國立交通大學

資訊科學系

碩 士 論 文

利用演化式計算找尋蛋白質結構之相似結構元

PRODEC：*PRO*tein structure motifs *D*etected by

*E*volutionary *C*omputing

研究生：陳音璇

指導教授：胡毓志　博士

中 華 民 國　九 十 四　年 六 月

利用演化式計算找尋蛋白質結構之相似結構元

PRODEC：*PRO*tein structure motifs *D*etected by

*E*volutionary *C*omputing

研 究 生：陳音璇　　Student：Yin-Hsuan Chen

指導教授：黃明經　　Advisor：Ming-Jing Hwang

胡毓志　　Yuh-Jyh Hu

國 立 交 通 大 學

資 訊 科 學 系

碩 士 論 文

A Thesis

Submitted to Department of Computer and Information Science

College of Electrical Engineering and Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer and Information Science

June 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年六月

利用演化式計算找尋蛋白質結構之相似結構元

研究生：陳音璇　　指導教授：黃明經博士

胡毓志博士

國立交通大學資訊科學研究所

摘要

同一分類蛋白質中的一群相似結構元，不但可以描述此分類的結構特性，對於蛋白質功能的分析也扮演著重要的角色。本論文提出了一個以演化式計算為基礎，並結合分群演算法的模型，來找尋蛋白質結構之相似結構元。透過演化運算子的運作，及適應性函數的導引，研究模型的確能自動尋找到結構非常相似的相似結構元。本研究以著名的蛋白質功能區域 – EF-hand 所屬的蛋白質分類為實驗對象，驗證本研究對於找尋同一分類蛋白質的相似結構元能力。本研究比以往的方法更自動、解決或規避了一些以往研究方法所面臨的困難處，並同時兼顧了合理的時間。本研究結果亦提出了一群結構相當相似的相似結構元，部分相似結構元還兼具了蛋白質功能及分類上的意義。

PRODEC：*PRO*tein structure motifs *D*etected by *E*volutionary *C*omputing

Student：Yin-Hsuan Chen　　Advisor：Dr. Ming-Jing Hwang
Dr. Yuh-Jyh Hu

Institute of Computer and Information Science
National Chiao Tung University
Hsinchu, Taiwan, 300, Republic of China

Abstarct

Large-scale functional annotations of proteins can be greatly aided by the identification of a set of motifs that characterize a specific SCOP fold. In this study we describe a new computational method, PRODEC (*PRO*tein structural motifs *D*etector using *E*volutionary *C*omputing), to automatically discover structure motifs in proteins. A key feature of PRODEC is that each PRODEC motif is a duo consisting of a sequence pattern and a structure pattern. PRODEC, based on genetic computation, begins with an initial population of random motifs. Through the evolutionary process, PRODEC iteratively improves the statistical significance of motifs by modifying their configurations. To evaluate each new pattern, a novel scoring function is developed that measure motifs conserved both in 1D sequence and 3D structure. At last, we provide a modified clustering method to refinement the final results. By evolutionary computing operators and clustering, PRODEC can automatically connect short or subtle motifs together and then extend them to longer ones. Tests indicate that PRODEC can successfully detect fold-specific conserved, flexible, and longer structural motifs. Comparing with conventional methods, PRODEC has better performance in finding flexible and long motifs than other motif discovery methods.

致謝

首先，要感謝黃教授、胡教授這兩年來對我的指導與鼓勵。由於老師平時的督促與專業知識的傳授，教導我學習及研究的方法，讓我對生物資訊領域有更深刻的了解。另外，還要感謝總是撥控解答我技術問題的元智資管葉老師，您豐富的學問、體貼的心，一直是我學習效法的對象。

這兩年一路走來，最終能順利畢業，得感謝中研院各同事在專業知識上不厭其煩的教誨，也感謝交大實驗室的宛嫻、美華、萬田三位學長姐在生活及專業上的關心鼓勵，還有勁伍、昀君、秉蔚、世彥及電信所同學秀琴等諸位同學，陪我一起度過每一次的歡笑與難關，也感謝博班學長們、豐茂、登貴、貫中等可愛的學弟們，為我生活帶來了許多歡笑。充實的兩年，因為你們而更豐富、更精采。

再來，還要感謝大學同學千華、美惠、Steve、學長傳詔、學長博群、學長展熙、學姊昭慧，你們的專業技術指導外，更是我不定期歡樂的來源，因為你們，讓我研究生生涯看到更廣、歡笑更多。

最後，最感激的是遠在高雄但永遠默默支持我的家人，精神上的鼓勵與安慰，讓我擁有勇氣去度過每一個難關。還要感謝男朋友洪懷謙，體貼我一路走來的辛苦，包容我壓力大時的無理取鬧。

今日的我順利畢業，謝謝大家!!

目錄

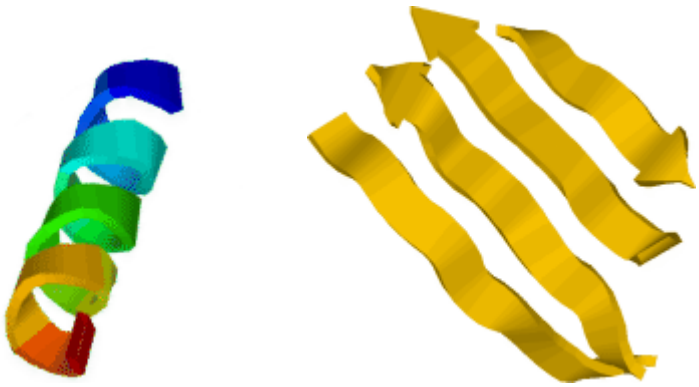
第一章　緒論

1.1 蛋白質的基本性質與結構

蛋白質是由 20 種胺基酸(amino acid)所構成。這些胺基酸藉由鍵結相連組成多胜肽鏈（polypeptides），而在立體空間上摺疊後產生複雜的立體結構。依照她們結構上的複雜度，可以將蛋白質的結構分成四個成次。依簡單到複雜各為蛋白質的一級結構(primary structure)、二級結構(secondary structure)、三級結構(tertiary structure)、及四級結構(quaternary structure)。

蛋白質一級結構(primary structure)是指胺基酸的序列，其中一端稱為 N-端(-NH2，胺基)，另一端則稱為 C-端(-COOH，梭基)。因為各級結構的訊息都決定於胺基酸的序列，因此不同的序列決定了最後蛋白質的形狀。

蛋白質二級結構(secondary structure)是蛋白質骨架(多胜肽鏈)氫鍵結排列。胜肽骨架中鍵的性質扮演著重要的腳色，在每一個胺基酸內部在骨幹上有兩個鍵可相當程度的旋轉，不同的旋轉角度造成了不同的蛋白質結構。而此胺基酸長鏈繞成的二級結構形狀是相當規律的，其原因是因藉著胜肽鍵(peptide bond)的雙鍵性質形成胜鍵平面；又因其構造內各原子之間特殊的引力或斥力使得兩相鄰胜鍵平面間的轉動限制在一定角度範圍，而造成規律的兩種主要結構，一為α螺旋(α-helix)及β褶板(β-sheet)。圖 1-1 為兩者的真實形狀：



(a)　　(b)

圖 1-1 蛋白質二級結構 (a) α-helix　(b) β-sheet

蛋白質的三級結構(tertiary structure)是胺基酸分子內所有原子的三度空間排列，也就是各個二級結構彼此再相互組合所構成的完整立體空間中的三級結構，其構成的作用力有離子鍵、氫鍵、疏水鍵、金屬離子等；其中疏水鍵對水溶性蛋白質三級結構之穩定性貢獻相當大，它可以穩定地包埋在分子內部，以維持蛋白質的完整三級結構。

蛋白質的四級結構(quaternary structure)是超過一條以上多胜肽鍵蛋白質的性質。很多蛋白質的三級結構本身即為獨立且具有活性的分子，但有些則要再加上輔脢、輔因子(如金屬離子)或輔基(prosthetic group)，有些則以再以醣分子修飾成為醣蛋白(glycoprotein)或是要接脂質成為脂蛋白(lipoprotein)，甚至更有的要再與其它相同或不同的蛋白質分子結合而形成四級結構，最典型的如血紅蛋白(hemoglobin)。構成四級結構的每一單位分子，稱為次體(subunit)，通常各次體之間是以二級鍵(interaction forces)為主要結合力量，像是許多病毒的蛋白質外殼，就是由具有規律而巨大的四級結構所組成。

1.2 蛋白質相似結構元(Protein Motif)的重要性

1.2.1 何謂 Protein Motif？

蛋白質相似結構元(Motif)目前並無統一的定義，但大致上蛋白質不同子領域的科學家、生物學家對其有著共同的認知，所謂的 Motif 是描述蛋白質某個特定觀點的具有相同或類似功能的小單位。蛋白質雖然看似複雜，細究之下實為有規則性存在。生物學家將這些存在的規則性統一歸納以便更深入的研究或是作為其他應用。其中，Motif 可以就蛋白質功能、結構及序列三方面大致分類。

以結構的角度來觀察，雖然不同的蛋白質有不同的結構，但若將蛋白質分解為數個小立體結構，可以發現不同的蛋白質會有相同或相似的子結構，二級結構的α螺旋及β褶板是最典型的例子。

Motif 就功能面來觀察，蛋白質是維持生物體運作的最重要物質，不同的蛋白質有各自不同的功能，各司其職且又彼此互相合作完成生物體中的代謝、生殖、生長、活動等等。單一生物活動通常需要大量不同功能的蛋白質參與，由於分工細膩，這群蛋白質中會有部分蛋白質是會具有非常相似或相同的功能；又，不同物種生物體上也會有著相同或相似的生化功能。這些負責著相似或相同生化功能的不同蛋白質，雖然立體結構或胺基酸序列不一定相同，就功能面上它們也可以歸類於同一分群。

就胺基酸序列而言是最單純的。單純觀察胺基酸序列上胺基酸的排序，某些在蛋白質演化過程中屬於近親的兩蛋白質通常會有相似的胺基酸序列，縱使不是近親的兩個蛋白質，也是有片段序列相似的可能性存在。

1.2.2 Protein Motif 的重要性

分析蛋白質功能、結構、或是最單純的胺基酸序列，往往是成本非常高昂的。例如，蛋白質結構的解構主要是以核磁共振(Nuclear Magnetic Resonance, NMR)與 X-ray 晶體繞射技術(X-ray Diffraction)實驗測量出來的。雖然實驗的技術不斷進步，但不管是時間或金錢實驗成本仍然很高，無法做大規模的蛋白質結構測量；另外又如蛋白質的功能，往往必須是透過大量實驗或是臨床應用等等實務經驗的累積，才能知曉其確切功能。在這種情況下，有許多學者希望能透過其他較便宜、快速的方法來推斷新蛋白質功能與結構，或是更深入了解研究中的蛋白質。因此，Protein Motif 的出現便提供了生物學家很好的解決之道。往後面對新發現或是人工合成的蛋白質，便可以根據這些歸納出來找到的 Motif 快速地對該蛋白質有初步預估及認識。

1.2.3 Protein Motif 研究的困難處

相較於基因體定序研究，蛋白質體的議題到目前為止仍是非常複雜難解的。蛋白質由胺基酸序列構成最基本元素，胺基酸不同的順序構成了不同的立體空間結構，不同的結構有著不同的生物化性，三者彼此息息相關但其相關性卻又不容易清楚界定。相似的胺基酸序列可能有著不同的立體結構，相似的立體結構也有可能有著不同的生物化性，換個角度來看，相似的生物化性可能有著不同的立體結構，不同的立體結構也有可能有著不同的胺基酸序列。三者之間的關聯性沒有絕對的規則可循，就算是勉強歸納出規則，卻往往有著太多的例外。

三者之間微弱的關係造成了複雜且難以解決的研究議題。圖 1-2 為三種不同層面的 Motif 之間的關聯圖；

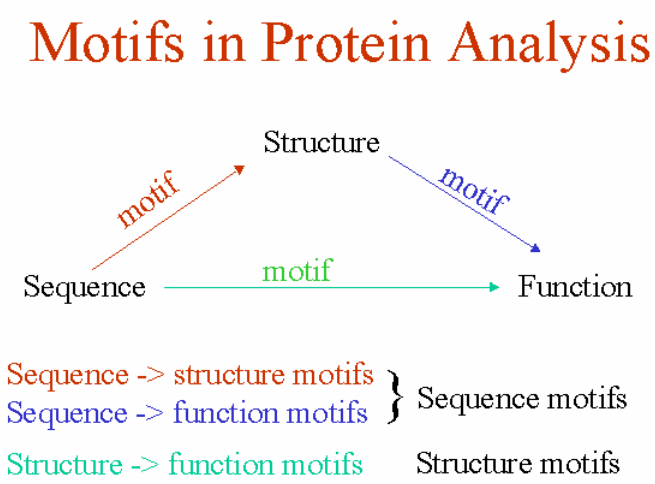


圖 1-2 Motifs in Protein Analysis

出自：http://www.ii.uib.no/~inge/talks/ebi-nov-99/sld009.htm

1.3 研究目標

隨著蛋白質資料庫(Protein Data Bank) [*Berman,et al,* 2000] 中紀錄的蛋白質量日益龐大，生物學家對於快速、簡單、便宜的分析蛋白質方法需要性越來越急迫。值得慶幸的是，越來越多被分析出來的蛋白質提供了大量有用的資訊，這些資料透過系統化、有條理的歸納整理後，能指引著生物學家在面對新蛋白質茫茫不知所措時，能有個粗略的分析方向或是基本資訊。

本研究將研究範圍鎖定至蛋白質結構與胺基酸序列之間的關聯性，期望能找到一些非常有彈性的規則來描述兩者之間微弱的關聯性，因此，在本文中，將 Protein Motif 定義為蛋白質相似結構元，目的是找出結構類似的子結構，並探討其和胺基酸序列之間的關係。

