行政院國家科學委員會專題研究計畫　成果報告

子計畫二：擬亂數產生器與編碼及其密碼之應用(3/3)

計畫類別：整合型計畫
計畫編號：NSC93-2213-E-009-010-
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日
執行單位：國立交通大學資訊工程學系(所)

計畫主持人：陳榮傑

計畫參與人員：陳榮傑

報告類型：完整報告
報告附件：出席國際會議研究心得報告及發表論文
處理方式：本計畫可公開查詢

中　華　民　國 94 年 10 月 31 日

行政院國家科學委員會補助專題研究計畫 ☑ 成 果 報 告 □ 期中進度報告

理論密碼學與應用
子計畫二：擬亂數產生器與編碼及其密碼之應用（3/3）

計畫類別：□ 個別型計畫　　☑ 整合型計畫
計畫編號：NSC　93－2213－E－009－010－
執行期間：　93 年　8　月　1　日至　94　年　7　月　31　日

計畫主持人：陳榮傑教授
共同主持人：
計畫參與人員：胡鈞祥、梁漢璋、蔡志彬、楊葉薰

成果報告類型(依經費核定清單規定繳交)：□精簡報告　☑完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：國立交通大學資訊工程學系

中　華　民　國　94　年　10　月　30　日

中文摘要

在隨機演算法中，如何產生一個亂數，以及亂數對隨機演算法進行去隨機化的動作，是非常重要的課題。亂度萃取器是一個能夠從微弱的亂度源中萃取亂度出來的演算法，這是我們目前用來對隨機演算法進行去隨機化的主要方法。我們可以利用亂數產生器做出亂度萃取器，這些亂數產生器都是要建立在一些非常困難的問題上，所以我們研究的方向之一是探討亂度萃取器與已知的困難問題之間的關聯性，以及如何更有效率的利用亂數產生器。

許多的密碼與通訊的演算法中，擬隨機序列產生器已是一個不可或缺的重要原件，包括了串流密碼演算法、區塊密碼演算法與擬亂數產生器均會使用到它，由這些演算法所建構出的密碼系統，其安全性也往往與產生器所輸出的序列是否夠隨機有直接的關係，因此，如何建構一個好的擬隨機序列產生器已經變成了一個很重要的研究課題。在串流密碼中使用到的擬隨機序列產生器：密鑰流產生器，近年來，相關理論發展得非常的迅速，以線性反饋移位暫存器為基礎的密鑰流產生器，由於結構簡單，且已有理想的數學工具分析其隨機性與不可預測性，因而成為目前串流密碼的主流。一個密鑰流產生器是否能夠防止各種攻擊法的攻擊，是判別它安不安全的依據，同時也是檢驗產生器的輸出序列是否具備不可預測性的重要指標，這與密鑰流產生器的組成員件『布林函數』有密切的關係，一個具有平衡性、相關免疫性、傳播特徵與非線性度的布林函數將比較能夠抵擋各種攻擊法的攻擊。我們將針對串流密碼中三種不同形式的擬隨機序列產生器：過濾產生器、組合產生器與鐘控產生器進行研究，希望藉由研讀產生器中布林函數的相關性質，找出一個具備良好特性的布林函數來，以此建構出具有不可預測性的擬隨機序列產生器。另外，我們也將針對1967年Golomb 所提出的擬隨機序列性質進行研究，並實際撰寫五種統計檢測的方法，來檢驗我們所設計出來的擬隨機序列產生器所產生的序列是否符合Golomb 所提出的要求。

目前建構非線性度高的相關免疫函數大致上朝著兩個方向在研究，其一為在固定布林函數之相關免疫性階數的條件下，尋找一個非線性度高的組合函數；另一種則是由具有最高非線度性之 Bent 函數出發，透過各種函數結合方式，組合出具有相關免疫性的布林函數來。在此我們將以第一種方式建構布林函數，也就是在固定函數之相關免疫性階數的條件下，透過基因演算法的演化機制，得出一個固定相關免疫階數與非線度高的函數。

關鍵詞：隨機演算法、亂度萃取器、擬亂數產生器、密鑰流產生器、線性反饋移位暫存器、相關免疫、非線性度、Golomb 準則、基因演算法。

英文摘要

In randomized algorithms, it is an important subject how we could generate a random number and de-randomize randomized algorithm with random numbers. An extractor is an algorithm that is able to extract randomness from weak random source. It is now the major method on de-randomizing a randomized algorithm. We could make extractors from pseudorandom generators, and these pseudorandom generators are based on some very hard predicate. And one aspect of our research is to explore the relation between extractors and some well-known open hard problems, and the way to use pseudorandom generators more efficiently.

Pseudorandom sequence generators are essential components in many cryptographic algorithms including stream-cipher algorithms, block-cipher algorithms and pseudorandom number generators. The security of many cryptographic systems depends upon the unpredictability of the numbers generated. Therefore, constructing a good pseudorandom sequence generator becomes important. The theory of keystream generators in stream cipher has been developed for many decades. Most modern constructions of stream ciphers are based on linear feedback shift registers (LFSR) due to their simple structures. In addition, a lot of mathematical tools and theory are developed to help analyze the randomness and unpredictability of the numbers generated. In stream cipher, a keystream generator should be able to defend all possible attacks which are caused by the weakness of designed Boolean functions. The designed factors of a Boolean function include balance, correlation immunity, propagation characteristics and nonlinearity.

In this project we have investigated three kinds of LFSR-based pseudorandom number generators: filter generators, combining generators and clock-control generators and study those characteristics of related Boolean functions for us. In 1967, Golomb was the first to establish some criteria for pseudorandom sequences. To follow Golomb's criteria we finally use five statistical tests to justify the goodness of our proposed pseudorandom sequences.

There are two directions to construct functions of high nonlinearity and correlation immunity. One is to find a high nonlinear combination function under the constraint of fixed correlation immunity. The other is based on high nonlinear bent functions to produce high correlation immune Boolean functions. In this project we use genetic algorithm approach and adopt the first direction. The resulting Boolean functions work well.

Keywords: randomized algorithm, extractor, pseudorandom generator, keystream generator, linear feedback shift register, correlation immunity, nonlinearity, Golomb's criteria, genetic algorithm.

目錄

報告內容

一、前言

在密碼學的應用上，常常會因為安全上的需要而產生一些亂數，例如金匙的產生、密碼協定中輔助的亂數等等。但是因為真正的亂數產生不易，又要能夠符合我們所需的長度，因此我們便需要一個假亂數位元產生器(pseudo-random bit generator，簡稱為 PRBG)。一個 PRBG 是以一個較短的亂數作為種子(seed)，經過運算之後，擴展為一長度較長的數字，使其看起來像是真正的亂數，讓我們無法分辨。

二、研究目的

一個擬隨機序列產生器的好壞，主要取決於其所產生的擬隨機序列是否具備隨機性與不可預測性，而本計畫實際以線性反饋移位暫存器與布林函數建構出擬隨機序列產生器，並透過線性反饋移位暫存器與布林函數理論的分析說明其隨機性與不可預測性，希望能對此領域有所貢獻。

三、文獻探討

密鑰流產生器它會產生一個假隨機序列(Pseudorandom Sequence)稱為密鑰流 $Z = z_1 z_2 \cdots$，串流密碼便是以這個密鑰流序列，逐一將明文(Plaintext)中的每個位元透過XOR運算單元運算，將明文加密成為密文(Ciphertext)：

$$c_i = z_i \oplus m_i \qquad i \geqq 1$$

其中，c_i、z_i與m_i分別為密文、密鑰流與明文序列中第i個位元。而一個密鑰流產生器是否有很高的密碼強度(也就是是否具備雜化與亂化兩種特性)，主要取決於密鑰流產生器的設計，密鑰流產生器可視為參數k的有限狀態自動機（finite state machine）[30]，如圖 3-1 所示。

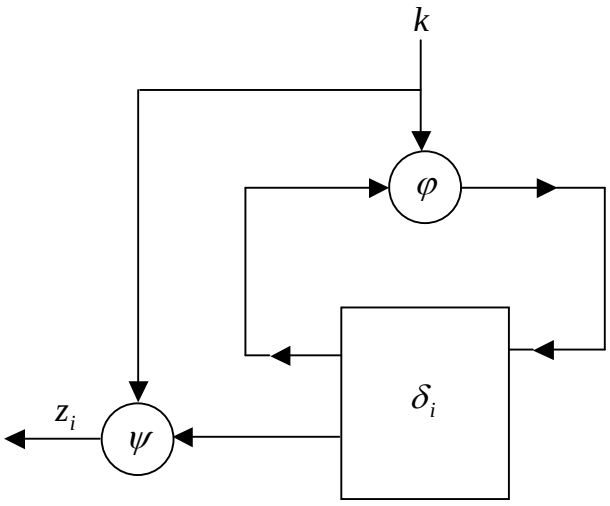

圖 3-1：密鑰流產生器之有限狀態自動機

它是由一個輸出序列集{z_i}、狀態集{δ_i}、兩個函數φ、ϕ和一個初始狀態δ_0所組成。其

中狀態轉移函數 $\varphi \mid \delta_i \rightarrow \delta_{i+1}$，將現在的狀態 δ_i 轉變為下一個狀態 δ_{i+1}。而輸出函數，$\psi \mid \delta_i \rightarrow z_i$，將狀態 δ_i 變成輸出序列的一個位元 z_i。

