分性：設$d(A)\geqslant 2k+1$。我們要證明若$X\neq Y$，X誤傳成$\overline{X}$，Y誤傳成$\overline{Y}$，其錯誤均不超出k個，則$\overline{X}\nsim\overline{Y}$，因

$$|X-\overline{X}|\leqslant k,\quad |Y-\overline{Y}|\leqslant k$$

由三角不等式及$|X-Y|\geqslant 2k+1$，得

$$\begin{aligned}|\overline{X}-\overline{Y}|&\geqslant|\overline{X}-Y|-|Y-\overline{Y}|\\&\geqslant|X-Y|-|X-\overline{X}|-|Y-\overline{Y}|\\&\geqslant 1\end{aligned}$$

故$\overline{X}\nsim\overline{Y}$，即$X$與$Y$不可能混合誤傳。

必要性：設$d(A)\leqslant 2k$，則可找到X，$Y\in A$且$|X-Y|=q\leqslant 2k$，我們不妨假定X與Y在$\alpha_1\alpha_2\cdots,\alpha_q$位置上不同，取$\overline{X}$與$X$僅在$\alpha_1\cdots\alpha_k$位置上不同，則

$$\begin{aligned}|\overline{X}-X|&=k\\|\overline{X}-Y|&=q-k\leqslant k\end{aligned}$$

故$\overline{X}$之錯不能改正，因不知它是由X或Y而來，本定理證畢。

因此若要編一個可以改正k個（或k個以下）的校正碼，只要在每個爲m位的信息碼後加若干位的數碼，使得它的輸送碼有$2k+1$的離散度就可以了。

定理6：對具m個數碼的信息碼A，若加上t位的檢錯數碼，使之可以自動校正k或k個以下的錯誤，則t必滿足。

$$1+\binom{n}{1}+\binom{n}{2}+\cdots+\binom{n}{t}\leqslant 2^t \quad (11)$$

上式中$n=m+t$。

證明：在A中有2^m個碼字。任一個碼字若錯了0，1，…，k位時，必不會與另一個碼字錯了0，1，…，k位時相同。因每個碼字有$\binom{n}{1}$個與它差一位的，$\binom{n}{2}$與它差2位的，…，的字碼，加上自己，可見每一個A中的元素都佔有了

$$1+\binom{n}{1}+\binom{n}{2}+\cdots+\binom{n}{t}$$

個碼位，因Z^n只有2^n個元素，故

$$\left[1+\binom{n}{1}+\binom{n}{2}+\cdots+\binom{n}{t}\right]2^m \leqslant 2^n$$

消去2^m，得⑾。

注意：一般書中對原碼字爲m位，檢定碼字爲t位的線性二元碼的新碼以（m，t）碼表之。

在⑾中，因$n=m+t$，並沒有公式可以表示t的解，但用代入法，很容易算出在給定m時，最小可以滿足⑾之t，我們稱之爲t之下界，它們的值可在表五中看到。譬如說$m=10$，$k=2$，則至少得加8個檢定數碼不可，但如何加這8個檢定數碼，却是一個難題。因爲在⑶式中的f可以千變萬化。在此我們只討論一種規則的編碼法，一般稱之爲線性碼，或群碼（見註三），我們要求在編碼：

$$f: Z^m \to Z^n \tag{12}$$

中的f，爲一矩陣F_0，用符號表示：

$$f(X_m)=X,\ X_m \in Z^m,\ X \in Z^n$$

F爲一$m\times n$的矩陣，且

$$X=X_m F \tag{13}$$

爲了保持前面m位原碼的數碼不變，我們將M寫成2個子矩陣，$m\times m$的恒等矩陣I_m及$m\times t$的矩陣D，即

$$F=\begin{bmatrix}1&0&0&\\0&1&0&D\\\vdots&&\vdots&\\0&&1&\end{bmatrix}=[\,I_m \vdots D\,]$$