另外，由於目前已經知道結構的蛋白質有三萬個左右，三萬條胺基酸及三萬個結構之間的關係，可想而知一定是非常的複雜，要找到通則是非常的困難。因此，為了尋找出真正有意義且實用的相似結構元，本研究將以同類蛋白質為實驗對象，蛋白質的分類依據則遵循 SCOP(Structure Classification of Proteins)[*Murzin et al,* 1995]的分類。

本研究的目標是在一群同類的蛋白質中，找到一群可以描述此類蛋白質特性，且結構相似、長度較長的相似結構元，並具有蛋白質功能上的意義或是其他的應用。

1.4 論文架構

第一章我們介紹蛋白質結構上的基本性質、何謂 Protein Motifs 及其重要性、以及本研究的研究目標。

第二章則回顧過去與本論文研究相關的文獻、評量蛋白質相似結構元的方法、

第三章討論本研究所提出之方法。系統模型可分為前置作業模型、基因規劃模型、以及後至作業模型等三個子模型，詳細介紹每個步驟做法以及每個階段的成果。

第四章介紹實作時所採用的實驗資料、分析實驗資料特性、及本研究結果與其他相關研究比較分析。

第五章分析本研究在不同層面上的貢獻及缺點、結論、以及未來展望。

第二章　文獻探討

2.1 蛋白質相似結構元找尋法概述

2.1.1 Sequence / Structure Alignment

胺基酸序列或是蛋白質結構的比對，是最基本、簡單且行之有年的做法，概念來自於 DNA 序列的比對。胺基酸序列比對發展的早，原理是將所要比對的胺基酸序列才列在一起，透過不同的序列比對(sequence alignment)演算法，並考慮胺基酸彼此之間的化性、物性、及分類，找出序列相似的區域，並以不同的方式呈現出來，構成所謂的 Motif。圖 2-1 為有名的 ClustalW[*Gibson et al*, 1994]序列比對工具的實際比對圖：

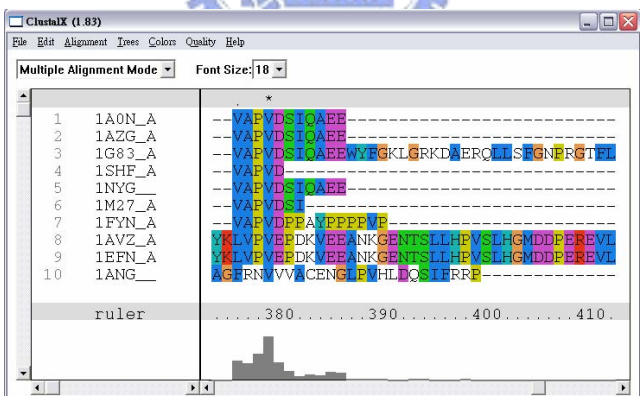


圖 2-1 胺基酸序列比對工具 ClustalW

蛋白質結構的比對因為牽涉到空間座標所以較為複雜。蛋白質結構資料中會紀錄每個原子在空間的 X 軸、Y 軸、Z 軸座標位置，但因在蛋白質解構的過程中並無統一的座標系統，所以在結構比對前必須先將預比對的結構全部轉換至同一個參考座標，通常是取蛋白質的質心位置，再調整質心位置為同一個座標位置，

其他原子再與以位移，此步驟稱為 Superimpose。堆疊好預比對的結構後，先找出相應的原子，再計算相應原子的位置差距。有名的 Swiss-Port 資料庫有提供相關工具來方便使用者做蛋白質結構比對。

不管是胺基酸序列或是蛋白質比對，最大的問題發生在預比對的對象超過兩個，即所謂的 Multiple Alignment。共同的問題是比對的順序不同，會影響最後比對的結果。另外，結構比對因為是立體空間，原先制定的座標系統方向並無統一，導致原本比對不像的結構，很可能只要轉個角度就比對上了。

因此，單純的比對雖然簡單、方便，但是因為比對演算法本身的一些限制導致比對出來的結果並不是很精確，除了不斷改善最基礎的比對方法外，科學家們更是努力的尋找非比對演算法概念的其他方法。

2.1.2 Structure Alphabet

結構代號(Structure Alphabet)是一建立在序列比對的概念上，融入部分的空間立體結構資訊，是介於胺基酸序列比對及蛋白質結構比對之間的方法。胺基酸總共有 20 種，不同的胺基酸代號隱含著胺基酸的化性、物性，不同的胺基酸代號依序著發生次序一字排開而形成了蛋白質最基本的胺基酸序列。以此概念為基礎，將紀錄胺基酸化性物性的代號，改變成紀錄空間結構的另一組代號並依序排開，就形成了一條紀錄蛋白質空間結構的序列，最後再將轉換好的兩條或多條結構代號序列比對，找出序列相似的區域，即所謂結構的相似區域。

相關研究例如 SA-Search [*Etchebest et al*, 2005]、Foldzilla[*Hwang et al*, 2004]等資料庫，即以自訂的結構代號轉換原始的胺基酸序列，再將多條的結構代號序列進行比對得到最後的相似結構元。

此方法的優點在於將原本非常複雜的結構比對，透過結構代號的轉換，比對難度降低至序列比對。最後比對的成功與否，除了序列比對演算法本身的優缺點外，最重要的是結構代號的轉換。執行結構代號轉換前，必須將蛋白質結構區段與以分群，分群後的群數及內容主宰了最後的相似結構元品質。當分群的群數越多，也就表示小結構種類越多，那麼最後相似結構元的精準度就會較高，相對的較少結構種類的結構代號會造成相似結構元不夠精準。隨著精準度的要求提高，小結構分群的複雜性就越高，所需要的成本就越多。

2.1.3 Clustering

分群法(Clustering)是電腦科學中發展已久的演算法。概念是透過物件彼此之間的距離衡量，將距離小的、也就是性質相似的歸為一類，非常適合來將胺基酸序列或是蛋白質結構相似的小片段劃分在同一群中。

分群法成敗的關鍵在於序列或結構之間的距離衡量方式，另外，分群依據的特性選擇與順序也非常重要。以胺基酸序列及蛋白質結構的為例，可以僅以序列或結構特性來分群，也可以將兩者特性順序前後不同來分群。蛋白質結構預測領域中非常著名的 David Baker，在 1998 年提出的模型便是先將相似的胺基酸序列分群，分群好的序列再每一群自己以結構資訊再分群[*Baker et al,* 1998]。

另一著名的研究為 Wangikar 等人針對長度為 8 個胺基酸的相似結構元，篩選 56 個蛋白質相關屬性進行 Clustering [*Wangikar et al*, 2004]。

此類方法優點是能避免掉序列或結構比對的缺點，分群時所需的蛋白質特性是依據研究者自己的認知來挑選。但，分群法傳統的一些問題在此應用上也無法

避免。例如，必須先事先決定分群的群數，這就和尋找相似結構元的背景知識相衝突，在分群前是無法知道此群蛋白質中到底有多少相似結構元。

2.1.4 Discovery

隨著被找出來的蛋白質相似結構元越來越多時，此領域發展的較後期出現了有別於前述三者的方法。聰明的科學家將眾多相似結構元資料庫予以整理歸納，找出形成相似結構元的一些規則，再加上生物背景知識的支援，訂定出有可能的相似結構元模型(templates)，再計算各個模型在蛋白質中出現的次數、結構相似度等分數，來區別模型的好壞。好的模型經過整理合併，便可以得出最後的相似結構元。最有名的是 2002 年的 TRILOGY [*Bradely et al*, 2002]，找到的相似結構元兼具了胺基酸突變的現象，立體結構上也非常的相似。

此種以先制定模型的方式來尋找相似結構元，的確避免了前面三種方法的缺點，但模型訂定的方式和數量卻成為新的關鍵點。除非是考慮詳細周延、且數量夠大的模型，否則最後找得到的相似結構元數量一定會限制於模型的內容，非常容易忽略掉沒有模型但實際存在的相似結構元。另外，模型的訂定也不可能太大，擴充性較差。以 TRILOGY 為例，模型是以三個胺基酸為基礎，若要加大的話就得有碰巧可以合併的模型出現才能合併為包含四個胺基酸以上的相似結構元。

2.2 蛋白質結構相似度衡量方法

基於蛋白質結構的複雜性，結構相似程度是個很難精準衡量的數據。過去十多年來的發展與修正，有了較為大眾信賴且採用的標準出現，其一是以原子之間的距離來計算，另一個是蛋白質主幹上的旋轉角度為依據。

2.2.1 Atom Distance

原子彼此之間的距離是最直覺的衡量方式，但衡量過程稍嫌複雜，最廣為人知的就是 Root Mean Square Deviation，簡稱 RMSD。如前面所述，衡量距離前一定要有同樣的參考座標，所以必須將所有原子的座標做重疊(superimpose)的動作，重疊一起後蛋白質開始旋轉，直到找到最多對應到的原子為止。最後再將對應到的原子兩兩計算座標軸上的距離。公式(1)為 RMSD 的計算方法；

$$RMSD_{\alpha\beta}^2 = \frac{1}{N}\sum_{i=1}^{N}\left(r_{\alpha,i} - Qr_{\beta,i}\right)^2 = R_{g\alpha}^2 + R_{g\beta}^2 - 2\left(\frac{\sum_{i=1}^{N} r_{\alpha,i}\cdot Qr_{\beta,i}}{\sqrt{\sum_{i=1}^{N} r_{\alpha,i}^2 \cdot \sum_{i=1}^{N} r_{\beta,i}^2}}\right) R_{g\alpha}R_{g\beta} \quad (1)$$

其中 N 為原子數目，Q 為 Rotation Matrix，也就是執行 Superimpose 時原子需要位移的量，α、β分別代表兩個不同原子，R 為兩原子之間的距離。

RMSD 值越小，代表原子重疊在一起後的距離越小，也就是結構越相似，但不同長度的胺基酸序列其包含的原子數目不一樣，會影響 RMSD 的公平性。若不同長度的胺基酸序列，單從 RMSD 值的大小來斷定 RMSD 比較大的結構比較像，而沒有考慮序列長短因素，是非常不公平的。本研究除了引用 RMSD 作為衡量結構相似度的標準外，由於所找到的相似結構元長度不一定相等，為了評分

公平，因此多引用另一個指標，ACC，好更精準的去比較不同長度序列的結構相似程度。

ACC 是由 Skolnick 在 2001 年提出。ACC 是 RMSD 去除掉長度因素的值，概念是蒐集不等長度的區段結構並計算其 RMSD，當樣本數夠大時，便能描繪出特定長度 RMSD 的統計曲線，得知 RMSD 平均值及變異數，有了這些資料後，就能輕易的將任何長度的相似結構元，比較其 RMSD 在同等長度下的表現。ACC 值介於 0 到 1 之間，0 是最差，1 是最好，也就是說當某相似結構元的 ACC 值為 1，表示其結構在同等結構元中是非常相像，0 則完全不像。若是兩個不同長度的相似結構元要比較哪一個結構相似度較高，ACC 的值也能馬上與以辨識。圖 2-2 不同長度下 ACC 值大於 0.98 的機率[*Skolnick et al*, 2001]：

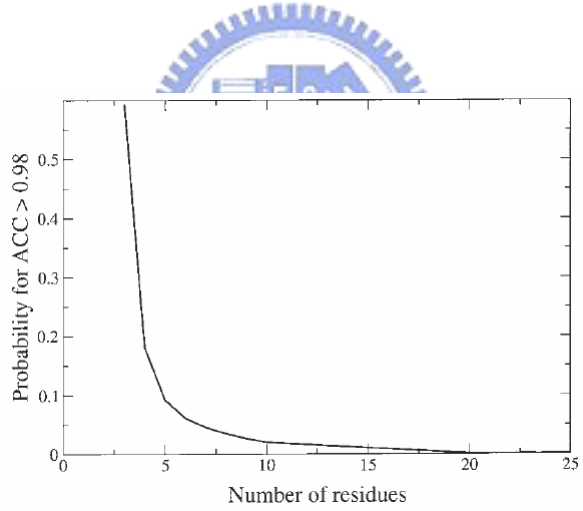


圖 2-2　不同長度之相似結構元，其 ACC>0.98 的機率

由此圖可以觀察到當結構元長度大於 5 時，其 ACC>0.98 的機率快速遞減，也就是當結構元長度大於 5 個胺基酸時，要找到結構相似的機率是很低，且隨著長度遞增，結構相似程度高的機率就越低。

2.2.2 Torsion Angle

旋轉角(Torsion Angle)指的是蛋白質結構主幹上，Cα 和兩邊相連接的原子 C 及 N 之平面所夾的角度，不同的角度會造成不同形狀的蛋白質結構。連接 N 原子的稱為 Φ(phi)，C 端的為 Ψ(psi)，圖 2-3 為兩個夾角的示意圖：

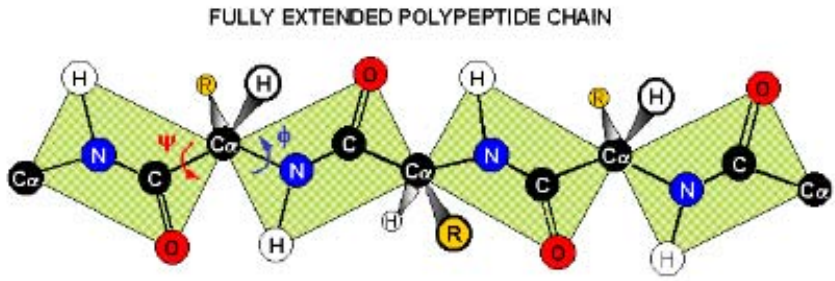


圖 2-3Φ(phi)、Ψ(psi) angle

旋轉角角度左右了蛋白質主幹的結構，知名生物學家，俄國人 G. N. Ramachandran [*Ramachandran et al*, 1977]統計了不同結構的旋轉角，發現大部分結構的旋轉角會落在特定的區域，圖 2-4 為 α-helix 及 β-sheet 兩種結構的旋轉角度落點，有明顯的差異。