一般，串流密碼系統依其密鑰流產器的不同，大致上可以分為兩類，一類是以線性反饋移位暫存器(linear feedback shift register，簡稱 LFSR)、布林函數(Boolean function)、正反器(flip-flops)、多工器(multiplexers)與隨機記憶體(random access memories)所組成之密鑰流產生器；另一類，則是以數論(number theory)為基礎所架構出來密鑰流產生器；前者，以線性反饋移位暫存器與布林函數架構之串流密碼系統大致上可區分為三大類：

(a) 過濾產生器：

一類是針對一個線性反饋移位暫存器，非線性地過濾其中的一個狀態(State)，這類的密碼演算法叫做過濾產生器（Filter Generator）[5,31,27,29,38]，如圖 3-2 所示，其中非線性的布林函數 $f(x)$ 我們稱為過濾函數（Filter Function）。目前關於串流密碼的研究中，使用過濾產生器產生密鑰流最著名的密碼系統為 SNOW 密碼系統[14]；

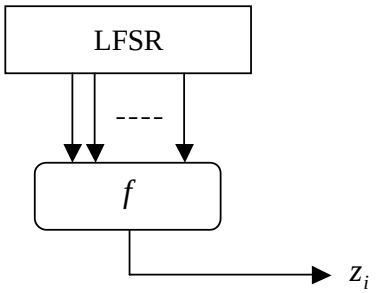

圖 3-2：過濾產生器

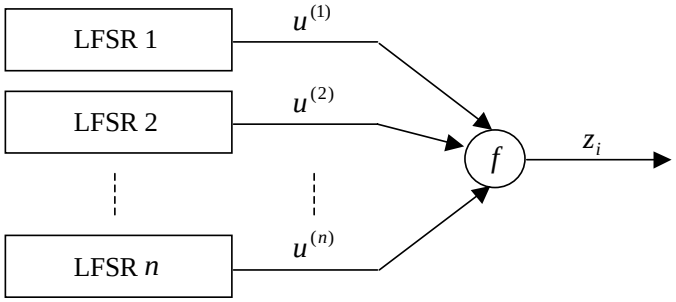

圖 3-3：組合產生器

(b) 組合產生器：

另一種是將幾個線性反饋移位暫存器的輸出透過非線性連結函數(Connection Function)產生密鑰流，這類的密碼演算法叫做組合產生器(Combination Generator)[8,11,12,21,27,39]，如圖 3-3 所示，其中非線性的布林函數 $f(x)$ 我們稱為組合函數（Combining Function）。組合產生器在應用上可說是非常的普及，如藍牙所設計的

無限網路卡與其中的密文函數E_0[15,19,28]即是以組合產生器作為密鑰流產生器。

(c) 鐘控產生器

第三種稱為鐘控產生器(Clock Control Generator)[22, 6, 45, 10, 20, 23, 32, 40, 44]，此種產生器與前面兩種最大的差別在於鐘控產生器的輸出是以一個或多個線性反饋移位暫存器作為控制樞紐產生密鑰流，如圖 3-4 所示，其中非線性的布林函數 $f(x)$我們稱為控制函數（Control Function）。GSM 系統中的加密演算法 A5[3,7,24,17,33]；微電腦上面的軟體 SOBER[4,16,24,25,34,35,36,37]均是使用鐘控產生器作為串流密碼系統之主要架構。

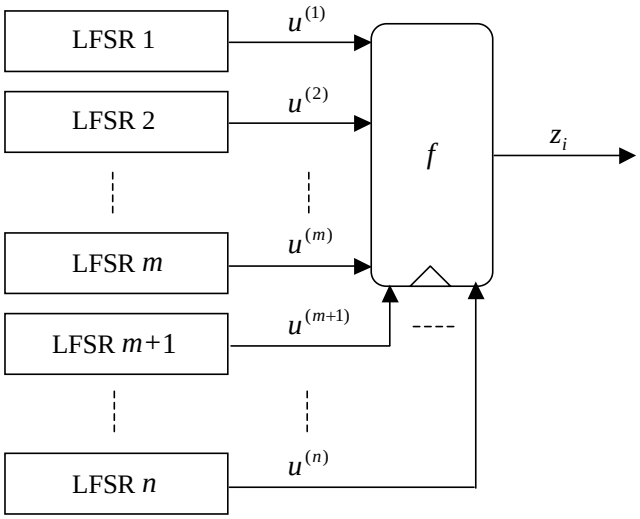

圖 3-4：鐘控產生器

此外，針對串流密碼系統必須具備的兩大條件，在串流密碼系統攻擊方式方面，分別有相對應的攻擊方式攻擊不符合條件的串流密碼系統，此與另一個組成元件布林函數的安全性有密不可分的關係。對於不具備雜化條件之串流密碼系統，將會被最佳仿射近似攻擊法（best affine approximation attack，BAA 攻擊）[2] 攻擊。此種攻擊法是由 Rueppel 與 Shan 於 1987 年所提出的，主要的攻擊概念是將非線性線性布林函數所產生的密鑰流以單一個線性反饋移位暫存器來模擬，直接猜測出近似的密鑰值；而對於不具備亂化條件之串流密碼，則會被相關攻擊法（correlation attack）攻擊，此類型的攻擊法最早是由 Siegenthler [25] 於 1984 年所提出的，其最大的特點，是透過串流密碼系統中每個 LFSR 的輸出序列與密鑰流序列間的相關性，猜測可能的密鑰值。而目前相關的研究，大都以組合產生器所建構的串流密碼系統作為攻擊的對象。以下，我們針對這兩大類的攻擊方式做更詳盡的說明。

1. 相關攻擊法：

針對使用串流密碼系統，為破解密鑰流產生器所產生的密文或找出系統的密鑰值，Siegenthaler 在 1983 年提出一套相關攻擊法，此篇論文使用”分割與克服”(Divide and Conquer)的技巧，觀察密鑰流產生器中，布林函數的輸出與輸入序列間的相關性(此

與布林函數的相關免疫性有直接的關係)，猜測其密鑰值，再藉由統計檢定方法驗證猜測值的正確性，經過反覆類似的猜測與檢定過程，得到正確的密鑰值，便可達到破解密文的目的。往後關於相關攻擊法的研究，均以此篇為基礎，加以改進。

2. 最佳仿射近似攻擊法：

此種攻擊法的概念是將非線性線性布林函數所產生的密鑰流以單一個線性反饋移位暫存器來模擬，直接猜測出近似的密鑰流值，其輸出序列的近似程度，與布林函數的非線性度有極大的關係，當串流密碼系統所選用的布林函數之非線性度夠大時，將可抵擋此種攻擊法的攻擊，反之，將會遭受它的攻擊，如此，對於不知道系統密鑰的攻擊者而言，他可以在不知道密鑰的情況下，找出十分相似的密鑰流來，進而，與密文做 XOR 運算，解出明文來。

3. 區別攻擊法

區別攻擊法是去辨別串流密碼器的輸出密鑰流與真正的隨機二元序列之間是否有不同。如果可以辨別得出來，而且所花的時間小於暴力攻擊法，則表示攻擊成功。在[16]中，P. Ekdahl 與 T. Johansson 提出一個針對 SOBER-t16[24]與 SOBER-t32[25]的辨別攻擊法。其作法是試圖利用一個線性函數去逼近 SOBER-t16 與 SOBER-t32 之中的非線性濾波函數 (nonlinear filtering function, NLF-function)，並以其逼近的誤差作為辨識的依據。

四、研究方法

我們針對擬亂數序列產生器中的布林函數，利用基因演算法物競天擇的特性找出一個具有好的性質的布林函數來取代現有的布林函數。

五、結果與討論

5.1 使用基因演算法之動機

串流加密系統中常見的兩種攻擊方式分別為相關攻擊法（correlation attack）與最佳仿射近似攻擊法（best affine approximation attack），這兩類的攻擊方式我們已在前面做了詳盡的說明。針對這兩類攻擊方式，尋找一個能夠抵抗它們的串流加密系統是個重要的研究課題，目前已有相當多的學者進行此方面的研究，探討何種組合函數能夠抵擋兩類攻擊法的攻擊。但目前的相關研究中往往只尋找出一個能夠抵擋相關攻擊法的攻擊可是卻無法抵擋最佳仿射近似攻擊法的組合函數，或者相反。