亦即

$$X=X_m F=[\,X_m \vdots X_m D\,] \tag{14}$$

如何找到D，使得$d(A)\geqslant 2k+1$是製造校正碼的關鍵，它的原理却很簡單，先定義一個$n\times t$的矩陣M如下：

$$M=\begin{bmatrix}D\\I_t\end{bmatrix} \tag{15}$$

上式中I_t爲一個$t\times t$之恒等矩陣，則若$X\in A$，由⑭及⑺(ii)可知

$$\begin{aligned}XM&=[\,X_m \vdots X_m D\,]\begin{bmatrix}D\\I\end{bmatrix}\\&=X_m D+X_m D\\&=0\end{aligned} \tag{16}$$

將X與M以另一形式展開，即令$X=(x_1, x_2, \cdots, x_n)$，把$M$分解爲$n$個$t\times 1$的向量

$$M=\begin{bmatrix}v_1\\v_2\\\vdots\\v_n\end{bmatrix}$$

則16成爲

$$\begin{aligned}XM&=(\,x_1 \cdots x_n\,)\begin{bmatrix}v_1\\\vdots\\v_n\end{bmatrix}\\&=\sum_{i=1}^{n} x_i v_i=0\end{aligned} \tag{17}$$

由這個關係，我們可以證明編校正碼的一個基本定理。

定理7：在⑭中的D，當它寫成⑮⑰式時，若沒有任何$2k$個$v_1, v_2, \cdots, v_n$加起來爲0，則$d(A)\geqslant 2k+1$。

證明：由⑰式可知，$\sum_{i=1}^{n} x_i v_i=0$，但沒有$2k$個$v_1, \cdots, v_n$加起來是0的，所以$(x_1, x_2, \cdots, x_n)$中至少要有$2k+1$個1才行。因此除了$X=0$之外，$|X|\geqslant 2k+1$，因A爲一封閉子集（註四），由定理2可知

$$d(A)=\min_{\substack{X\in A\\X\neq 0}} |X| \geqslant 2k+1$$

本定理得證。

定理8：若 $k=1$，則任何 n 個不相同的 v_i 可使得(14)的轉換滿足 $d(A)\geqslant 3$，又用此法時 t 必滿足 $m\leqslant 2^t-t-1$。

證明：因任何兩個不相同的 v_i 相加不等於零向量0。由定理7很容易證得本定理。因 t 位檢定碼字只有 2^t-1 個異於0的向量，而我們需要 n 個不同的，故必須有 $n\leqslant 2^t-1$，即 $m\leqslant 2^t-t-1$。

系理：當 $k=1$ 時，用(14)所製成的最經濟（指 t 大小而言）的校正碼，就是最經濟的校正碼。

證明：比較定理8與定理6之系理1，可知當 $k=1$ 時，不滿足 $m\leqslant 2^t-t-1$ 的 t，也就是不滿足(11)，即不可能以任何方式做成校正碼。

現以 $m=4$ 爲例，由 $m\leqslant 2t-t-1$ 可知最小的 t 是3，M 中 v_i 的選法只有一種，我們要用全部的 $2^3-1=7$ 個向量。可令

$$M=\begin{bmatrix}1&1&1\\1&1&0\\1&0&1\\0&1&1\\1&0&0\\0&1&0\\0&0&1\end{bmatrix}$$

因此

$$F=\begin{bmatrix}1&0&0&0&1&1&1\\0&1&0&0&1&1&0\\0&0&1&0&1&0&1\\0&0&0&1&0&1&1\end{bmatrix}$$

表三中有這種字碼表

表三　由 $X=X_mF$，(17)式所製成的一位自動校正碼

原碼（或信息碼）				檢定碼		
x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	0	0	0	0	0	0
0	0	0	1	0	1	1
0	0	1	0	1	0	1
0	0	1	1	1	1	0
0	1	0	0	1	1	0
0	1	0	1	1	0	1
0	1	1	0	0	1	1
0	1	1	1	0	0	0
1	0	0	0	1	1	1
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	1	0	0	0	0	1
1	1	0	1	0	1	0
1	1	1	0	1	0	0
1	1	1	1	1	1	1