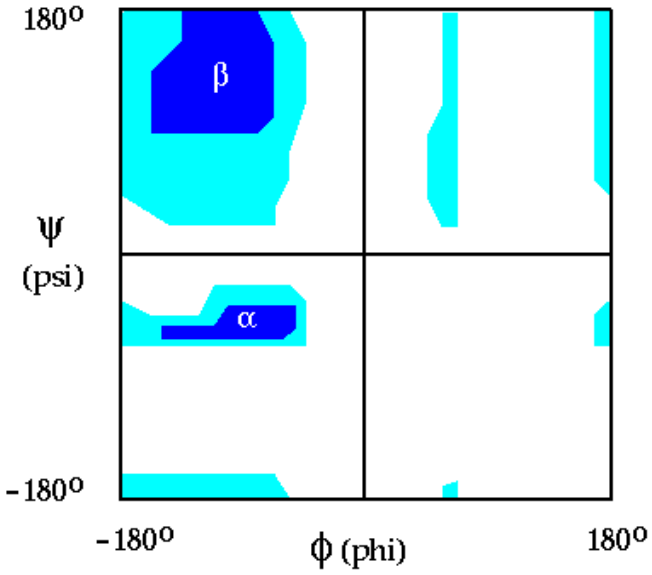


圖 2-4 Ramachandran Plot

單以旋轉角度的一致性作為衡量結構相似度標準的研究較少，通常是以

RMSD 為主，旋轉角為輔。例如，David Baker 在 1998 年提出那有名的 I-sites 資料庫，在挑選每個分群所代表的相似結構元時，便引入 mda 及 dme 兩個標準。dme 其實就是 RMSD，mda 便是旋轉角的差異值，公式(2)和公式(3)分別為 I-sites 中兩者的計算方式：

$$dme = \sqrt{\frac{\sum_{i=1}^{L}\sum_{j=i-5}^{i+5}\left(\alpha_{i\to j}^{s1} - \alpha_{i\to j}^{s2}\right)^2}{N}} \tag{2}$$

$$mda(L) = \max_{i=1,L-1}\left(\Delta\Phi_{i-1}, \Delta\Psi_i\right) \tag{3}$$

本研究在前半段會以旋轉角相似度為主要標準，RMSD 及 ACC 為次要標準，後半段則以 ACC 為主要篩選標準。

第三章　方法論

3.1 模型設計的目的與概念

找出一群蛋白質共有的相似結構，隨著蛋白質數量的增加而變成非常困難。舉例來說，單純只看序列長度皆為 100 的兩條蛋白質，欲找長度為 3 的相似結構元(Motif)，搜尋的次數就高達 $\sum_{k=2}^{98\times 2} C_k^{98\times 2}$ ，約等於 $2^{98\times 2}$ 可觀的搜尋空間(Search Space)。因此，若有 M 條平均長度 N 的蛋白質序列，尋找長度 3 的相似結構元將會是：

$$\sum_{k=2}^{X} C_k^X = 2^X \text{，} X = (N-2)\times M$$

由此可知，蛋白質數量越多、蛋白質長度越長、及欲尋找的相似結構元長度越長時，尋找的困難度將會非常的高，因此我們需要一個能有效減小搜尋空間、提高搜尋效率、並能正確計算結構相似度的演算法。

本研究的目的在於希望能找出某特定一個蛋白質分類中，一組能描述此分類特徵的相似結構元。圖 3-1 為被分在同一分類的蛋白質與其相似結構元之間關係的示意圖，右邊橢圓代表一群蛋白質，以系統角度而言即為搜尋空間(Search Space) ，橢圓上的小方形則為存在的相似結構元，也是本研究欲找到的解。

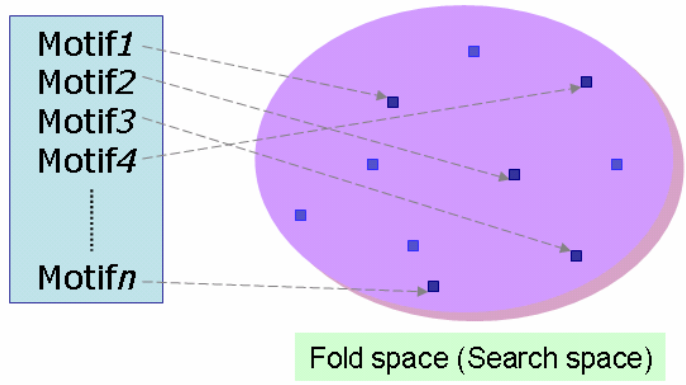


圖 3-1 相同分類蛋白質及其共同結構相似元關係示意圖

由於相似結構元的數目及散落位置是未知的，而搜尋空間在 3.1 章討論中可知是非常的大，為了有效找出散落各處的相似結構元，本研究採用演化式計算(Evolutionary Computing)中的基因規劃法(Genetic Programming)。基因規劃法簡稱 GP，和耳熟能詳的基因演算法(Genetic Algorithm)極為類似，同屬於監督式學習(Supervised learning)的演算法，透過學習的方式來獲得最佳答案。

基因規劃法在 1992 年由 John Koza [John Koza, 1992] 首先提出，基本架構遵循演化式計算的精神，以生物演化過程中的「物競天擇，適者生存」概念為理論基礎，跳脫過去鑽研高深數學、發展複雜算式來解決問題的模型，製造讓各種可能解互相競爭的機制，模擬生物進化過程，讓系統能自動演化出最佳解。此演算法的優點在於能解決日益複雜的各種問題，設定的限制較少所以能夠更貼近複雜的真實世界。缺點是會耗費巨大的演算資源，最後找到的解答有時不能證明是全域最佳解(Global Optimal)，尤其在 NP-hard 等類型問題中更是困難。不過只要透過細心調控的演化，通常能得到近似最佳解。在處理搜尋答案空間過大的問題時，基因規劃法提供了極佳的解決途徑。

3.2 蛋白質結構語言定義

蛋白質共同結構元的表達語言定義目前並無統一規定，過去的研究提出了好幾種不同的表達語言。例如：PROSITE 資料庫中定義的語法，簡潔又容易辨認，但記載資訊較少；以 Matrix 格式來記載較多資訊的表達方式，較複雜且不好直覺辨認；或是以轉換過之二級結構序列來記載。其中，以 PROSITE 定義之相似結構元表達語言最為熟知且接受，且符合本研究要找尋相似結構元類型的要求，因此遵循 PROSITE 所定義的表達語言，作為本研究之相似結構元表達語言的標準。

3.2.1 相似結構元語言定義

RPOSITE 定義相似結構元的表達語言極為簡單，在 PROSITE 網站中提供的 porsuser.txt 檔案中有詳細的說明，如下：

1. 使用 IUPAC one-letter codes 來表達 20 個慣用的胺基酸。
2. 符號 'x' 表示該位置接受 20 個胺基酸中的任一種。
3. 符號 '[ ]' 表示該位置接受括號中所包含的所有 IUPAC one-letter 表示的胺基酸。例如：[ALT] 表示該位置可以是 Ala、Leu 或是 Thr。
4. 符號 '{ }' 表示該位置**不**接受括號中所包含的所有 IUPAC one-letter 表示的胺基酸。例如：{AM}表示該位置不可以為 Ala 和 Met，意即可以是 Ala 和 Met 以外的任何胺基酸。
5. 符號 '-' 來區隔相似結構元中相鄰的位置。
6. 符號 '(N)' 或 '(N_1, N_2)' 中所包含的 N 數字，表示其前面相連相似結構元該位置型態重複的次數或重複次數的範圍。例如：x(3) 表示 x-x-x，x(2,4)表示 x-x、x-x-x、x-x-x-x。

表 3-1 為實際例子，表的下欄部分為符合相似結構元的胺基酸序列：

Motif	[AT]-O(2)-[KIJHG]-{LPQAVNM}-x(1,2)-D		
	TOOKIPPD	TOOHGJJD	AOOIGWID
Possible A.A. Sequence	TOOGDMND	TOOKUFED	AOOJCAWD
	TOOJPWD	TOOGUHHD	AOOHIJKD

表 3-1 相似結構元 s 定義語言及其合法之胺基酸序列

3.2.2 相似結構元在系統中之語言定義

上述遵循 PROSITE 的相似結構元表達語言，並無法直接編碼為系統可辨認且執行的語言，因此另外定義了系統可執行的語言方式，以利整個模型的操作。

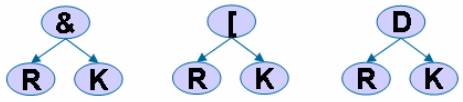
本研究採用樹狀結構(Tree)的資料結構來編譯相似結構元，樹狀結構中的節點(node)記載了相似結構元之結構及胺基酸型態。節點又可分為中間節點(internal node)及末端節點(terminal node)，各記錄著不同資訊。表 3-2 為不同節點分別可記錄的資訊：

Node Types	可記錄之資訊
Internal Node	'[ ]'、'{ }'、'&'
Terminal Node	IUPAC alphabet

表 3-2 相似結構元系統定義語言的樹狀結構中，不同節點所記載的資訊

中間節點紀錄的資訊為符號'[ ]'、'{ }'、及'&'，前兩者代表的意義和 PROSITE

定義的相同，最後一個則是連接各個位置的相似結構元及樹的結構。另外，不論是紀錄哪種資訊的中間節點，皆有兩支腳來連接其下一層的子結點，且中間節點可以接中間節點或末端節點，末端節點已位在樹的底部所以不能再接子節點。圖 3-2 為舉例說明：



R-K　　[RK]　　R-D-K

圖 3-2 中間節點連接子結點示意圖

轉換樹狀結構語言至 PROSITE 相似結構元表達語言的方法，遵守 Pre-Order 的規則，從樹的左下角節點開始，由左至右，由下至上，結束於樹的最右下方節點，非常直覺且可以輕鬆在兩種系統語言和相似結構元語言中自由轉換。圖 3-3 為解碼樹結構的相似結構元的示意圖：

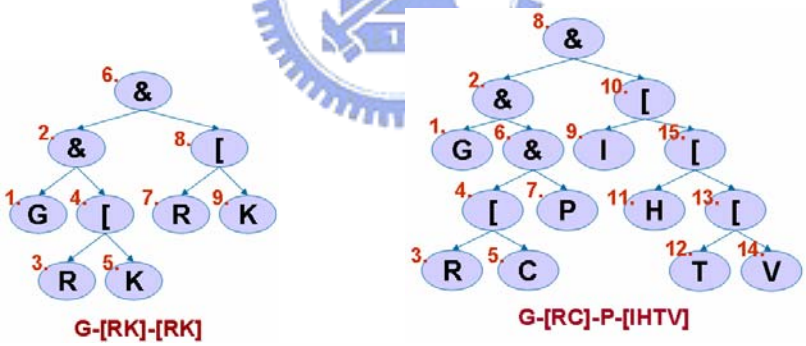


圖 3-3 樹狀結構的相似結構元語言轉譯為正規相似結構元語言

3.2.3 語言的優點

本研究所定義的相似結構元表達語言遵循 PROSITE 訂定的標準表達語言 (Standard Expression Method)，不但廣泛被使用也非常容易被人理解，使用者能非常迅速理解本研究的結果。

另外以樹狀結構為基礎的系統中相似結構元表達語言，擷取了樹狀結構非常彈性的優點，巧妙地利用不同的樹狀結構，可以發展出結構較為複雜的相似結構元，相較於以其他資料結構編譯的相似結構元，例如：堆疊(Stack)、串連(Link List)等有著更大的可發展性及變化性。此也為本研究能較過去非序列排比方式(sequence alignment)找尋相似結構元的方法，例如 TRILOGY、I-sites 等，更有能力找到較為複雜相似結構元的重要原因。

3.3 模型架構

本研究的模型可分為三個子模型，分別為實驗資料(Input Data)的前置作業、中間部分的基因規劃法(Genetic Programming)、及最後相似結構元整理(Motif Refinement)的後置作業。圖 3-4 為模型的架構圖，其中包含了三個子模型：