針對相關攻擊法，可以選用相關免疫函數來防止；另外，防止 BAA 攻擊法的攻擊，

可以尋找一個最大普值 a 小（等同於非線性度大）的組合函數，為了兼顧兩類攻擊法不同形式的攻擊方式，本章節將針對相關免疫函數，尋找一個 a 值小能夠防止 BAA 攻擊的非線性組合函數。但在尋找此類型的線性組合函數在實作上有一點難度，舉例來說，假設欲尋找一個 a 值小的 $(6,12)$ 相關免疫函數，我們必須在 $5.25\cdot 10^{19}$ 這麼大的樣本空間中尋找，此時暴力窮舉法將不敷使用，因此尋找一個比較有效且規律的方式來尋找是個值得探討的問題。而基因演算法[26]（genetic algorithm，簡稱 *GA*）正符合我們的需求，此種技術可以運用在搜尋龐大的樣本空間上，我們可以將組合函數當成演化的成員，以物競天擇的方式進行演化，由此篩選出優良的組合函數。在此將針對 *GA* 作深入的研究，利用它來尋找優良的相關免疫函數。

5.2 基因演算法簡介

基因演算法（或稱為遺傳演算法 Genetic Algorithms；簡稱 *GA*）是由密西根大學的 John Holland 及其同事、學生在 1975 所發展出來的[26]，其主要目的如下：一、以嚴密而具象的科學方法解釋自然界中『物競天擇、適者生存』的演化過程。二、將生物界中基因演化的重要機制於資訊世界以軟體作模擬。近幾年來，由於電腦科技的進步日新月異，在更穩定的系統支援下，*GA* 已被各個領域廣泛的應用，舉凡符號辨識（pattern recognition）[1]、人工智慧領域中的自我學習機制（mechanism learning）[1]及各類最佳化問題（optimal problem）[18,41]等，*GA* 皆提供了一種不同於以往的思考模式。

GA 主要是以達爾文的「進化論」為基礎，模擬生物界依「 適者生存，不適者淘汰」的生存演化法則，建立出一個保有自然特性的「人工遺傳系統」，以模擬和解釋生物自然進化的過程。簡單的說，*GA* 是一種模擬「物競天擇」的搜尋法則；每個物種在某個生存環境中彼此互相競爭、淘汰，只有適應性強的物種得以存活及繁衍。故其最基本的精神即在於演化（evolution）及篩選（selection）[42]。由程式設計的觀點視之，要將 *GA* 運用在搜尋上，首先必須先針對問題的型態定義每種資料的表示方式，通常將其編碼（coding）成二進位碼稱為資料的染色體（chromosome），而相似資料染色體所組成的集合稱之為基因世代或體群（population）；另外，所謂演化即是經由 *GA* 中的三個運算機制：複製（reproduction）、交配（crossover）與突變（mutation）交互運作產生新的個體，而篩選則是以一個預先定義的評分函數（fitness function）去建構其生存環境，所有的物種皆以符合其要求為終極目標進行演化，保留較符合適應函數的物種而淘汰較差者。*GA* 的優點在於它是一種穩健且有效的搜尋技術，而且相較於其它演算法，*GA* 有較小的機率會陷入局部最佳解中；而 *GA* 的缺點則在於計算時間長，但是此缺點也由於電腦技術的進步而漸漸的被克服了。

整合以上所述，一個簡易的基因演算法程式模型（pseudo code）[26]便可以由下圖（圖

5-1）所示，由圖中簡單的 *GA* 模型中，可以清楚得看出基因機制在產生了第一代物種之後（initial population $P(g)$），隨即以評分函數進行篩選，之後便進入迴圈以求得最佳解。

```
Procedure GA {
        g = 0;
        initial population P(g);
        fitness P(g);
        while(g < termination criterion) {
                g++;
                select P(g) from P(g-1);
                crossover P(g);
                mutate P(g);
                fitness P(g);
        }   // end of while
    }   // end of procedure GA
```

圖 5-1 基因演算法程式模型

深入迴圈內部，可以看出 *GA* 包含四個模組：選擇機制（selection）、基因運算子（operators）、基因世代（population）以及適應函數（fitness function），茲分述如下：

(1) 選擇機制：由上一次適應函數產生的結果，選取表現較佳的子代，並淘汰弱者。

(2) 基因運算子：運用基因演算法中的複製、交配及突變機制，衍生新的物種，其中的複製、交配及突變將在下一小節做較為詳盡的說明。

(3) 基因世代：分解子代與親代物種以供評估。

(4) 適應函數：以事先定義好的評估函數（即環境），計算每一物種的適應力。

四個模組間的關係簡繪如下圖 5-2。

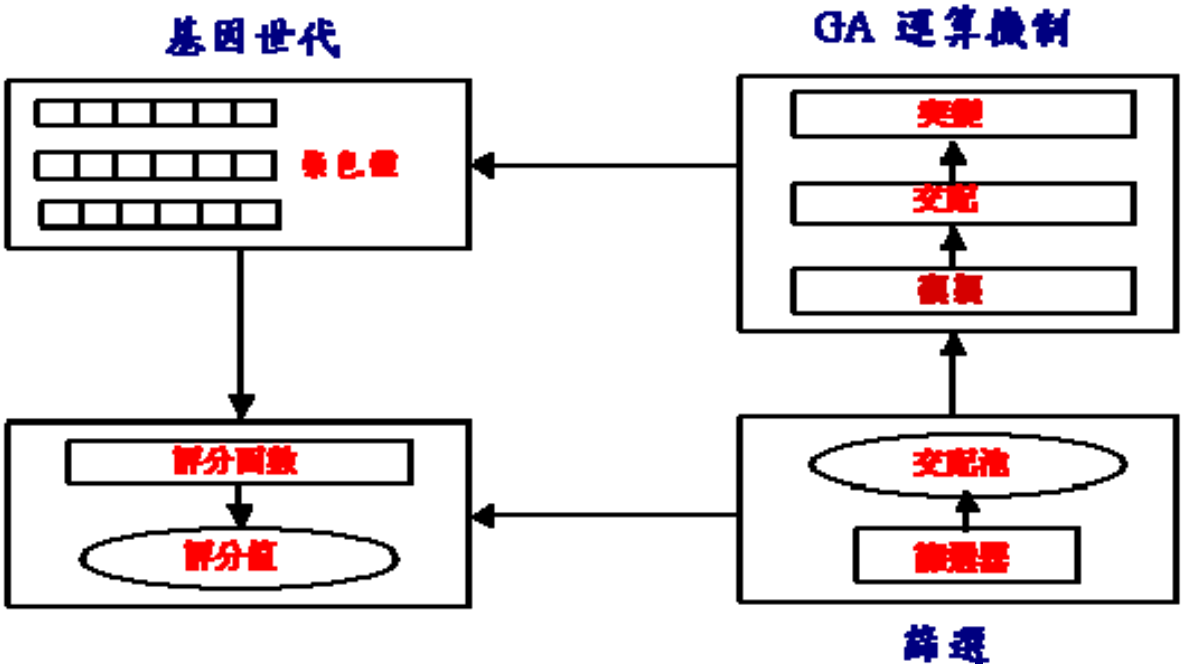

圖 5-2、基因演算法基本架構圖

5.3 基因演算法的基本運算子

前一節 5.2 已敘述了 *GA* 的整個基本流程，本節將更進一步詳述其中的三個重要運算子的運算技巧：複製（reproduction）、交配（crossover）與突變（mutation）。然而，*GA* 即是一個搜尋法則，在面對不同的問題時，其基本流程大致上改變不大，但其中的技巧卻因題目而異，而有所創新或改變。

首先談到複製，複製的目的即在篩選基因世代（population）中的染色體（chromosome），使較優良的染色體得以繼續留存，以重組優良的後代。目前應用較為廣泛的為輪盤法（roulette wheel）[18,9]。顧名思義，輪盤法猶如賭輪中珠子的移動，而珠子最後流得號碼區數值，就決定了孰勝、孰敗；不同的是，此一輪盤並不是等分格的輪盤，而是根據每個基因世代中的染色體的評分函數之百分比例設定，即評分函數越高者所佔據的盤面比例越大，其分格比例可以以下式求得：

$$\mathrm{Pr}_i = \frac{f_i}{\sum_{i=0}^{s} f_i} \tag{1}$$

其中 f_i 為第 i 個染色體由評分函數所評分出來的值，s 為基因世代中染色體的個數，Pr_i 為第 i 個染色體被選中的機率。以下舉一實例說明之。

範例：假設基因世代大小為 4 個 5 位元字串及其相對應的評分函數值如表 5-1 所示：

編號	染色體	評分值	百分比率
1	01101	169	14.4%
2	11000	576	49.2%
3	01000	64	5.5%
4	10011	361	30.9%