要解碼，也很容易，最原始的解法是比較收到的信號 $\overline{X}$ 與所有 A 裏的元素，取其最近者，若 $\overline{X}$ 只有一位或沒有錯誤，則由碼表中找出這個最近的輸出碼字，就是所求的校正的碼字了。例如

$$\overline{X}=0101000 \qquad (18)$$

把它與表一中的元素比較，可知它與0111000最近，如果只有一位錯，必定是由它產生的。

但如果 m 很大，A 中有 2^m 個元素，如作一一相比較，對大量的信號而言是可觀的計算量，若令 $e_0=0$，$e_i=$只有 i 位含1的 n 維向量，因只有一位錯，則在 $\overline{X}+e_i$，$i=0,1,\cdots,n$ 中必有一個屬於 A。換言之，若是第 j 位錯，則 $\overline{X}+e_j\in A$，因凡在 A 中之元素皆適合(17)，故

$$(\overline{X}+e_j)M=0 \qquad (19)$$

在計算時⑲可以簡化成 $\overline{X}M = e_j M$ ，故只要比較 $\overline{X}M$與 $e_j M$ 就可以了。$e_j M$ 是定向量，可以先存起來，而$\overline{X}M$又要乘一次就行了，若m不很小，$2^m \gg n$ ，比較 $n+1$ 個向量要比比較 2^m 向量快多了。

以表一爲例，$e_j M$如下表：

表四

e_j	$e_j M$	
(0000000)	(000)	（沒錯）
(1000000)	(111)	
(0100000)	(110)	
(0010000)	(101)	
(0001000)	(100)	
(0000100)	(011)	
(0000010)	(010)	
(0000001)	(001)	

再以⑱爲例

$$\overline{X}M = (\ 0101000\ ) \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = (\ 101\ )$$

與 $e_3 M$ 相合，即在第 3 位發生錯誤，原碼應爲$\overline{X} + e_3 = (\ 0111000\ )$，與直接比較之答案相同。

定理 8 之系理不可以推廣到 $k \geqslant 2$ 的情形去，因當 $k \geqslant 2$ 時，線性碼 t 的下限與任何碼的下限⑪不同。

定理9：若以⑭法編可以校正至多含 $2k+1$ 個數碼錯的碼，則 t 必須滿足

$$\binom{n-1}{1} + \binom{n-1}{2} + \cdots + \binom{n-1}{2k-1} < 2^t - 1 \qquad (20)$$

證明：用⑭編碼，我們必須找到 n 個 $t \times 1$ 的非 0 向量 $v_1, v_2, \cdots, v_n$ 使得沒有任何 $2k$ 個加起來會是 0 ，這種向量的找法可用“篩”的方法，先取任 2 個異於 0 的向量 v_1, v_2 ，凡是與這兩個向量相加可以爲 0 的向量全部去掉，在餘下的向量中任取一個作v_3，篩去凡能與 v_1， v_2， v_3 加起來爲 0 的向量，…，假定我們 $v_1, v_2, \ldots, v_i$時已找到了 $2k-1$ 個向量，則我們必須對所有的 v_i ，二個 v_i 的和，3 個 v_i 的和，…一直到 $2k-1$ 個 v_i 的和都篩去，因爲它們都可以使M不合要求（例如我們取了 $v^* = v_1 + v_2 + v_3$ ，則 $v^* + v_1 + v_2 + v_3 = 0$）。這一共篩去了

$$\binom{i}{1} + \binom{i}{2} + \cdots + \binom{i}{2k-1}$$

個向量，因一共只有 $2^t - 1$ 異於 0 的 t 位向量，故對 $i = 1, 2, \cdots, n-1$ 而言

$$\binom{i}{1} + \binom{i}{2} + \cdots + \binom{i}{2k-1} \leqslant 2^t - 1 \qquad (21)$$

都要成立，特別是當 $i = n-1$ 時左式最大時⑵時也得成立。故本定理得證。

滿足定理 9 的 t ，經我們用計算機算了一些數據，它們的值列在表五中。可見當 $k > 1$ 時，線性碼可能不是用的最小的 t 。但由於線性碼容易製，而又容易解，所以仍是最常用的編碼法。