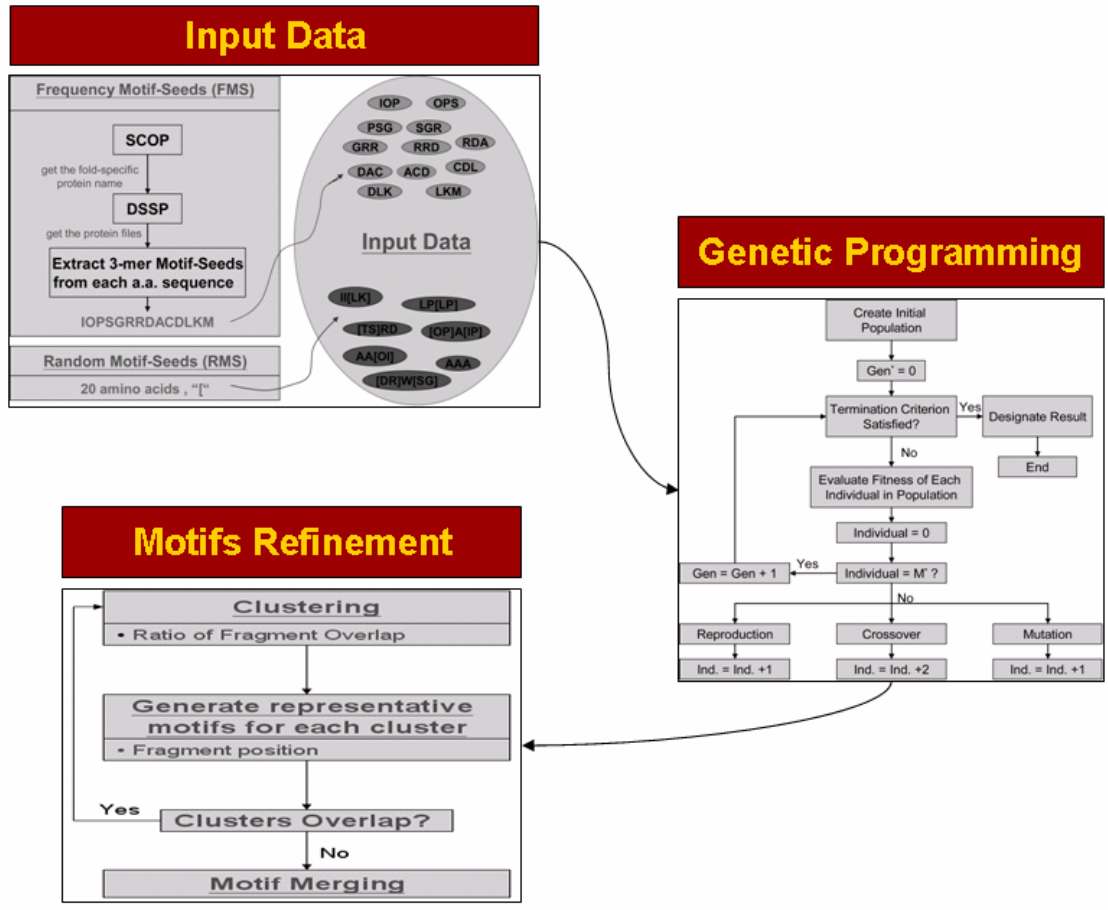


圖 3-4 本研究模型總流程圖

實驗資料模型為準備系統需要的各種資料，並輸入接續的基因規劃法模型中尋找相似結構元，最後後置作業模型將找到的相似結構元做最後的整頓及處理。各個子模型將會在後面小節詳細敘述。

3.3.1 前置作業

這個小節中，將介紹本研究模型所需的實驗資料格式、來源、以及收到這些資訊後的處理方法。

本子模型的來源資料可分為 Frequency-Motif Seeds (FMS) 及 Random-Motif Seeds (RMS)兩個部份，兩者代表意義及特性將在後面將以詳細說明。圖 3-5 為實驗資料子模型的流程示意圖：

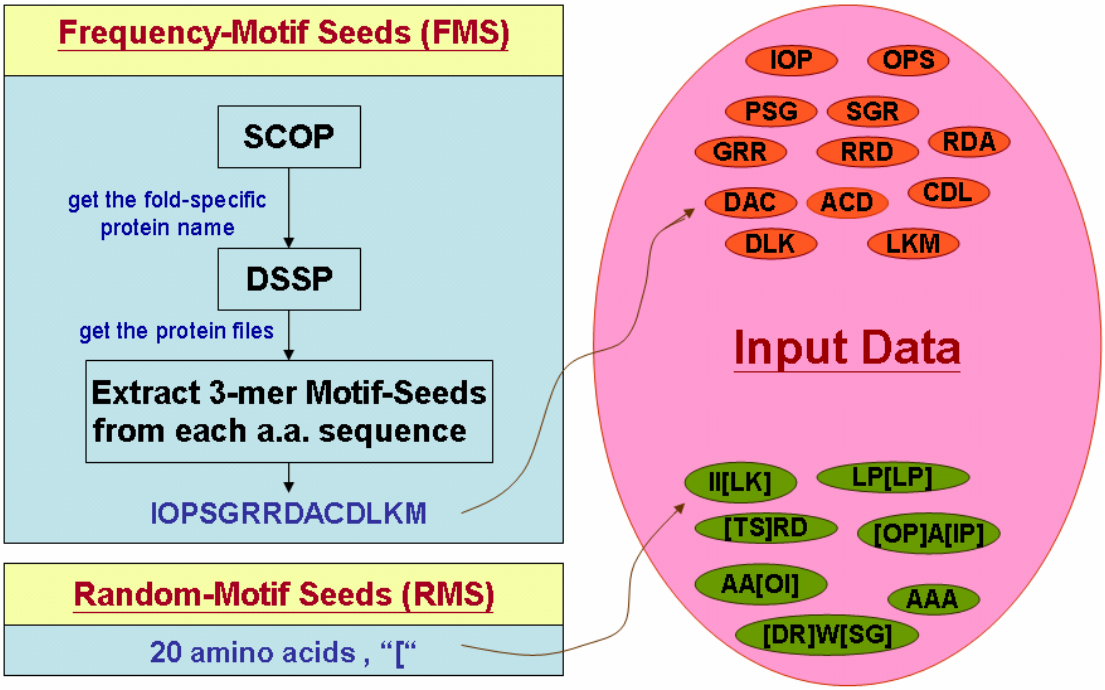


圖 3-5 Input Data 子模型流程圖

本研究的目的是希望能找出一個蛋白質分類中具有代表性的一群相似結構元，因此參考 SCOP 資料庫中對蛋白質的分類。SCOP 對於蛋白質分類總共有四個層級，從上到下依序為 Class、Fold、Superfamily、及 Family 等四層，本研究以第二層級- Fold 為分類標準。選擇 Fold 層級的理由是依據 SCOP 對於四個分類等級的定義，文獻中提到屬於同一 Fold 的蛋白質會有相似的主要結構，上一層

的 Class，分類的結構標準太寬鬆粗略，下一層的 Superfamily 則嚴謹至同原(homologous)的蛋白質功能(Protein Function)或結構。因此選擇 Fold 為標準是最符合本研究之需求。

取得了屬於同一 Fold 的蛋白質 PDB ID 資料後，便根據 PDB ID 至 DSSP 資料庫中下載完整的蛋白質資料。選擇下載 DSSP 資料庫的原因是系統在找尋答案的過程中會使用到蛋白質主幹的扭轉角(Torsion Angle)的資訊，DSSP 檔案格式中便包含了已算好的扭轉角數據，不需自行再計算，節省系統建置時程並可避免自行計算可能發生的錯誤。

取得了以上屬於同一 Fold 的蛋白質資料後，每一條蛋白質序列逐一掃描，以長度 3 為單位，一次位移一個胺基酸，取出不同的小結構片段，即為 Frequency-Motif Seeds (FMS)。圖 3-6 為 FMS 擷取過程示意圖：

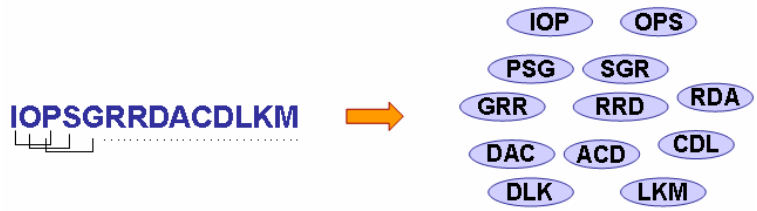


表 3-6 FMS 從蛋白質序列擷取過程示意圖

Random-Motif Seeds (RMS)則是系統隨機產生。根據前面章節 3.2.2 系統相似結構元語言定義中，系統隨機產生再翻譯出來的相似結構元有較大的機會是包含 '[ ]'、'{ }' 符號，較複雜且具有彈性，能包含胺基酸突變的情形。

FMS 和 RMS 各自有其對系統的影響及貢獻。FMS 是已經存在於目標蛋白質中的較小相似結構元，提供了基因規劃法較佳的搜尋起始點，能縮短系統整個

搜尋時程，提高效率。RMS 則是提供了相似結構元各種變異的可能性，使得最後找到的相似結構元能有機會包含蛋白質演化過程中胺基酸突變(Mutation)的情形。表 3-3 為 FMS、RMS 功能整理表：

	FMS	RMS
Source	Real protein sequences	Random
Objective	provide good starting search point	provide motif-evolution potential
Contributions	speed up the system	consider a.a. mutation situation

表 3-3 FMS 與 RMS 比較圖

實驗資料包含 FMS、RMS 除了上述的優點外，另外兩者合併使用，會擴大互相競爭生存的解數量，激烈的競爭往往能提升最後解的品質。這也符合了過去生物演化過程所觀察到的現象：太相近的物種互相繁殖，容易有家族疾病；不同人種或物種交配，繁衍的後代通常品質會較好，變異性較高，較能抵抗突然而來的意外狀況。基因規劃法的研究中，也有部份學者致力於探討互相競爭的解數量和最後找到解之間的關係，研究結果顯示，基因規劃法的確在演化過程中也遵守著自然界的現象，當互相競爭的解的數量越多，找到最佳解的速度也會越快 。

3.3.2 基因規劃法

基因規劃的概念是針對一個可能解的族群(putative solution)，透過演化中交換及突變等過程尋找最佳解。圖 3-7 為基因規劃法的流程圖，大致上和其他演化式計算的流程相同。

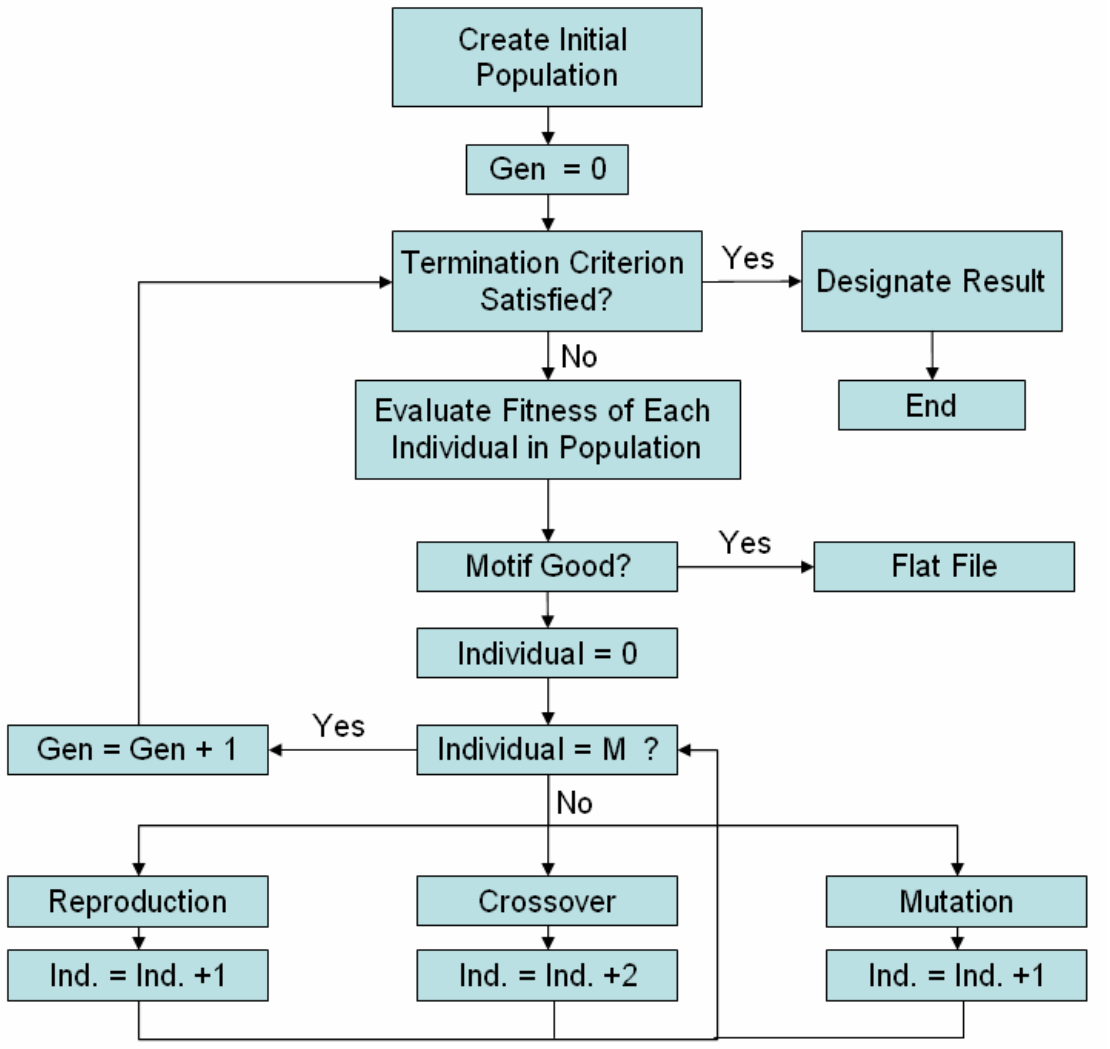


圖 3-7 基因規劃法流程圖

第一代，也就是流程圖中的初始代(Initial Population)，由實驗資料章節中所描述的 FMS 及 RMS 組成，族群中的每一個個體(Individual)即代表著一個相似結

構元，將每一個個體以適應函數(Fitness Function)計算其適應分數(Fitness)，也就是每個相似結構元的品質，這些分數作為選擇下一代族群的基準，從上一代到下一代的過程中可能經過交換(Crossover)、突變(Mutation)、或是單純的複製(Reproduction)等步驟。重複上述的演化過程直到我們訂定的演化終止條件。由於我們希望找到的是一群不錯的相似結構元，所以在演化的過程中若出現夠好的相似結構元，意即符合預設條件(何謂「夠好」的 Motif，將在下面章節詳細討論)，即把這相似結構元輸出儲存。

下面的小章節將根據基因規劃法系統中的小細節作更進一步的詳細介紹。

3.3.2.1 基因規劃系統中之參數

基因規劃系統中的參數及其名稱，和其他演化式計算演化法是相同的。以下是簡單的整理及介紹：

1. Generation：意指繁衍的代數
2. Population：每一代所包含的全部解的統稱
3. Individual：一個解即為一個 individual (個體)
4. Selection：選取某個 individual 的方式
5. Fitness：individual 的適應性分數
6. Fitness Function：算出 Fitness 的函式
7. Crossover：兩個 individual 交配產生新的 individual
8. Mutation：一個 individual 上的某處突然改變
9. Reproduction：完全複製選定的 individual
10. Population Size：參數，每一代所包含解的個數
11. Crossover Rate：參數，每一代 individual 以 Crossover 方式產生下一代的機率