表 5-1 染色體的評分函數與比例值表

表 5-1 中第一個染色體的評分函數值為 169，其佔基因世代適應值總和的百分比為 $\frac{169}{1170}=14.4$，其它染色體的百分比以此類推。

■

那麼要如何選擇輪盤中的分格呢？一般採用均等分佈（uniform distribution）的亂

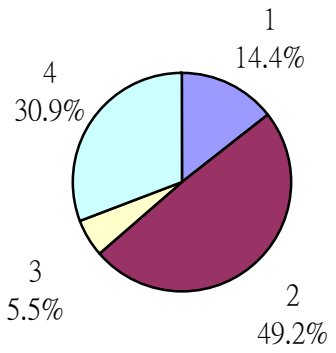

圖 5-3 染色體之輪盤各槽大小

數產生器即可以選擇，如上例中首先隨機產生 0 到 100 之間的任意實數，若該數落在圖 4.3 中 0 到 14.4 的區間內，則 1 號的染色體被選中，若介於 14.4 到 63.6 間則表示 2 號染色體被選中，若介於 63.3 到 69.1 則 3 號染色體被選中，如果是 69.1 到 100 則選擇 4 號染色體。由此觀點思考，越切合評分函數者，其被選出的機率相對變大，在精神上符合適應者較易生存的原則。

所謂的交配，簡單來說就是將經由複製完成的染色體，以兩兩配對的方式作部分內容的交換，以達到訊息交換的目的。其基本理念在於優良的親代會將好的基因遺傳給下一代，以致於可以創造出適應性更高的物種。交配運算子設計的良莠與否在整個基因演算法中佔了舉足輕重的地位；交配頻率過高，將造成物種基因的劇烈變化而無法適度保存上一代的優良基因，但頻率過低也會有停滯在部最佳解上的問題。過去國內外的學者對染色體的交配方式均多有探討[18,26]，以下便就最常見的三者做一簡介。

(1) 單點互換運算子（one-point crossover operator）：

一般來說，從生物的觀點上，交配即發生在兩個染色體互相交換（exchange）其部分之染色體組合。如 John Holland 所提出的單點互換（one-point crossover），乃是經由一隨機選取的一“切點”（cut point），將染色體分成兩部分，而做兩部分的互換。如圖 5-4 所示，兩個母染色體之最後兩位元互換，而產生子染色體。

對單點互換運算來說，一個重要的特徵即是它可以製造出與母染色體完全不同的下一代。其次，還有一個重要的特性，即是它無法辨識出在兩母染色體

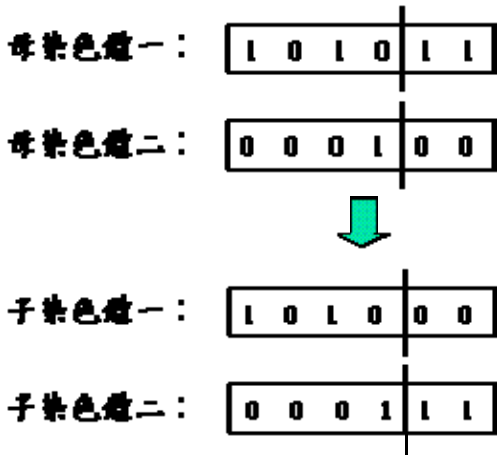

圖 5-4：單點互換運算子之範例

中相同位置的單一位元之異同。如圖 5-5 所示，兩母染色體組合中，第 2、3、4、5 位元都有相同的值，即使在發生交配後，兩個染色體的 2、3、4、5 位元的值依然沒有任何改變。因此，在一個極端的例子下，如兩母染色體完全相同，則單點互換運算子將無任何作用。

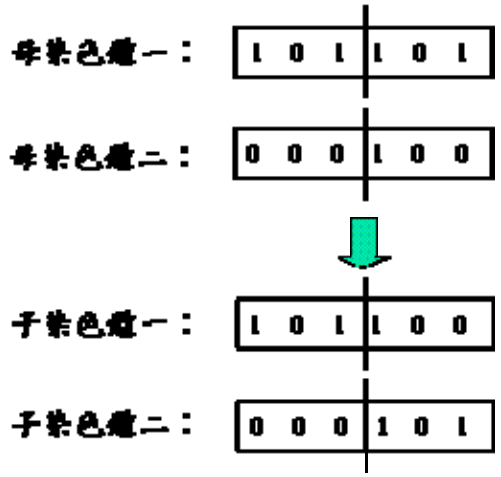

圖 5-5：單點互換運算子特例說明

(2) 雙點互換運算子（two-point crossover operator）：

其原理與單點運算子極為相似，但為了增加組合上的多樣性，以兩個分點來取代單一分點。此運算子最大的優點為可創造單點互換機制下無法創造的基因組合。以下以圖 5-6 說明之：兩個母色體彼此將雙分點之間的基因 00 與 11 互換，而產生子染色體。

圖 5-6 中的的母染色體，若僅由單點運算子進行交配，則會因兩端點的不一致而永遠無法達成如圖所示的基因組合。理論上而言，雙點運算子已可以含括任

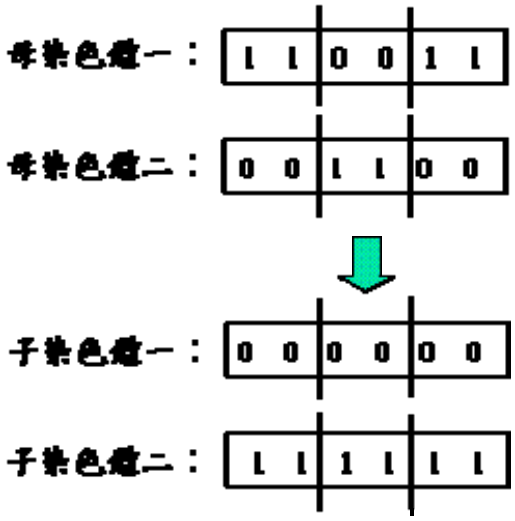

圖 5-6：雙點互換運算子之範例

意的基因組合。

(3) 均一化互換運算子（uniform crossover operator）：

許多基因演算法的相關研究者，急欲尋找能有更多切點的運算子，以符合各類的問題。1989 年，Syswerda 提出均一化交配運算子，允許兩條染色體在單一交配過程中組合各種基因。均一化交配運算子仍是以兩個親代染色體產生兩個子代染色體，而兩個子染色體中的位元取決於由隨機亂數的方式產生一個與染色體上基因個數相同的交換基準（tempate），如圖 5-7 所示，若基準為“1”時，表示第一個子染色體之位元由第一個母染色體提供，若為“0”時，即表示由第二個母染色體提供，而第二個子染色體則由第一個子染色體選擇完後的另一個母染色體提供。

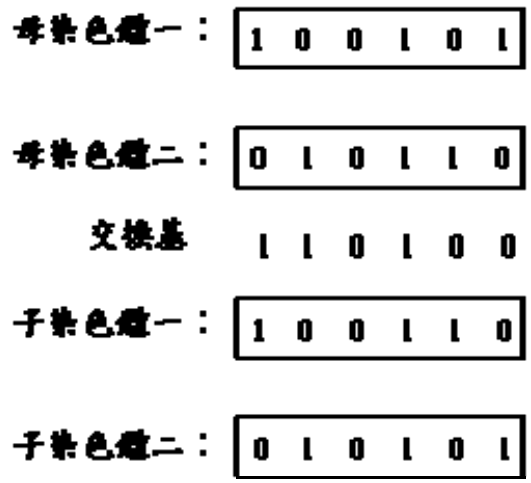

圖 5-7：均一化互換運算子範例

均一化交配運算子與前面所提之單、雙點交換運算子最大的不同在於以下兩點：第一、均一化交配運算子之作用與染色體在編碼特性上的位置無關，不向單、雙切點互換運算子，有越多的固定位元在切割出的染色體中，則越容易造成子染色體的不完整；相反的，對於均一化交配運算子來說，在有些狀況下，亦可能造成一個好的染色體組合遭到大量的破壞。其次，單點與雙點互換運算子較均一化交配運算子具區域性（localalize），較能一致性的傳承保留住原染色體的特性。

所謂的突變，其意義在擷取一種不可預測的訊息，以防止物種在一連串的複製與交配的過程中，侷限在一個區域最佳化（local optimial）的環境中無法跳脫。在自然界中，突變的機率極低，通常只有在百萬分之一左右，雖然是一個很小的機率，卻會帶給物種革命性的改變，因此善定基因演算法中的突變率，將有助於收斂到最佳解的速度提高及物種適應力的增強[17]。在實際的運用中，往往都提高突變率，以期望加快產生適應性強的物種。