五、結　論

有人說「好的理論，它必是簡潔的」我們已看到了校正碼理論簡潔的一面，其中最主要的關鍵在於　明距離的定義及式⑰。但我們所舉的例子都在 $k = 1$ ，即改正一個錯誤的情形。再向前推，就應驗了另一個有人說的：「簡單的東西都已經被人做完了，所剩的全是難或繁的題目了」。在 $k > 1$ 的情形，編碼與解（

表五 對m位信號碼所要加的最小尾位數以作成k位校正碼。
在線性碼t的下界由公式⑳得出，一般下界t由公式⑾得出。

	k/m	1	2	3	4	5	6	7	8	9	10	20	30	40	50
1.	線性碼t	2	3	3	3	4	4	4	4	4	4	5	6	6	6
	一 般t	2	3	3	3	4	4	4	4	4	4	5	6	6	6
2.	線性碼t	4	7	8	8	9	9	10	10	10	11	13	14	15	16
	一 般t	4	5	6	6	7	7	7	7	8	8	9	10	11	11
3.	線性碼t	6	11	12	13	14	15	15	16	16	17	20	22	23	24
	一 般t	6	8	8	9	9	10	10	10	11	11	13	14	15	16
4.	線性碼t	8	15	16	18	19	20	20	21	22	22	26	29	31	32
	一 般t	8	10	11	11	12	12	13	13	14	14	17	18	19	20

或校正）碼就要變得複雜了。一般所用的是一種叫做循環碼（cyclic code）的方法，讀者若有興趣，可以在本文所附之參考資料中找到。

另外一個研究方向是針對碼字中數碼錯成串的改正。因有時一個強烈的干擾因素可以破壞一成串的電碼，這時候，本文所談的方法就失去了效用。譬如說，$m=7$位的ASCII碼加上$t=4$位的檢定碼可以校正一位錯誤，但如果通到了一個強的干擾，可以把一串十位的數碼全破壞了。當然我們再也無法用這十位數來使原信號復原了。很顯然的，我們若用一個一個碼字的傳法，則我們無法恢復一個破壞得很厲害的碼字。但我們若把所有碼字比方說一百個碼字，同時放在一起加校正碼，則有可能可以恢復一長段的錯誤。事實上可以有一種叫Reed-Solomon的碼，它只要在具833數碼的信號碼上加上56個檢定碼就可以改正不多 於22 個成串的數碼的錯誤。以ASCII碼爲例，833個數碼含有119個信號，而只要加 56個檢定碼就可以改錯，是非常令人驚奇的事。它比奇偶檢定碼還要經濟，此地的主要關鍵在於此 22 個或 22 個以下的錯誤必發生在成一串的 22 個數碼中。但它的理論也不是本文的範圍可以包括的。

最後，我們要提到的是這些編碼理論都有其實際用途，而且市面上有不少用這些方法製成的硬、軟體裝置。

註一：這是假定兩次錯誤爲獨立事件，有時音會干擾一般信息，則相鄰的錯誤就不能假定爲獨立事件了，在第五節內會提到這個問題。

註二：令全部碼的奇偶數爲1也可以。

註三：熟悉較高深數學的讀者可知下面二結果
(i)$H(X,Y)$是一個量度（measure）。
(ii) Z^n 可以想成一個交換群，或在$\{0,1\}$域中的線性向量空間。

註四：因2^m個碼全用上了，$X\in A$，$Y\in A$，則$X=X_mF$，$Y=Y_mF$，可知$X+Y=(x_m+Y_m)F\in A$。

參考書目

1. Djinitri Wiggert, "*Error-Control Coding and Applications*", Artech House, Inc., Dedham, Massachustts, 1978.
2. Judith L. Gersting, "*Mathematical Structure for Computer Science*", W. H. Freeman and Company, San Francisco, 1985.
3. Vera Pless, "*Introduction to the Theory of Error-correcting Codes*," Wiley & Sons, 1982.