12. Mutation Rate：參數，每一代 individual 以 Mutation 方式產生下一代的機率

13. Reproduction Rate：參數，每一代 individual 直接複製產生下一代的機率

3.3.2.2 適應函數(Fitness Function)

適應函數是主宰任何演化式計算成功與否的最重要關鍵因素，不同的適應函數會引導世代至不同的演化方向，最後也影響演化的結果。好的適應函數不但能把世代在廣大的搜尋空間中立即引導到正確的演化方向，好的適應函數更能減少不必要的搜尋，提升搜尋效率及品質。

過去的研究中，並沒有採取演化式計算相關的演算法尋找相似結構元，所以並沒有一套標準公式可作為依據。本研究的目的是尋找同屬一個分類的蛋白質中具代表性的相似結構元，且能越長越好，由此可知我們希望得到的相似結構元會是具有普遍性(prevalence)、立體結構相似(structure similarity)、夠長(motif length)、能處理蛋白質胺基酸突變的特性的。因此我們就根據這些條件來訂定我們最重要核心的適應性函數：

$$fitness(x) = w_a f_a(x) + w_o f_o(x) + w_p f_p(x) + w_l f_l(x) \qquad (1)$$

其中：x 表示第 x 個個體，f_a 為 Torsion Angle P-value Score；也就是立體結構相似程度的分數；f_o 為 Occurrence Score，相似結構元在實驗資料中的蛋白質結構中發生次數的分數(Occurrence)，也可看作此相似結構元包含了多少個區域結構(Fragments)；f_p 為有多少個蛋白質結構包含此相似結構元；最後 f_l 為相似結構元長度的分數；w_a 、w_o 、w_p 、w_l 則為各個分數的權重，以下會一一詳細說明。

Torsion Angle P-value Score，目的在於評斷一個相似結構元所包含的各個區域結構(Fragment)是否彼此結構相似。本研究使用扭轉角度相似程度為基準而不是傳統的 RMSD，理由除了前面第二章許多研究指出扭轉角度是影響蛋白質結構的因素之一外，RMSD 和 Torsion Angle P-value 所需要的計算量相差是非常大的。在相同的系統環境設定下，執行一世代所需要的計算時間，RMSD 花費 1.5 小時，Torsion Angle P-value 只要 4 分鐘，兩者相差了 22 倍之多。因此，在執行時間及最後結果兩個考量因素下，選擇以 Torsion Angle P-value 為衡量結構是否相似的基準。

Torsion Angle P-value 的計算方式非常簡單。系統執行前先將實驗資料中所有的蛋白質結構，胺基酸序列逐一掃描後，把相同序列、相等長度的子結構集合在一起並算其扭轉角度的統計值，包括了 phi-angle 平均值、phi-angle 變異數、psi-angle 平均值、psi-angle 變異數四個統計值。當我們建立了不同長度的背景扭轉角度統計資料後，便能輕易算出系統開始執行後所找到的相似結構元，其結構相似度在背景統計值的表現。

f_o 和 f_p 兩個分數分別探討，在此群蛋白質結構中，共發現了幾次(f_o)及發生在多少蛋白質上(f_p)。這兩種分數的目的在於引導系統能找到較普遍(prevalence)的相似結構元，包含的區域結構及蛋白質越多分數就越高。如果去掉這兩種分數，單純以結構相似度來引導系統學習方向，可以預見的是系統將很容易被一些特定且結構極相似的相似結構元拉著走。若是只是單純考慮 f_o 分數，有可能會高估大多數的區域結構只發生在少數蛋白質結構上的相似結構元的適應值，此種相似結構元其實並不是那麼符合我們尋找較普遍相似結構元的目標。因此，同時加上 f_o 和 f_p 這兩分數後，便能符合本研究希望找到具 Fold 代表性的

相似結構元的要求。

f_l 為相似結構元長度的分數，本研究想要找到較長的相似結構元，因此相似結構元的長度越長， f_l 分數就越高。

最後的適應分數為四個子分數的權重總和，既然是相加在一起，一定要把四個分數都限定在某個範圍之內，才不會因為分數衡量刻度的不同，造成最後適應分數的不公平，這個動作稱之為正規化(Normalize)。我們將四個子分數範圍限定在 0~1 之間，0 是最差，1 是最好。f_a 是 *P-value* 分數，*P-value* 的性質是越小越好，因此我們透過 $1-P\text{-}value$ 的轉換過程，使得 f_a 分數符合我們正規化的要求； f_o 則是此相似結構元所包含的區域結構數目除上當代全部有效相似結構元中子結構的最大數； f_p 把相似結構元所包含的蛋白質數目除上全部蛋白質的數目； f_l 和 f_o 算法一樣，把相似結構元長度除以當代全部有效相似結構元中最長的長度即可。所謂有效的相似結構元須符合 : 1.確實有找到 Fragments 2.Torsion Angle 分數 ＞0.5。方程式(2)至(5)為四個子分數的公式，圖 3-8 為各個子分數算法的例子：

$$f_a = \frac{1}{2}\left(P_{\Phi}\left(X \le \frac{V_{\Phi} - \overline{V}_{\Phi}}{\sigma_{\Phi}} \right) + P_{\Phi}\left(X \le \frac{V_{\Psi} - \overline{V}_{\Psi}}{\sigma_{\Psi}} \right) \right) \qquad (2)$$

V_{Φ} ：motif phi variance、

$\overline{V}_{\Phi}$ ：average of motif phi variance for all motifs with same length、

σ_{Φ} ：variance of motif phi variance for all motifs with same length、

V_{ψ} ：motif psi variance、

$\overline{V}_{\Psi}$ ：average of motif psi variance for all motifs with same length、

σ_{Ψ} ：variance of motif psi variance for all motifs with same length

$$f_o = \frac{\#.\ of\ \ Fragments}{Max(\#.\ of\ \ Fragments\ of\ \ all\ \ Motifs)} \tag{3}$$

$$f_p = \frac{\#\ of\ \ proteins\ that\ \ motif\ \ occurs}{total\ \#\ of\ \ proteins} \tag{4}$$

$$f_l = \frac{motif\ \ length}{Max(all\ \ motifs\ \ length)} \tag{5}$$

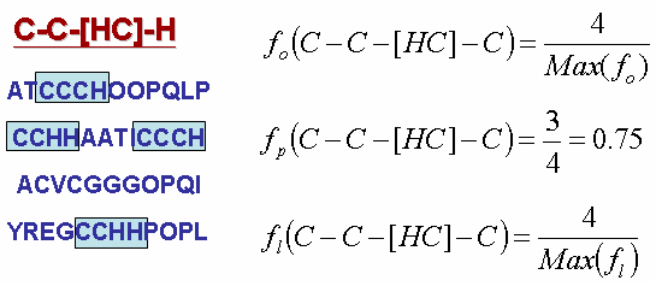


表 3-8 適應性函數算法實例

正規化的步驟，能有效的避免整體適應分數，因為子分數刻度不同值也不同而失去公平性及準確性。但，根據(1)的公式，仍潛藏著適應分數被錯估的危險。公式(1)是單純的把四個分數乘上權重再總和，如果其中一個子分數表現非常好，其他三個卻表現很差，最後的適應分數仍有可能被高估或錯估。為了防範此不公平性的發生，也為了符合本研究的目標，我們使用 F 分數(F-score) 將公式(1)修改為：

$$fitness(x) = \frac{1}{\frac{1}{w_a + w_o + w_p + w_l}\left(\frac{1}{w_a f_a(x)} + \frac{1}{w_o f_o(x)} + \frac{1}{w_p f_p(x)} + \frac{1}{w_l f_l(x)}\right)} \tag{6}$$

經過正規化及 F-score 的轉換後，便能正確的評估每個相似結構元的分數，引導系統往正確的方向學習，並符合本研究的目標。

3.3.2.3 挑選母代機制(Selection)

挑選母代在演化過程中扮演非常重要的角色，被挑選到的母代會進行複製(Reproduction)、交換(Crossover)、突變(Mutation)等不同運算子的操作而產生子代，因此挑選母代機制的好壞，會影響子代的表現，進而影響整個系統的搜尋效能。好的挑選方式將會大大降低系統整體的搜尋時間，減少不必要的搜尋。

挑選母代機制的方式大可分成三種，隨機選取(Random Selection)、輪盤選取(Roulette Wheel Selection)、及競賽選取(Tournament Selection)。隨機挑選，顧名思義即由系統隨機挑選，沒有考慮個體(individual)因素。輪盤選取法是考慮個體的適應分數，分數越好的被挑選到的機率越高。而競賽選取則為包含隨機及輪盤選取兩種方法的特性，先隨機挑出 N 個個體，再比較彼此的適應分數。三種方法各有其優缺點，隨機選取較少人採用，因選取時沒有考量個體的優劣，即喪失了適應分數的功效，也不符合「適者生存」定理。輪盤選取則是完全沒有隨機概念，大者恆大，很容易落入區域性最佳解(local optimal)。因此，競爭選取法保留了隨機機制，也符合「適者生存」定理，是目前最普遍採用的選取方法。本研究即採用競爭選取法，N 則設定為大部分研究最常設定的 7。

3.3.2.4 交換運算子(Crossover)

此運算子事交換兩個母代個體中一個節點(node)或是一顆子樹(subtree)。圖 3-9 為節點交換及子樹交換的示意圖：

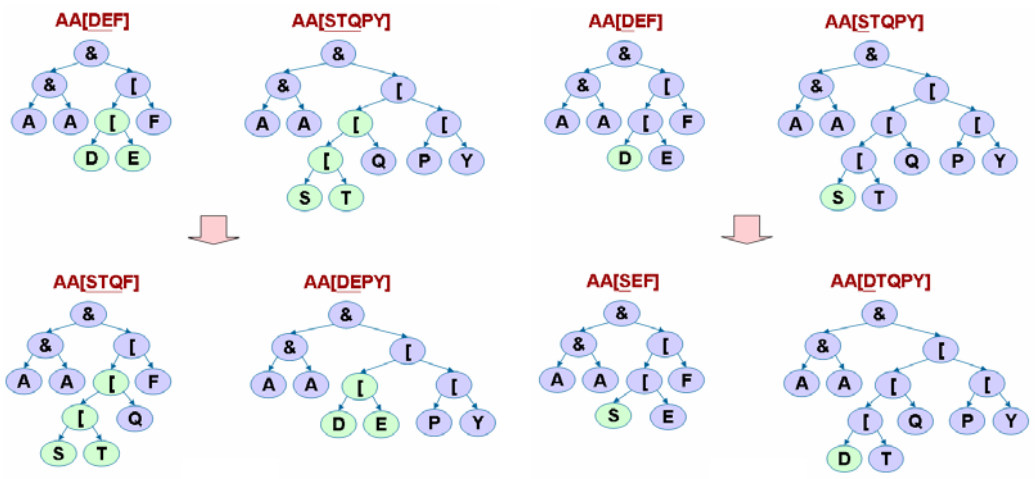


圖 3-9 Crossover 實例操作 (a)子樹交換 (b)節點交換

不論是哪一種交換，都可以達到產生新個體的功用。不過特別注意的是，本研究使用的是樹狀資料結構，兩棵樹交換的情形，相較於其他字串資料結構的交換會較複雜，也會有較多限制。不過本模型中中間節點和末端節點所記載的資訊是不同的，不管是節點之間、節點與子樹、或子樹之間的交換，都不會出現不合法的狀況，頂多是發生機率極小的無效交換。圖 3-10 為無效的交換實例：

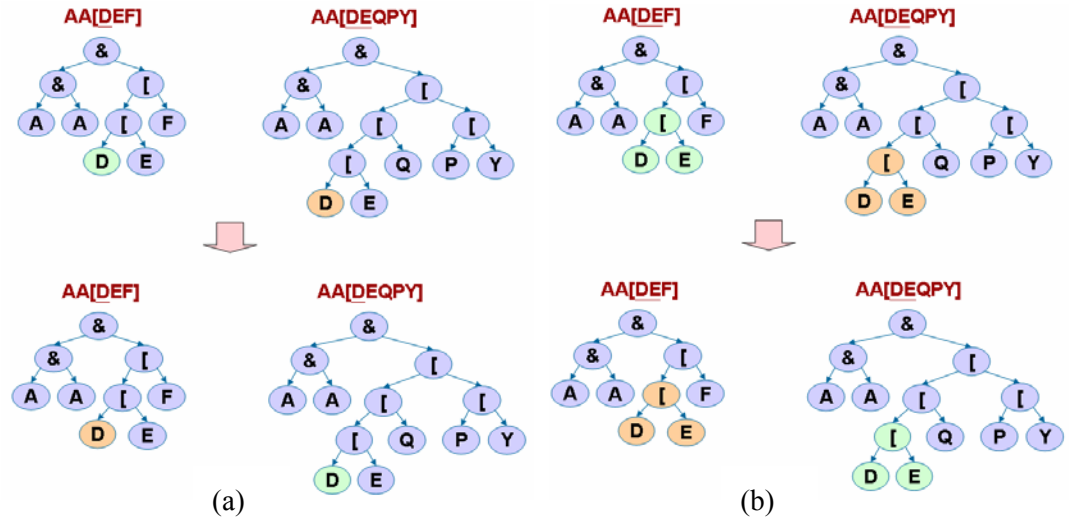


圖 3-10 無效的 Crossover (a) 無效的節點交換(b) 無效的子樹交換

3.3.2.5 突變運算子(Mutation)