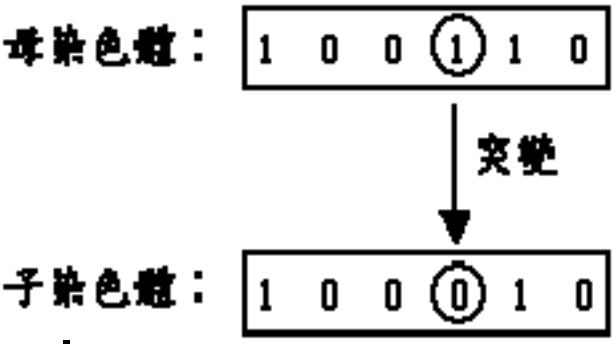

圖 5-8：突變運算子範例說明

突變的做法是就母色體的每個基因逐一產生對應的隨機亂數，並依系統突變率（mutation rate）來控制，若定系統突變門檻值為 0.0001，則當隨機值大於 0.001 時不發

生突變，反之則引導突變發生，如圖 48 之第四個基因發生突變，則子染色體的第四個基因由原本的 1 轉變成為 0。

5.4 以基因演算法尋找相關免疫函數實例

在文獻探討中我們已看過了幾種相關攻擊法，因此對於此類型攻擊方式，如何防範它成為設計一個串流密碼系統的重要研究課題，值得深入探討。目前關於此方面的文獻已相當的多，最早提出防禦相關攻擊法技術的文獻是在 1984 年由 Siegenthaler[43]所提出的，主要的方法是建構一個相關免疫函數作為串流密碼系統中的組合函數，可以保證在某個門檻範圍內之 LFSR 的輸出序列間沒有相關性。然而，在文獻探討的最後，我們也研究過最佳仿射近似攻擊法，防範此種攻擊法最好的方式是尋找一個非線性度高的布林函數作為組合函數。因此，為了兼顧防範相關攻擊法與最佳仿射攻擊法，尋找一個具有相關免疫性且非線性度高的布林函數是必須的。

目前建構非線性度高的相關免疫函數大致上朝著兩個方向在研究，其一為在固定布林函數之相關免疫性階數的條件下，尋找一個非線性度高的組合函數；另一種則是由具有最高非線度性之 Bent 函數出發，透過各種函數結合方式，組合出具有相關免疫性的布林函數來。在此我們將以第一種方式建構布林函數，也就是在固定函數之相關免疫性階數的條件下，透過基因演算法的演化機制，希望能得出一個非線度高的函數。而一個固定相關免疫階數之函數我們將以定理 5.1[43]中的建構方式產生。

定理 5.0：是一個 $(m,n)CI$ 函數，則 $f(x)$ 亦是一個 $(t,n)CI$ 函數，$1\le t\le m$。

證明：假設布林函數的每個輸入變數 X_i（$1\le i\le n$）間統計獨立（independent）的。

當 $t=m-1$ 時，由機率的貝氏定理，可以知道

$$\begin{aligned}&\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-1}}=a_{m-1})\\&=\Pr(X_{i_m}=1)\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-1}}=a_{m-1},X_{i_m}=1)\\&\quad+\Pr(X_{i_m}=0)\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-1}}=a_{m-1},X_{i_m}=0)\end{aligned}$$

由於 $f(x)$ 是一個 $(m,n)CI$ 函數，因此

$$\begin{aligned}&\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-1}}=a_{m-1})\\&=\Pr(X_m=1)\cdot\Pr(f(x)=1)+\Pr(X_m=0)\cdot\Pr(f(x)=1)\end{aligned}$$

因此 $\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-1}}=a_{m-1})=\Pr(f(x)=1)$，這證明了 $f(x)$ 是一個 $(m-1,n)CI$ 函數。

當 $t=m-2$ 時，同樣的由機率的貝氏定理，可以知道

$$\begin{aligned}&\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-2}}=a_{m-2})\\&=\Pr(X_{i_{m-2}}=1)\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-2}}=a_{m-2},X_{i_{m-1}}=1)\\&\quad+\Pr(X_{i_{m-1}}=0)\Pr(f(x)=1\mid X_{i_1}=a_1,...,X_{i_{m-2}}=a_{m-2},X_{i_{m-1}}=0)\end{aligned}$$

由於 $f(x)$ 是一個 $(m,n)CI$ 函數，因此

$$\Pr(f(x)=1 \mid X_{i_1}=a_1,...,X_{i_{m-2}}=a_{m-2})$$
$$=\Pr(X_{m-1}=1)\cdot\Pr(f(x)=1)+\Pr(X_{m-1}=0)\cdot\Pr(f(x)=1)$$

因此 $\Pr(f(x)=1 \mid X_{i_1}=a_1,...,X_{i_{m-2}}=a_{m-2})=\Pr(f(x)=1)$，這證明了 $f(x)$ 是一個 $(m-2,n)CI$ 函數。以此類推即可證明對於所有 t，$1\le t<m-1$，$f(x)$ 均是 $(t,n)CI$ 函數。

■

定理 5.1：若 f_1 與 f_2 是 $(m,n)CI$ 函數，且 $P(f_1=1)=P(f_2=1)=p$，則 $n+1$ 變數的二元函數定義為

$$f(X_1,X_2,\cdots,X_{n+1})=X_{n+1}\cdot f_1(X_1,X_2,\cdots,X_n)+(X_{n+1}+1)\cdot f_2(X_1,X_2,\cdots,X_n)$$

是個 $(m,n+1)CI$ 函數，且 $P[f(X_1,X_2,\cdots,X_n)=1]=p$。

證明：假設 $Z_1=f_1(X_1,X_2,\cdots,X_n)$ 與 $Z_2=f_2(X_1,X_2,\cdots,X_n)$，很明顯的 Z_1、Z_2 與 X_{n+1} 在統計上是獨立的，則對於任意的 $x_1,x_2,\cdots,x_m$ 與 x_{n+1} 滿足

$$P(Z_i=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m,X_{n+1}=x_{n+1})$$
$$=P(Z_i=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m)$$
$$=P(Z_i=1),\qquad i=1,2 \qquad (2)$$

若以 $Z=f(X_1,X_2,\cdots,X_n,X_{n+1})$ 表示，我們可以從(1)可以知道，當 $X_{n+1}=1$ 時

$$Z=f(X_1,X_2,\cdots,X_n,X_{n+1})=f_1(X_1,X_2,\cdots,X_n)=Z_1 \qquad (3)$$

當 $X_{n+1}=0$ 時

$$Z=f(X_1,X_2,\cdots,X_n,X_{n+1})=f_2(X_1,X_2,\cdots,X_n)=Z_2 \qquad (4)$$

又由(2)與(3)、(4)可推得下面結果：

$$P(Z=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m,X_{n+1}=1)$$
$$=P(Z_1=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m,X_{n+1}=1)=P(Z_1=1)$$
$$P(Z=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m,X_{n+1}=0)$$
$$=P(Z_2=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m,X_{n+1}=0)=P(Z_2=1)$$

而 $P(Z_1=1)=P(Z_2=1)=p$，因此可以推出

$$P(Z=1 \mid X_1=x_1,X_2=x_2,\cdots,X_m=x_m,X_{n+1}=x_{n+1})=P(Z=1)=p$$
$$\Rightarrow P(Z=1 \mid X_{i_1}=x_{i_1},X_{i_2}=x_{i_2},\cdots,X_{i_m}=x_{i_m},X_{n+1}=x_{n+1})=P(Z=1)=p$$

由定理 5.0 知道

$$\Rightarrow P(Z=1 \mid X_{i_1}=x_{i_1},X_{i_2}=x_{i_2},\cdots,X_{i_m}=x_{i_m})=P(Z=1)=p$$

因此得證 $f(X_1,X_2,\cdots,X_n,X_{n+1})$ 是個 $(m,n+1)CI$ 函數 ■

上面的定理中，明確的說明了建構一個相關免疫函數的方法，以下舉一個實際的範例說明產生函數的方法。

範例：建構 $(2,7)CI$ 函數

步驟一：（初始狀態）

$$f(X_1, X_2, X_3, X_4) = X_1 + X_2 + X_3$$
$$f_2(X_1, X_2, X_3, X_4) = X_1 + X_2 + X_4$$

步驟二：

$$f_1(X_1, \cdots, X_5) = X_5 \cdot f_1(X) + (X_5 + 1) \cdot f_2(X)$$
$$= X_1 + X_2 + X_4 + X_3X_5 + X_4X_5$$

假設選取的排列（permutation）為$1 \to 5, 2 \to 3, 3 \to 2, 4 \to 1, 5 \to 4$

則 $f_2(X_1, \cdots, X_5) = X_5 + X_3 + X_1 + X_2X_4 + X_1X_4$

步驟三：

$$f_1(X_1, \cdots, X_6) = X_6 \cdot f_1(X) + (X_6 + 1) \cdot f_2(X)$$
$$= X_1 + X_3 + X_5 + X_1X_4 + X_2X_4 + X_2X_6 + X_3X_6 + X_4X_6 + X_5X_6$$
$$+ X_1X_4X_6 + X_2X_4X_6 + X_3X_5X_6 + X_4X_5X_6$$

假設選取的排列為$1 \to 3, 2 \to 4, 3 \to 5, 4 \to 6, 5 \to 1, 6 \to 2$則

$$f_2(X_1, \cdots, X_6) = X_1 + X_4 + X_5 + X_1X_3 + X_4 + +(X_6 + 1) \cdot f_2(X)$$
$$= X_1 + X_3 + X_5 + X_1X_4 + X_2X_4 + X_2X_6 + X_3X_6 + X_4X_6 + X_5X_6$$
$$+ X_1X_4X_6 + X_2X_4X_6 + X_3X_5X_6 + X_4X_5X_6$$

■

透過上述的建構方式，我們將實際產生 $(6,12)$ CI 函數當作基因演算法的基因世代進行演化實驗，真正的去尋找一個非線性度大的相關免疫函數。關於前面章節提到之基因演算法的相關運算子，分別定義如下：

(1) 編碼（coding）：

我們採用相關免疫函數當作基因演算法的基因世代，首先必需設計一個相關免疫函數與其相對應之染色體的對應方式，才能進行演化的動作。由前面的範例中可以發現，其使用排列來建構每個相關免疫函數，而且每組排列方式均唯一代表一個相關免疫函數。因此，對於相關免疫函數最簡單的編碼方式即是以一組排列來代表它，茲以上面範例作說明，$(2,7)$ CI 函數的染色體表示為：

$$\begin{bmatrix} [2 & 3 & 4 & 5 & 1] & \\ [3 & 4 & 5 & 6 & 1 & 2] \end{bmatrix}$$

其中，排列$[2,3,4,5,1]$表示相關免疫函數中的變數 X_1 以 X_2 取代，X_2 以 X_3 取代，X_3 以 X_4 取代，X_4 以 X_5 取代，最後將 X_5 取代成 X_1；在此，我們並非以二位元字串的方式來代表每個物種的染色體，而是以列舉式的排列來表示，雖然此種方式來定義相關免疫函數所對應的染色體有些特別，但與基因演算法的基本觀念並不抵觸，依然適用於演化上的複製、交配及突變等運算子。

(2) 複製（reproduction）：

首先隨機選取 20 個 $(6,12)CI$ 函數作為第 0 基因世代中的染色體，每個染色體以下面的形式表示：

$$\begin{bmatrix} [2 & 6 & 5 & 1 & 7 & 3 & 8 & 9 & 4] & & \\ [6 & 10 & 2 & 3 & 4 & 8 & 5 & 9 & 7 & 1] & \\ [4 & 7 & 10 & 5 & 3 & 1 & 6 & 9 & 2 & 11 & 8] \end{bmatrix} = \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix}$$

其中，π_1, π_2, π_3 分別為三個大小為 9,10,11 的排列。

接著定義一個評分函數（fitness function），此函數主要的功能是提供一個判定組合函數優劣的標準。依據評分函數值，以輪盤法（weighted roulette wheel）的方式隨機挑選染色體進入生殖池（mating pooling）做為交配運算之用。本文中針對關免疫函數 $f(x)$ 所提出的條件函數與機率分佈策略定義如下：

評分函數 $fitness(F) = Max - \max\left|S_{(f)}(w)\right| + \delta$ ，$\delta = (Max - Min)/4$

其中 $\max\left|S_{(f)}(w)\right|$ 為函數 $f(x)$ 的譜值；Max 與 Min 分別代表基因世代所有染色體中最大與最小的譜值；且 δ 中的除數 4 可依據函數的譜值大小不同給予適當的值。

機率分佈函數 $\Pr_i = \dfrac{fitness(f_i)}{\sum_{i=1}^{N} fitness(f_i)}$

(3) 交配（crossover）：

基因世代中的染色體是以一組組排列的方式來表示，此時染色體的交配機制如單點互換運算子、雙點互換運算子及均一化互換運算子均不符合使用，這些運算子將會破壞排列應有的特性。因此，我們定義了另一種交配的方式，針對每個染色體中的排列進行運算，使得進行交配運算後產生的子代依然是一組排列，茲說明如下，假設要將兩個染色體 $(6,12)CI$ 函數 f_1, f_2 進行交配，f_1, f_2 表示如下：

$$f_1 = \begin{bmatrix} [2 & 5 & 6 & 4 & 8 & 3 & 9 & 1 & 7] & & \\ [6 & 5 & 1 & 4 & 3 & 2 & 9 & 8 & 7 & 10] & \\ [11 & 3 & 2 & 1 & 4 & 6 & 9 & 5 & 7 & 8 & 10] \end{bmatrix},$$

$$f_2 = \begin{bmatrix} [4 & 6 & 1 & 2 & 7 & 9 & 8 & 3 & 5] & & \\ [1 & 3 & 5 & 2 & 6 & 4 & 7 & 10 & 8 & 9] & \\ [1 & 10 & 8 & 7 & 2 & 5 & 4 & 11 & 6 & 3 & 9] \end{bmatrix}.$$

首先將染色體中相同位數的排列分成一組，再將每組的排列分割成三個等位數的區段，如果排列的位數無法被 3 整除，此時將剩餘的位數分至中間的區段中，舉例說明之，10 位數的排列中，$\dfrac{10}{3} = 3.333$，因此頭尾兩段區間的位數為 3，中間區間的位數為 $10 - 3*2 = 4$。繪簡圖說明如下：

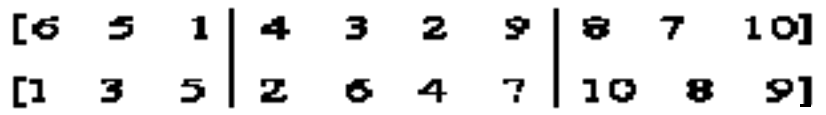

圖 5-9：交配運算子切割範例說明

分好區段後，即可進行交配，此時以中間區段的數作為互換的依據。首先將兩個排列的 4 與 2 互換，再將 3 與 6 互換，將 4 與 2 互換，最後將 9 與 7 互換。茲舉例說明如下：

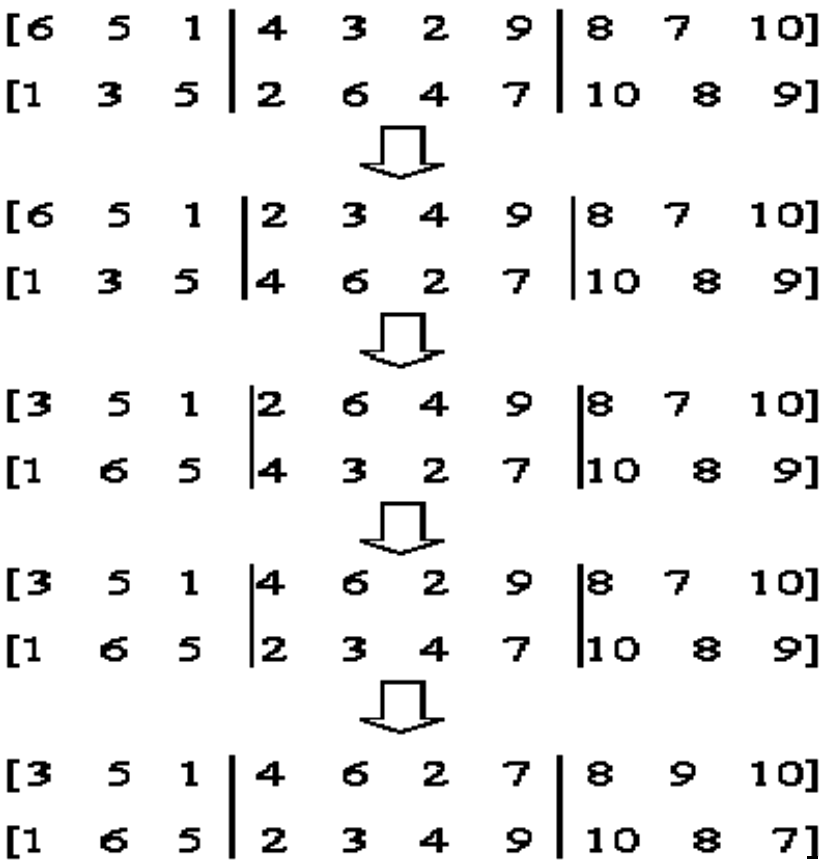

圖 5-10：交配運算子範例說明

此種交配方式的優點在於可以確保所有的排列均可能發生，不會忽略了任何一個排列，這就符合了演化交配的目的。

(4) 突變（mutation）：

本文中所制訂的突變函數為：在一個很小的突變率下（0.0001），針對每個基因世代第 i 個染色體中第 j 個排列的 r_1 與 r_2 位置做互換的動作， j, r_1, r_2 均為隨機選取的變數。舉例如下:假設 $j=2, r_1=4, r_2=7$ ，則

$$\begin{bmatrix} [2 & 5 & 6 & 4 & 8 & 3 & 9 & 1 & 7] & & \\ [6 & 5 & 1 & \underline{4} & 3 & 2 & \underline{9} & 8 & 7 & 10] & \\ [11 & 3 & 2 & 1 & 4 & 6 & 9 & 5 & 7 & 8 & 10] \end{bmatrix} \longrightarrow$$

$$\begin{bmatrix} [2 & 5 & 6 & 4 & 8 & 3 & 9 & 1 & 7] & & \\ [6 & 5 & 1 & \underline{9} & 3 & 2 & \underline{4} & 8 & 7 & 10] & \\ [11 & 3 & 2 & 1 & 4 & 6 & 9 & 5 & 7 & 8 & 10] \end{bmatrix}.$$