突變運算子的設計除了傳統認知的可以幫助脫離區域性最佳解(local optimal)外，在本實驗還能微調已經找到的相似結構元，使其更加接近最佳解。例如：現階段找到的相似結構元為 [AT]-D-[KO]，包含的子結構有 ADK、ADO、TDK、TDO 四種，假設其中 ADK、ADO 事實上是雜訊，和其他兩個結構上並沒有非常相似，反倒是 IDK、IDO 這兩個子結構和 TDK、TDO 更為相似，那麼藉由突變運算子的演化下，便有機會演化出[IT]-D-[KO]這個真正的相似結構元。

突變運算子和交換運算子相同，在樹狀資料結構上會有較彈性的變化，和較多的限制，其方式可分為點突變(Point Mutation)及子樹突變(Tree Mutation)。圖 3-11 為突變運算子操作的實例及限制條件：由圖可知，並不是所有的突變動作都能造成有效的改變，某些例子下突變前後的相似結構元長相相同，不過這種情形發生的機率在本系統中非常小，浪費的計算資源相較於所需要的全部計算資源，可說是非常少，可以予以忽略。

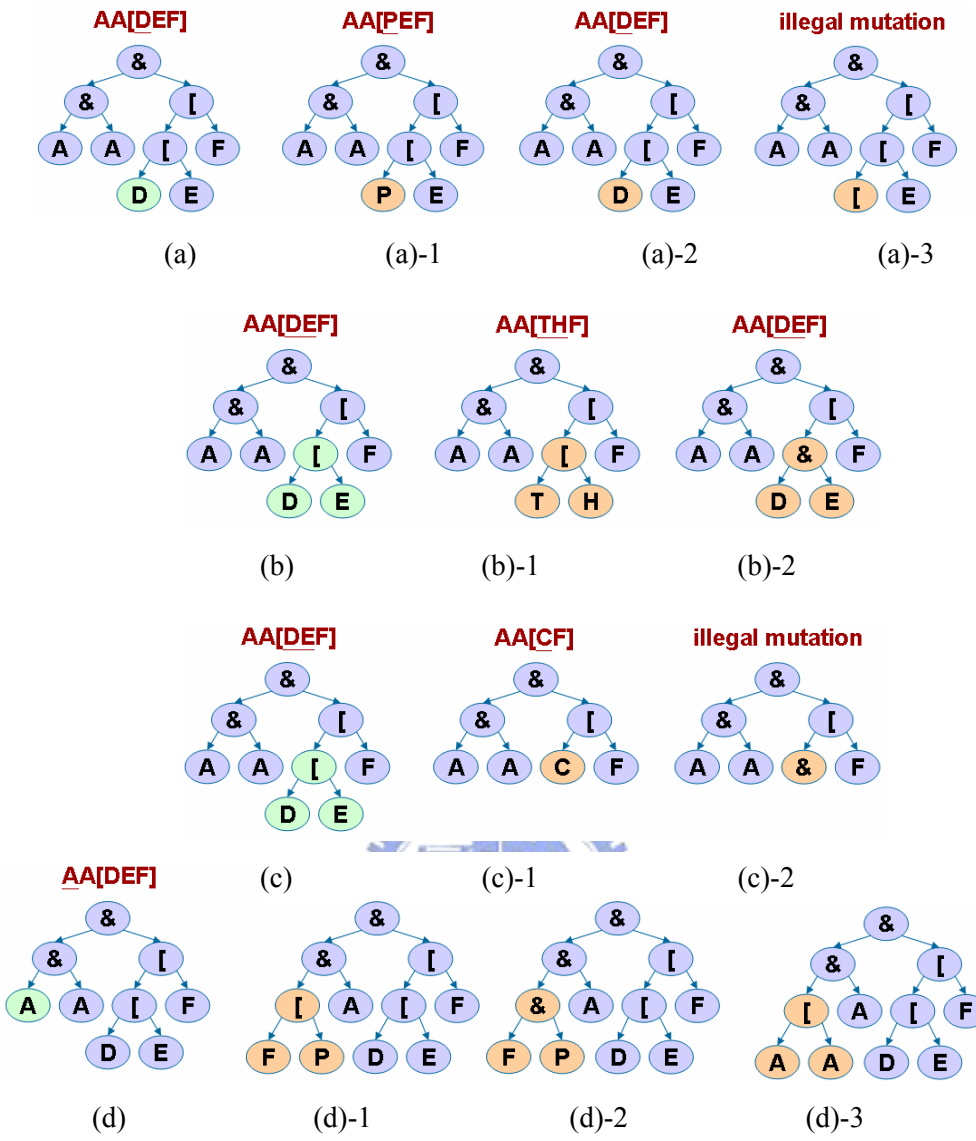


圖 3-11 (a)節點突變 (b) 子數突變 (c)子樹突變為節點 (d)節點突變為子樹

(a)-1 有效突變 (a)-2 無效突變 (a)-3 不合法突變 (b)-1 有效突變 (b)-2 無效突變

(c)-1 有效突變 (c)-2 不合法突變 (d)-1 有效突變 (d)-2 有效突變 (d)-3 無效突變

3.3.2.6 複製運算子 (Reproduction)

此運算子只是單純的將被選取的個體完整的複製到子代中而不做任何修改，隨著演化的進行，擁有較佳適應分數的個體可以藉此增加在族群中的數量獲得更大的競爭優勢。

3.3.2.7 清除重複個體 (Redundancy Removal)

族群在演化的過程中，適應分數高的個體會有較大的優勢繼續生存、擴大其族群，因此演化一段時間後，族群中的個體彼此之間變異性會下降，最後很可能導致族群中全都是相似個體，落入近親繁殖的惡況。因此，此運算子功能為清除相同或類似的個體，其目的是保持演化過程中族群的多樣性(diversity)。不過如果太常清除，適應分數高的個體就無法在族群中建立其優勢地位，所以我們設定每繁衍 10 代執行一次。

3.3.3 後置作業

本研究系統模型的最後一部份，是將前面第二部份基因規劃法模型中找到的相似結構元做最後的整理，選出最具代表性的相似結構元，並建立成資料庫。

需要做相似結構元整理的理由是，屬於同一 Fold 的蛋白質結構中，會有相似結構元發生的區域(region)是特定的，基因規劃法執行的過程中藉由適應函數的引導，會自動地幫我們找到這些有相似結構元發生的區域，卻也會同時找到數個相似且能代表這個區域的候選相似結構元(Candidates)，因此我們需要更進一步的處理動作，找出每個區域最具代表性且唯一的相似結構元。圖 3-12 為 GP 執行後相似結構元散落的示意圖，每一小方塊即代表一個相似結構元，小方塊聚集之處就是相似結構元極為可能發生的區域。換句話說，每一相似結構元發生區域，會有許多候選相似結構元(candidates)，且候選者彼此之間很容易會有重疊或重複的。圖 3-12 下方即列出了後選相似結構元之間可能的情形。

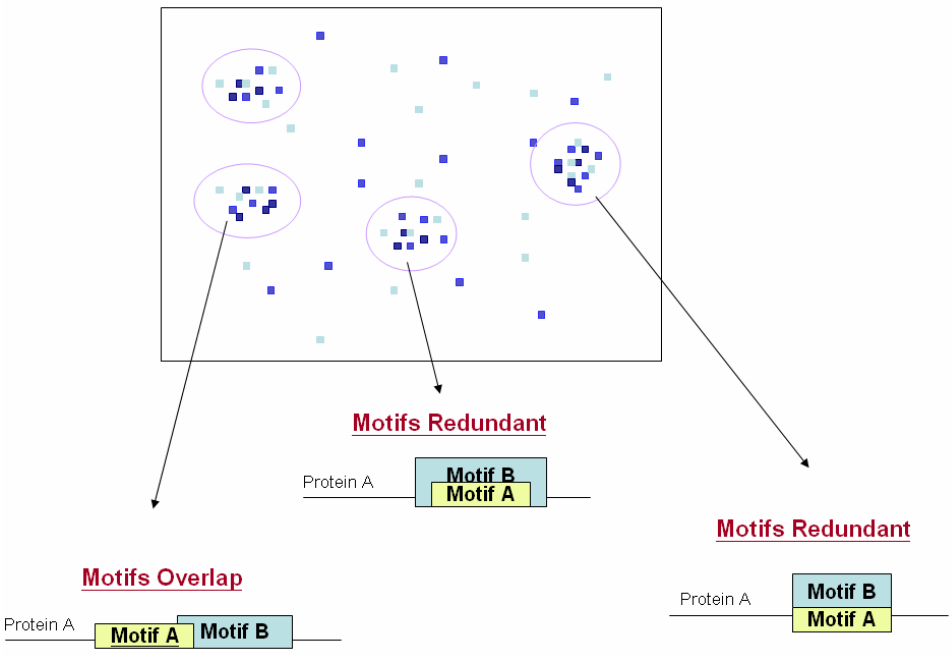


圖 3-12 基因規劃法執行完後的搜尋空間示意圖，小方塊為找到的相似結構元

為了解決上述 Motifs 彼此之間重疊或重複的情形，後置作業的部份採取分群(Clustering)的概念，把所有的相似結構元進行分群，再挑選出每一群代表性的相似結構元。圖 3-13 為後置作業的流程圖：

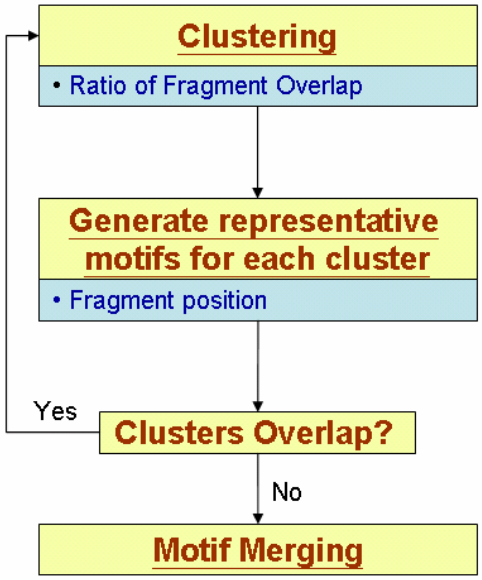


圖 3-13 後置作業流程圖

第一步驟分群(Clustering)，主要目的是將擁有相似結構的相似結構元找出來並劃分為同一群，也就是把先前基因規劃模型在搜尋空間劃分的邊界實際畫出來。分群的標準是依據相似結構元所屬區域結構的重疊比率來衡量，原因是當兩個相似結構元擁有部份相同的區域結構，便可以認定這兩相似結構元在結構上必定有某程度的相似，其餘不同 Fragments 的部份，則是導因於兩相似結構元在演化過程中的隨機機制。公式(3) 為 Fragments 的重疊比率(Fragment Overlap Ratio)算法：

$$\text{Fragments Overlap Ratio (FOR)} = \frac{\text{\# of Fragments Overlap}}{\text{Max \# of Fragments (MotifA, MotifB)}} \quad (3)$$

圖 3-14 為 Fragment Overlap Ratio 的實際例子：

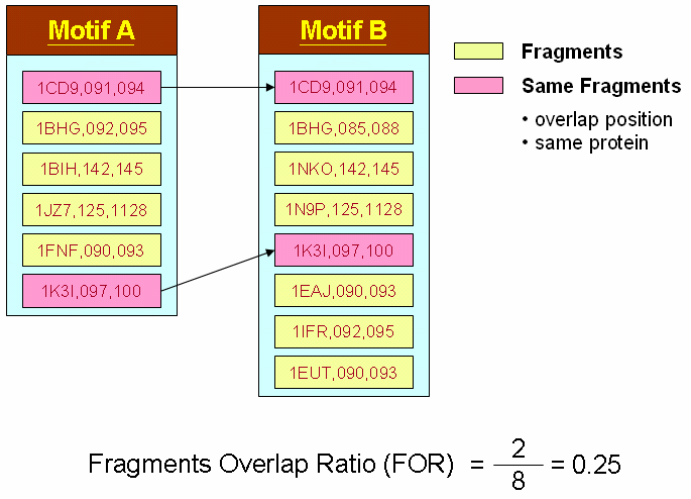


圖 3-14 子結構重複比率(FOR)計算實例

相同於其他傳統的分群演算法，分群的動作會一直重複不斷，直到分群數及分群內容不再變動為止。唯一不同的是，通常分群演算法會先等到分群數及分群內容收斂後才開始處理每一群的資料，而本研究中則是每次分群後即刻處理分群內容，選出當次每一群中具代表性的相似結構元。流程需要如此改變的原因是若不先處理每次分群群內的數個相似結構元，接下來的分群結果將會和上一次一模一樣，為了確實達到分群的效果，每次分群後每一群先產生出代表性的相似結構元，下次的分群便可根據這些新的相似結構元繼續進行直到收斂為止。

選取每一分群中代表性相似結構元的依據是區域結構在蛋白質序列發生的位置。本研究是以Fold層級的觀點去找尋代表性相似結構元，因此區域結構在序列上發生的位置就顯得很重要。當大部分區域結構發生的位置相近時，該相似結構元就可以視為所屬Fold在該序列位置的代表性相似結構元。因此，我們將被歸於同一群的所有區域結構，統計其所發生的序列位置，找出最密集的區域所包含的區域結構並產生一個新的相似結構元，若這些區域結構結構相似度達到成為相

似結構元的標準，即是該分群的代表性相似結構元(Representative Motif)；若沒有達到成為相似結構元的標準，便逐一淘汰最不像的區域結構，直到整體結構相似度達到成為相似結構元的標準為止。

經過上述嚴苛條件篩選後的相似結構元，雖然每個結構元本身所包含的子結構在胺基酸序列上發生的位置極為類似，但結構元之間卻仍然會有少許的重疊或是相連，因此在後置作業中最後一個步驟的目的便是把上述兩種狀況的結構元找出來並與以合併，相似結構元長度還能因此而變長，一舉數得。