根據上面定義的基因演算法相關運算子，我們可以撰寫程式來模擬布林函數演化的情況，我們以(6,12)*CI* 函數當作演化的成員進行實驗。首先隨機選取 8 個相關免疫函數作為第 0 基因世代的染色體成員，並透過適應函數計算每個染色體的譜值、評分函數值與機率分佈值，繪一簡圖如下表 5-2 所示。歷經演化 10 代後所

成員	$F_f(w)$	評分函數值	百分比
$f_1^{(0)}$	-1025	65	8.3%
$f_2^{(0)}$	1023	67	8.5%
$f_3^{(0)}$	-769	321	41%
$f_4^{(0)}$	1025	65	8.3%
$f_5^{(0)}$	-1025	65	8.3%
$f_6^{(0)}$	1023	67	8.5%
$f_7^{(0)}$	1023	67	8.5%
$f_8^{(0)}$	-1025	65	8.3%

表 5-2：第 0 基因世代譜值、評分函數值及機率分佈值表

成員	$F_f(w)$	評分函數值	百分比
$f_1^{(10)}$	769	65	4.2%
$f_2^{(10)}$	769	65	4.2%
$f_3^{(10)}$	-769	65	4.2%
$f_4^{(10)}$	767	67	4.3%
$f_5^{(10)}$	-513	321	20.7%
$f_6^{(10)}$	-513	321	20.7%
$f_7^{(10)}$	-513	321	20.7%
$f_8^{(10)}$	-513	321	20.7%

表 5-3：第 10 基因世代譜值、評分函數值及機率分佈值表

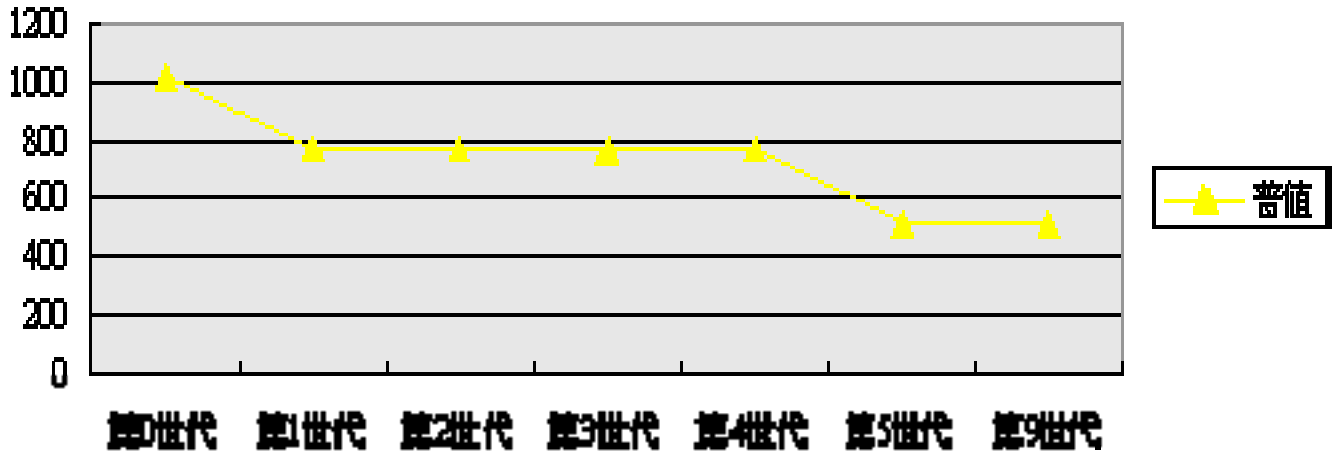

圖 5-11：譜值演化流程圖

產生的基因世代染色體如下表 5-3 所列。我們由圖 5-11 可以看出相關免疫函數的 $F_f(w)$ 值由第 0 世代的 $1023/4096$ 逐漸演化縮小成 $513/4096$，表示越後面的世代之相關免疫函數，具有更高的非線性度，由此可以知道透過基因演算法來尋找非線性度高的相關免疫函數是可行的。我們亦觀察到一個有趣的現象，由定理 5.1 中所建構出的相關免疫函數，由於主要是以排列的方式產生兩個函數建構而成，因此，建構出的函數大致上均是屬於同一類型的函數，具有相似的性質，因此函數的譜值只有某幾種特定的數值，所以在本實驗中，歷經了幾個世代的演化，其值均無明顯的差異。

參考文獻

[1] C. A. Ankenbrandt, B,P Buckles and F.E. Petry, “Scene recognition Using genetic algorithm with sematic nets”, Pattern Recognition Letters, Vol. 11, 1990, pp. 285-293.

[2] E. Biham and O. Dunkelman, “Cryptanalysis of the A5/1 GSM stream cipher,” in Progress in Cryptology – INDOCRYPT 2000, vol. 1977 of Lecture Notes in Computer Science, pp. 43–51, Springer-Verlag, 2000.

[3] A. Biryukov, A. Shamir, and D. Wagner, “Real time cryptanalysis of A5 on a PC,” in Procedding Fast Softeare Encryption 2000, New York:Springer-Verlag, 2000, Vol. 1978, pp.1-18.

[4] S. Babbage, C. D. Cannière, J. Lano, B. Preneel, and J. Vandewalle, “Cryptanalysis of SOBER-t32,” Fast Software Encryption 2003, to be published in LNCS.

[5] J. Bernasconi and C. G. Gunther, “Analysis of a nonlinear feedforward logic for binary sequence generators,” BBC Tech. Rep., 1985

[6] T. Beth and F. Piper, “The stop-and-go generator,” in Lecture Notes in Computer Science 209; Advances in Cryptology: Porc. Eurocrypt ’84, T. Beth, N. Cot, and I. Ingemarsson, Eds., Paris, France, April 9-11, 1984, pp. 88-92. Berlin: Springer-Verlag, 1985.

[7] M. Briceno, I. Goldberg, and D. Wagner, “A pedagogical implementation of A5/1,” Technical report, 1999. web publication, http://www.scard.org/gsm/body.html.

[8] L. Brynielsson, “On the linear complexity of combined shift register sequences,” in Lecture Notes in Computer Science 219; Advances in Cryptology: Proc. Eurocrypt ’85, F. Pichler, Ed., Linz, Austria, April 1985, pp. 156-166. Berlin: Springer-Verlag, 1986.

[9] B. P. Buckles, F. E. Petry, and R. L. Kuester, “Schema survival rates and heuristic search in genetic algorithm”, IEEE Trans, Sys, Man, and Cybernetics, 1990.

[10] W. G. Chambers and S. M. Jennings, “Linear equivalence of certain BRM shift-register sequences,” Electron. Lett., vol. 20, Nov. 1984.

[11] A. H. Chan and R. A. Games, “On the linear span of binary sequences obtained form finite geometries,” in Lecture Notes in Computer Science 263; Advances in Cryptology: Proc. Crypto ’86, A. M. Odlyzko, Ed., Santo Barbara, CA, Aug. 11-15, 1986, pp. 405-417. Berlin: Springer-Verlag, 1987.

[12] A. H. Chan, M. Goresky, and A. Klapper, “Correlation functions of geometric sequences,” Proc. Eurocypt ’90, I. Damgrad, Ed., Springer-Verlag (in press).

[13] J. H. Cheon, “Nonlinear Vector Resilient Functions.” *In Adances in Cryptology, CRYPT* 2001, Springer-Verlag, pp. 458-469, 2001.

[14] P. Ekdahl and T. Hohansson, “SNOW-a new stream cipher,” in Proceedings of First Open NESSIE Workshop, KU-Leuven, 2000.

[15] P. Ekdahl and T. Johansson, “Some results on correlations in the bluetoouth stream generator,” in 10th Joint conference on communications and coding, pp. 210–224, 2000.

[16] P. Ekdahl and T. Johnsson, “Distinguishing attacks on SOBER-t16 and SOBER-t32,” in Fast Software Encryption 2002, LNCS 2365, J. Daemen, V. Rijmen, Eds., Springer-Verlag,

pp. 210-224, 2002.

[17] P. Ekdahl and T. Johansson, “Another attack on A5,” in Proceedings of 2001 IEEE International Symposium on Information Theory, 2001, pp. 160-167.

[18] D. E. Goldberg, “Genetic algorithm in search, optimization, and machine learing”, Addison Wesley, Reading, MA, 1989.

[19] J. D. Golic, V. Bagini, and G. Morgari, “Linear cryptanalysis of bluetoouth stream cipher,” Advances in Cryptology – EUROCRYPT 2002, vol. 2332 of Lecture Notes in Computer Science, pp. 238–255, Springer-Verlag, 2002.

[20] J. Golic and M. V. Zivkovic, “On the linear complexity of nonuniformly decimated pn-sequences,” IEEE Trans. Inform. Theory, vol. 34, pp. 1077-1079, Sept. 1988.

[21] J. D. Golic, “On the linear complexity of functions of periodic GF(q)-sequences,” IEEE Trans. Inform. Theory, vol. IT-35, pp. 69-75, Jan. 1989.

[22] D. Gollman and W. G. Chambers, “clock-controlled shift-registers: A review,” IEEE J. Selected Areas Commun., vol. 7, pp. 525-533, May 1989.

[23] C. G. Gunther, “Alternating step generators controlled by de Bruijn sequences,” in Lecture Notes in Computer Science 304; Advances in Cryptology: Proc. Eurocrypt ’87, D. Chaum and W. L. Price, Eds., Amsterdam, The Netherlands, April 13-15, 1987, pp. 5-14. Berlin:Springer-Verlag, 1988.

[24] P. Hawkes and G. Rose. “Primitive specification and supporting documentation for SOBER-t16 submission to NESSIE.” In Proceedings of the First Open NESSIE Workshop, 13-14 November 2000, Heverlee, Belgium.

[25] P. Hawkes and G. Rose. “Primitive specification and supporting documentation for SOBER-t32 submission to NESSIE.” In Proceedings of the First Open NESSIE Workshop, 13-14 November 2000, Heverlee, Belgium.

[26] C. L. Karr, et al., “Control of an exothermic chemical reaction suing fuzzy logic and genetic algorithms”, Proc. International Fuzzy System and Intellignet Control Conference, 1992, pp. 246-254.

[27] E. L. Key, “An analysis of the structure and complexity of nonlinear binary sequence generators,” IEEE Trans. Inform. Theory, vol. IT-22, no. 6, pp. 732-763, Nov. 1976.

[28] M. Krause, “Bdd-based cryptanalysis of keystream generators,” Advances in Cryptology – EUROCRYPT 2002, vol. 2332 of Lecture Notes in Computer Science, pp. 222–237 Springer-Verlag, 2002.

[29] P. V. Kumar and R. A. Scholtz, “Bounds on the linear span of bent sequences,” IEEE Trans. Inform. Theory, vol. IT-29, pp. 854-862, Nov. 1983.

[30] J. L. Massy, “Cryptography and system theory,” Proceeding 24th Allerton Conference Communication , Control, Comput., Oct. 1-3, 1986.

[31] R. L. McFarland, “A family difference sets in non-cyclic groups,” J. Combinatorial Theory, Ser. A 15, pp. 1-10, 1973.

[32] P. Nyffeler, Binare Automaten und ihre linearen Rekursionen, Ph. D. thesis, University of Berne, 1975.

[33] S. Petrovi and A. Fster-Sabater, “Cryptanalysis of the A5/2 algorithm,” Cryptology ePrint Archive, Report 2000/052, 2000. Available on http://eprint.iacr.org/.

[34] G. Rose, “A stream cipher based on linear feedback over GF(28),” In C. Boyd and E. Dawson, Editors, ACISP’98, Australian Conference on Information Security and Privacy, Springer-Verlag, July 1998, Vol. 1438.

[35] G. Rose, “SOBER: a stream cipher based on linear feedback over GF(28),” Preprint, 1999.

[36] G. Rose, “S16&S32:Fast stream ciphers based on linear feedback over GF(2n),” Preprint, 2000.

[37] G. Rose and P. Hawkes, “The t-class of SOBER stream ciphers.” Available on http://www.home.aone.net.au/qualcomm.

[38] R. A. Rueppel, Analysis and Design of Stream Ciphers, Berlin: Springer-Verlag, 1986.

[39] R. A. Rueppel and O. Staffelbach, “Products of sequences with maximum linear complexity,” IEEE Trans. Inform. Theory, vol. IT-33, no. 1, pp. 124-131, Jan. 1987.

[40] J. A. Serret, “Cours d’algebre superisure,” Tome II, p. 154, Gauthier-Villars, Paris, 1886.

[41] J. D. Schaffer, et al., “A study of control parameters affecting online performance of genetic algorithms for function optimization”, Proc. Third Int. Conf. On Genetic Algorithms, Fairfax, VA, June 1989, pp. 51-60.

[42] C. E. Shannon, “Communications theory of secrecy sustems”, Bell Sys. Tech. Jornal, Vol. 28, pp.656-715, 1949.

[43] T. Siegenthaler, “Correlation immunity of non-linear combining functions for cryptographic applications”, IEEE Trans. On Inform. Theory, IT-30, pp. 776-780, 1984.

[44] B. Smeets, “A note on sequences generated by clock-controlled shift registers,” in Lecture Notes in Computer Science 219; Advances in Cryptology: Proc. Eurocrypt ’85, F. Pichler, Ed., Linz, Austria, April 1985, pp. 40-42. Berlin: Springer-Verlag, 1986.

[45] R. Vogel, “On the linear complexity of cascaded sequences,” in Lecture Notes in Computer Science 209; Advances in Cryptology: Proc. Eurocrypt ’84, T. Beth, N. Cot, and I. Ingemarsson, Eds., Paris, France, April 9-11, 1984, pp. 99-109. Berlin: Springer-Verlag, 1985.

計畫成果自評

本計畫依原訂進度順利進行。除了相關文件的蒐集與研讀外，目前也得到了很豐碩的成果。我們針對擬隨機序列產生器的核心元件—布林函數，利用基因演算法物競天擇的特性設計出具有優良性質的布林函數。後面的附錄列出本計畫期間我們所發表的論文。

可供推廣之研發成果資料表

附錄

A. 計畫期間發表之論文

Journal Paper Published

1. J.-S. Hwu, R.-J. Chen, and Y.-B. Lin, “An Efficient Identity-based Cryptosystem for End-to-end Mobile Security”, to appear in IEEE Transactions on Wireless Communications.
2. J.-S. Hwu, S.-F. Hsu, Y.-B. Lin, and R.-J. Chen, “End-to-End Security Mechanisms for SMS”, to appear in International Journal of Security and Networks.
3. Jyh-Shyan Lin, Jen-Chun Chang, and Rong-Jaye Chen, "New Simple Constructions of Distance-Increasing Mappings from Binary Vectors to Permutations,", to appear in Information Processing Letters.

Conference Paper

1. Han-Chang Liang, Jen-Chun Chang, Rong-Jaye Chen, "New Efficient Constructions of Binary Asymmetric Error-Correcting Codes," International Computer Symposium, Dec 2004.
2. Kai-Chiun Huang, Jen-Chun Chang, Rong-Jaye Chen, "A new construction of resilient functions over GF(p) with good cryptographic properties," International Computer Symposium, Dec 2004.
3. J.-S. Hwu, R.-J. Chen, H.-S. Lue, and J.-S. Lin, “Efficient Computation of the Weil Pairing in ID-based Cryptosystems”, International Computer Symposium, 2004, pp. 1297-1301.
4. J.-S. Hwu, R.-J. Chen, and Y.-B. Lin, “Authenticated Public-Key Distribution over WLAN/Cellular Dual Networks”, International Conference on Information Technology: Research and Education, 2005.

5. W.-T. Liu, Cheng-Kai Chen, and Rong-Jaye Chen, "Experimental Linear Attacks on Substitution-Permutation Networks," Proceedings of the 15th National Conference on Information Security, Kaoshiung, Taiwan, 2005.
6. Han-Chang Liang, and Rong-Jaye Chen, "A Trichotomy Reaction Attack on McEliece Public-Key Cryptosystem," Proceedings of the 15th National Conference on Information Security, Kaoshiung, Taiwan, 2005.