經過反覆上述動作後找到的相似結構元，雖然無法證明是每個結構相似區域(structure conserved region)的最佳解，但經過嚴苛條件的層層篩選下，可以說是和真正的相似結構元已經非常相近，或是即為真正的相似結構元，也是本研究最後的結果。

3.4 模型環境設定

本研究模型總共有三個子模型，每個子模型因有不同的目的而有不同的參數設定。一開始準備實驗資料模型部份，我們選擇SCOP40中所列出胺基酸序列一致性低於40%的蛋白質，目的是避免若將一致性太高的許多蛋白質納入模型中，系統將很容易受到胺基酸序列相似且結構也相似的結構區段所牽引，導致最終找到的相似結構元不夠普遍化，包含的子結構的胺基酸序列會太類似，也就是系統搜尋相同結構但胺基酸序列不相似的子結構的能力降低。因此本實驗選擇了大部分研究最常使用的SCOP40為實驗輸入的資料部份。

由於第三階段篩選相似結構元的過程是以結構相似程度(ACC≥0.95)、及相似結構元包含之子結構在胺基酸序列上發生的位置為篩選條件，因此在第二階段基因規劃模型的目的，就是希望能找到的相似子結構越多越好，在胺基酸上發生的位置越廣越好，如此這般到了第三階段時才不會因為嚴苛的篩選條件而把大部分的子結構通通過濾掉。因此在基因規劃模型中，適應性函數(公式3)的四個權重設定為 $w_a = 2$ 、 $w_o = 4$ 、 $w_p = 4$ 、及 $w_l = 1$ ，權重較大的 w_a 、 w_p 可以導引系統去尋找更多且不同的子結構。

而基因規劃模型中四個系統運算子的機率設定為：

Mutation rate = 0.5

Crossover rate = 0.3

Reproduction rate = 0.2

Redundancy Removal則為每十代世代執行一次。

本實驗參數設定中，特別將突變運算子的機率調得較一般問題在使用基因歸

化法時所設定的職還要高很多。理由是大部分的問題是要找全域最佳解，而本實驗則是希望能找到發生在不同區域的「大概相似子結構」，也就是多個區域最佳解，因此高的突變運算子機率不但可以幫助不會陷在區域最佳解，更能在搜尋空間中任意跳動、找尋不同區域的區域最佳解，達到我們的需求。

基因規劃模型中，我們將族群大小(population size)設定為8000，也就是一代當中同時有8000個相似結構元在互相競爭，另外代數設定為100代，執行完100代就停止。

另外，此子模型把認定是否為合格相似結構元的標準設定為ACC≥0.95且相似結構元的長度必須大於或等於4，唯有被認可且輸出的相似結構元，才能進入下一階段進行最後處理。

在最後一階段的後置作業模型中，誠如前面所提，所有步驟皆是以ACC≥0.95來認定相似結構元的合法性，保持整個系統的一致性。另外FOR則設定為0.3，因當FOR≤0.3時所作的分群(Clustering)，最後分群結束的群數都會收斂在某一定值，也就是當FOR≤0.3時，每一群之間不再有重疊，群之間的界線很清楚。

第四章　實驗結果

本研究選定SCOP分類中的a.39為實驗對象，並以SCOP40為挑選蛋白質結構標準。Fold a.39是一非常好的實驗對象，除了Fold的大小適中外，最主要是此分類中的蛋白質具有一著名的蛋白質作用區域(Function Site) – EF-hand，其他眾多開發方法的研究中也紛紛挑選此Fold為實驗對象。本實驗以SCOP40為挑選蛋白質結構的標準，一共包含了48個蛋白質結構作為本研究模型輸入的實驗資料。

本研究在基因規劃模型中是以扭轉角度來衡量相似結構元的結構相似程度，文獻探討章節中有許多研究都指出扭轉角度是影響結構形狀的因素之一，但若只是單用扭轉角度的變異值作為衡量標準，是否真的能找到RMSD也小的相似結構元呢？為了解答這個疑惑，我們將Fold a.39中48個蛋白質的胺基酸序列中取出不同長度的子結構，觀察其扭轉角度的變異值和RMSD之間的關係。若兩者成正比關係，即可證明單獨使用扭轉角度為衡量標準是可行的。圖4-1為Fold a.39不同長度的結構片段，其扭轉角和RMSD之間的關係。由圖中的曲線可明顯觀察到，當結構片段長度越長時，因為完全一樣的胺基酸序列越少，相對的扭轉角的變異值就越小。

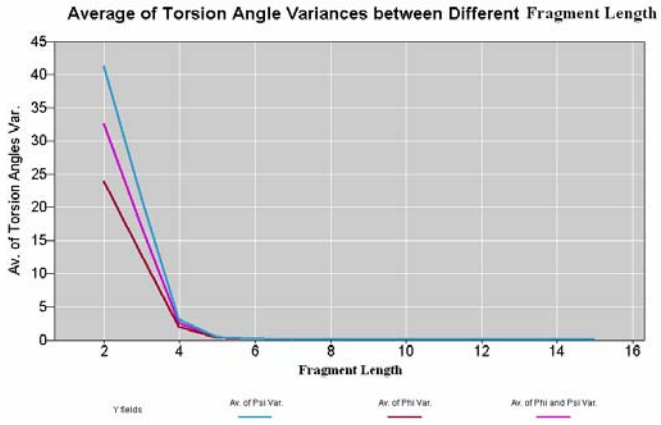


圖4-1 不同長度的結構相似元和扭轉角之間的關係

在第三章方法論中介紹了三個子模型，第一個模型是實驗資料的準備，包括了FMS及RMS。前面已經有說明FMS和RMS來源、特性、及功能，FMS為實際存在於胺基酸序列中長度為3的區域結構，目的是因提供了可能的相似結構元的區域結構而提升系統搜尋速度，圖4-2為使用及不使用FMS對基因規劃法搜尋速度的影響。從圖中可以輕易分辨出兩者的差異，沒有使用FMS的模型在前期必須花較多的時間去搜尋結構可能相似的區域結構，而使用FMS的模型則因為區域結構由系統提供，演化速度明顯較快，找到的相似結構元品質都較不使用FMS來得好。

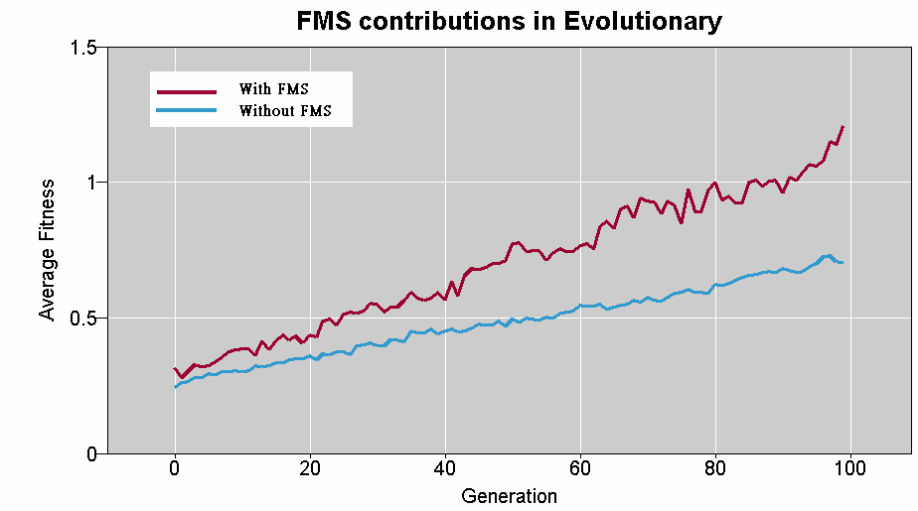


圖4-2 FMS對於基因規劃模型的貢獻

另外，RMS的功能為提供相似結構元可能的胺基酸突變組合，最後找出來的相似結構元才能較具變異性(diversity)。圖4-3為使用及不使用RMS兩種狀況下，變異性的相似結構元佔當代找到所有相似結構元的比例。從圖中可以明顯看出使用FMS和RMS為輸入資料的模型，再系統一開始執行後，擁有變異性相似結構元的比例就一直保持在一定水準，而且有逐漸向上攀升的趨勢；相較之下單只使用FMS直到演化至第九代時才開始出現變異性相似結構元，數目佔整個個體

的比例也不大。當然，在無限時間下兩個模型都能找到差不多的相似結構元，但加了RMS的模型尋找到最佳解的速度必定比單只用FMS來得快速。

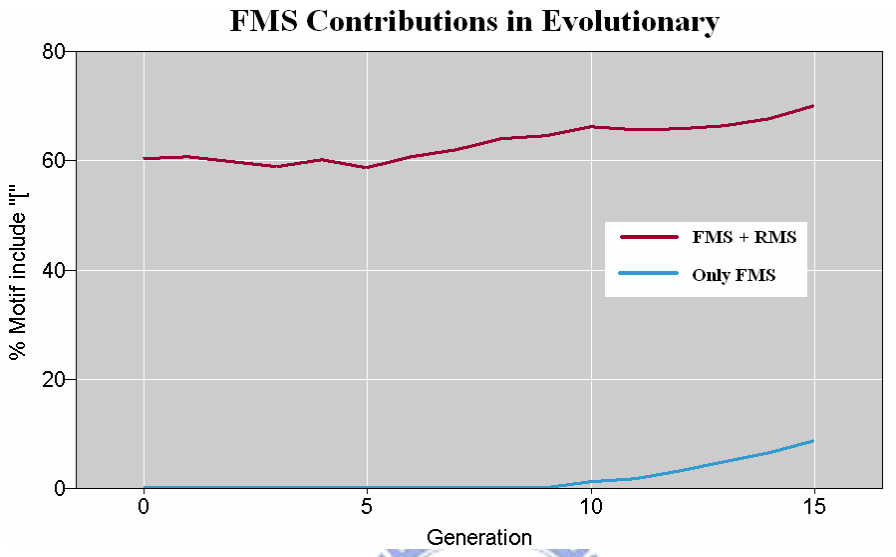


圖4-3 RMS對於系統的貢獻

確定好了實驗資料，圖4-4為基因規劃法執行時的表現。藍色線為當代最佳的適應分數，紅色則為平均適應分數。我們可以發現大約在六十多代時系統已經找到非常好的解，而群體則在大約一百代之後收斂，此時我們也找到夠多的候選相似結構元提供方法模型中的第三部份：後置作業。

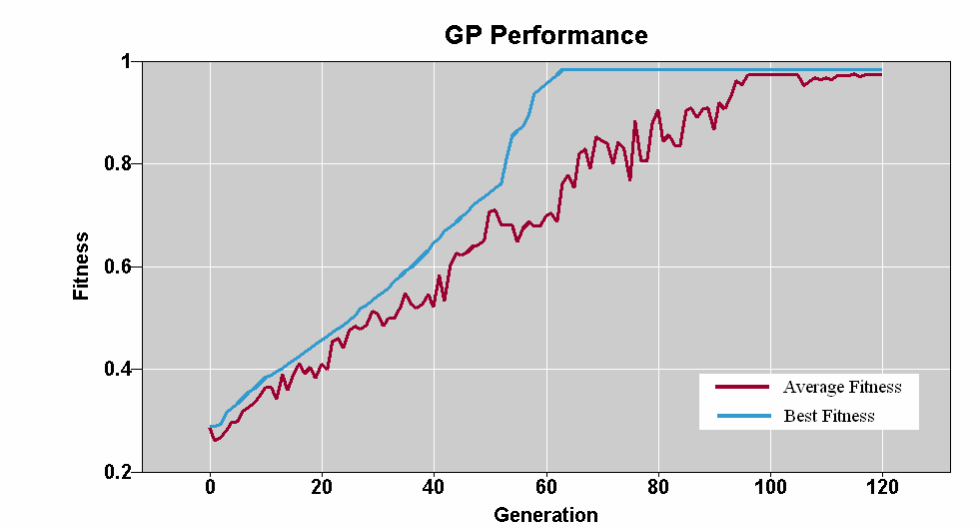


圖4-4 基因規劃模型的表現

表4-1 為本研究找到的相似結構元整理表，「Motif」欄位是相似結構元的長相，「RMSD」則是將該相似結構元所包含的區域結構兩兩算出其RMSD，再予以平均得到的值；「ACC」算法同RMSD，也是兩兩ACC的平均值。「RRMSD」則將RMSD轉換為length indenpendnce的值。「No. of Proteins」則是該相似結構元發生在多少蛋白質上；「No. of Fragments」是該相似結構元包含的區域結構數目；「Fold Specificity」為該相似結構元只存在於Fold a.39的比例，值是由該相似結構元掃描PDB中所有蛋白質的FASTA格式的胺基酸序列，統計該相似結構元在各個不同Fold出現的比例；「Start Position」是該相似結構元包含的區域結構在胺基酸序列上發生位置的範圍；「Length」為該相似結構元的長度；「Annotation」則是對於該相似結構元的一些描述，主要是該相似結構元具有的特徵和PROSITE及Foldzilla兩資料庫的比較。

No	Motif	RMSD	ACC	RRMSD	No. of Proteins	No. of Fragments	Fold Specificity	Start Position	Length	Annotation
DEC01	[EFNLT]-[EKSNT]-[DEIFL]-[AEIGK]-[DFGKNT]-[AITV]-[FLMY]-[DEFHGN]-[DEKQMS]-[AIKY]-[ADIKS]	1.387	0.962	0.23	8	8	80.9%	2-13	11	Fold-Signature
DEC02	[DK]-[EFV]-[FKQT]-[FLRST]-[DQRV]-[FLMY]	0.799	0.966	0.28	5	5	2.4%	10-27	6	EF-hand calcium-binding domain
DEC03	[AEGLSPNMT]-[ADEKLQS]-[ADEGLRNTQY]-[EILV]-[FHTLV]-[AFKLSMQY]-[DEIKLRQTMV]-[AEFGKMLV]-[EFIKVMSLY]-[ADKMNRQS]-[ADEFGSNKHT]-[ADEFGTSPINV]-[CDGFKLVNTS]	1.804	0.954	0.65	13	14	17.9%	24-52	13	Fold-Signature
DEC04	[DGLNRT]-[EKPT]-[ADGKSNV]-[EGVTY]-[ALVW]-[ADQR]-[DERSV]-[ILMVW]-[FIKM]-[EGKNS]-[EKMR]-[AKLFV]-D-[AILKR]-[DEN]-[EGKNST]-[DKN]	2.506	0.959	0.30	5	5	82.6%	47-61	29	Fold-Signature EF-hand calcium-binding domain
DEC05	[ADFLINTV]-[DELNQR]-[DEFKL]-[FKMTLQY]-[EKLMTV]-[AEFKPRVY]-[ALMDQV]-[DIFKLNQT]-[AFLNQSY]-[EIKLQV]-[ADEFGHMV]-[ADEGILMV]-[KLRNTV]-[ACDEGIHST]-[ACEGILST]	1.98	0.952	0.68	10	10	50.9%	68-84	15	Fold-Signature S,Casein kinase II phosphorylation site G is the N - myristoylation site],N-myristoylation site
DEC06	[EFQKT]-[DEGHNP]-[DEPSRV]-[FILTSV]-[AKSPT]-[DFLS]-[FHIL]-[FKMY]-[AKQR]-[ABFHGK]	0.919	0.983	0.14	6	7	100%	93-107	10	Fold-Signature S,Casein kinase II phosphorylation site

表 4-1 本研究所找到的相似結構元

No	Motif	RMSD	ACC	RRMSD	No. of Proteins	No. of Fragments	Fold Specificity	Start Position	Length	Annotation
DEC07	[DT]-[FKNR]-[DE]-[DHNR]-[DNST]-[GN]-[CIKMT]-[ILMS]-[DGNR]-[DFKR]-[ENQ]-[ADEY]-[FGL]-[IKLS]	1.908	0.955	0.34	5	5	100%	102-108	14	Fold-Signature EF-hand calcium-binding domain
DEC08	[DFIK]-[AQS]-[EIKR]-[ELNV]-[ADCE]-[ESV]-L-[ELY]-[IKL]-[DGVW]-[ALTV]	1.375	0.963	0.27	4	4	59.1%	117-125	11	Fold-Signatur
DEC09	[IKNPV]-[KLRT]-[FPQR]-[EIMN]-[EIKLV]-[DNRTV]-[ACKV]-[ELT]-[KLSRT]-[AILQT]	1.247	0.961	0.35	5	5	35.1%	140-147	10	Fold-Signature
DEC10	[AGR]-[DST]-[GST]-[FR]-[IKR]-[IR]-[TV]-[KL]	1.892	0.9560	0.32	3	3	84.6%	176-190	13	Fold-Signature
DEC11	D-[SV]-D-[RT]-[ST]-G-[KT]-[IL]-G-[FS]-[ES]-E-[FL]	0.369	0.982	0.04	1	2	35%	150-180	8	Fold-Signature EF-hand calcium-binding domain
DEC12	[FPQ]-[RV]-[QR]-L-G	0.213	0.998	0.063	3	3	2%	231-246	5	None

表 4-1 本研究所找到的相似結構元

詳細的相似結構元資訊，可以到 http://aneta.no-ip.com/drawmotifs.php 查看，我們將於後面討論的章節詳細探討找到的相似結構元之性質。

第五章　結論與討論

5.1 結論

利用蛋白質的胺基酸序列來探討蛋白質結構相關問題，一直是蛋白質結構領域中非常熱門的話題，生物學家也引領盼望著有好的方法及結果能拿來實際運用。有鑑於蛋白質本身即為非常複雜難解，近年來不少傑出科學家紛紛投入研究，但至今遲遲未有重大突破的研究方法出現。

本研究所提出的模型運用了生物資訊界還較少研究學者使用的基因規劃法。基因規劃法擁有非常好的解困難問題的能力，非常適合運用在蛋白質立體結構這種問題本身即為複雜的領域。基因規劃法傑出的解問題能力，再搭配上分群法的概念，構成了本研究模型的主要枝幹。雖然本模型無法徹底解決蛋白質結構問題中現存的所有問題，但對於過去已提出方法之瓶頸及困難點有著非常好的解決觀點，且模型執行完的結果也深具生物意義。

5.2 討論

本研究的結果和其他三個性質相似研究的結果相互比較，表 5-1 為四篇研究找到的相似結構元比較表，另外各個資料庫相似結構元的詳細資料分別於附錄中詳細紀錄。

DataBase	No. Of Motif	Avg. No. of Proteins	Avg. No. of Fragments	Avg. RMSD	Avg. ACC	Avg. Length (a.a)
PROSITE	17	12	23.11	4.06	0.7265	9.53
Foldzilla	4	10	13.25	2.24	0.943	18
Wangikar	1	36	63	N/A	N/A	8
PRODEC	12	5.6	6.1	1.37	0.965	10.36

表 5-1 四個相關資料庫的結果比較

資料庫 PROSITE 列出的相似結構元，是將 Fold a.39 中，屬於 SCOP40 的 48 條蛋白質序列的 FASTA 檔案格式，分別輸入至 PROSITE 網頁 (http://us.expasy.org/prosite/)上所提供的「Scan PROSITE」工具，PROSITE 會依據最新版資料庫資料，將每條胺基酸序列有的相似結構元回報給使用者。透過此便利的工具，將所得到的結果稍加整理，便能得到適合我們實驗資料的相似結構元。Foldzilla 及 Wangikar 兩資料庫的相似結構元則是直接由其網頁或論文上擷取。其中，RPOSITE、Foldzilla、及本研究的實驗資料為 SCOP40 所屬的蛋白質，而 Wangikar 則為 PDB95，包含了將近 256 個(全部 a.39 的蛋白質數量)蛋白質結構。

由表 5- 我們可以觀察到 PROSITE 的相似結構元屬於較短、較普遍、但結構卻最不相似；而本研究 PRODEC 找到的相似結構元特性則是結構相似度最高、長度也長、但普遍性上卻是最差的；Foldzilla 則夾在 PROSITE 和 PRODEC 之中；Wangikar 只找到一個相似結構元，數量太少而無法觀察其整體特性。

除了上述的巨觀比較外，相似結構元還可就蛋白質功能區域(Function Site)及蛋白質分類專一性(Fold-Specifivity)兩方面來討論。

蛋白質功能區域

在相似結構元的內容方面，PROSITE 全部的相似結構元都有其生物上的意義，17 個相似結構元中有 10 個是蛋白質功能區域，Foldzilla 則是 4 個中有 2 個包含了蛋白質功能區域，而本研究每一個相似結構元中都有部分或全部的子結構，和 PROSITE 或 Foldzilla 相似結構元的子結構重疊，部分相似結構元甚至是 PROSITE 或 Foldzilla 的子相似結構元(sub-Motif)，子相似結構元的意思是指該相似結構元包含的所有子結構都出現於另一個子結構較多的相似結構元中。表 5-2 為四個資料庫關於 EF-hand 找到的相關相似結構元：

DataBase	Motif	No. of Proteins	No. of Fragments	RMSD	ACC	Length
PROSITE	PS00018	33	51	6.67	0.58	23
Foldzilla	MTF00052	20	33	4.37	0.87	29
	MTF00053	7	7	2.71	0.93	17
Wangikar	Fa.39.1.3.5.523	36	63	N/A	N/A	8
PRODEC	DEC02	5	5	0.80	0.97	6
	DEC04	5	5	2.51	0.96	28
	DEC07	5	5	1.91	0.96	14
	DEC11	1	2	0.37	0.98	8

表 5-2 四個資料庫找到 EF-hand 相關的相似結構元

PRODEC 找到的四個和 EF-hand 相關的相似結構元，圖 5-1 為編號 DEC02 其所包含的區域結構及 PROSITE、Foldzilla 兩者相關區域結構的示意圖。

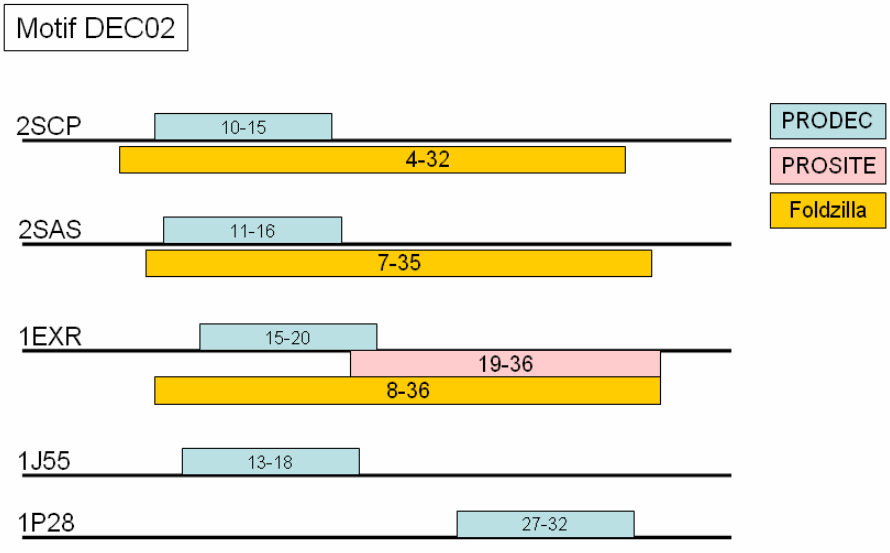


圖 5-1 DEC0 在胺基酸序列上與 PROSITE、Foldzilla 比較

不同顏色區塊代表來自不同資料庫的區域結構，區塊上的數字則為其在胺基酸序列所發生的位置。其中，只有 1EXR 為三個資料庫都有找到，有三個和 Foldzilla 相重疊，且重疊的部份傾向於 Foldzilla 的前半部。圖 5-2 為三個資料庫分別在蛋白質結構上所分布的區域。透過此結構的示意圖，我們可以發現 DEC02 找到的結構區段是在 Helix 的部分，相較於其他兩者其包含的 EF-hand 是不完整的。可能之一是 Loop 及接續的 Helix 在結構上變異性比較大的，另外一個原因為 DEC02 包含了其他並沒有 EF-hand 但結構相似的其他結構。

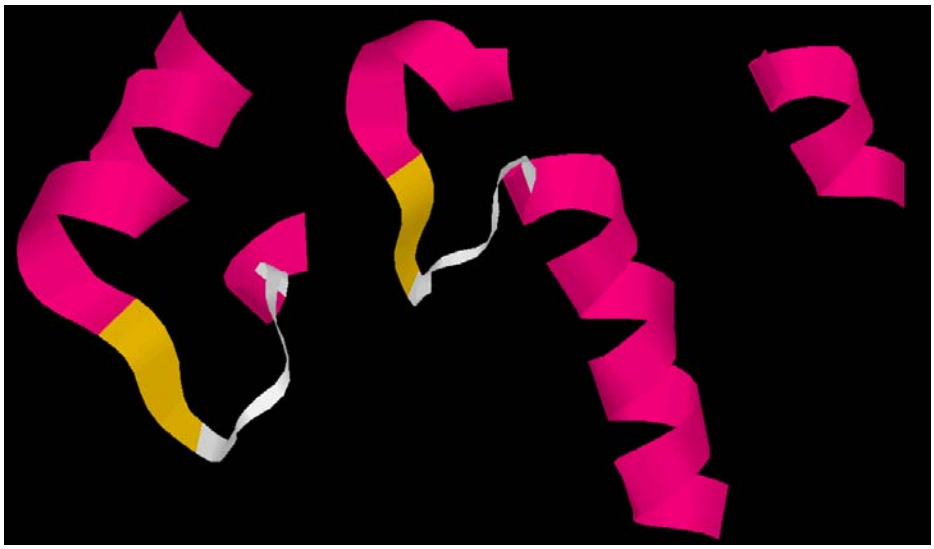


圖 5-2 PROSITE、Foldzilla、PRODEC 在蛋白質 1EXR 所找到的 EF-hand

左-PROSITE、中-Foldzilla、右-PRODEC

圖 5-3 及圖 5-4 則為 DEC04 在序列及結構上的示意圖。

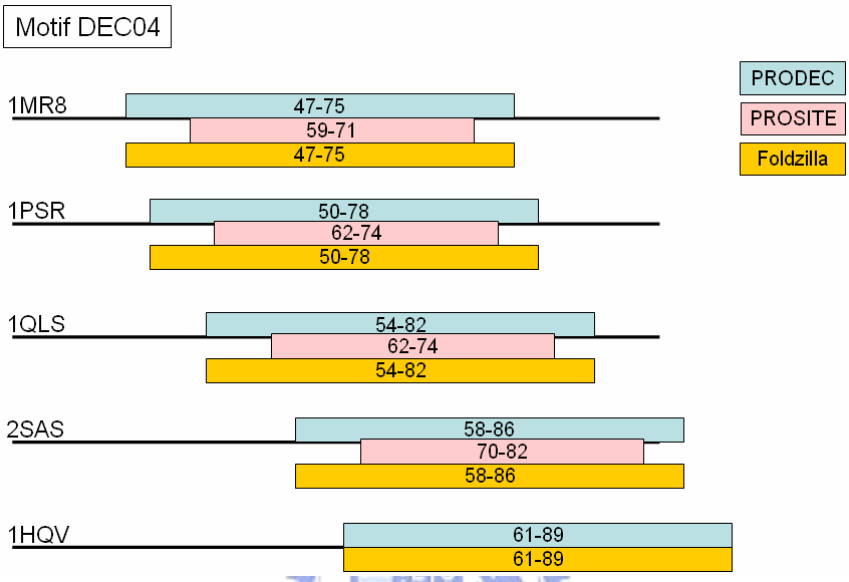


圖 5-3 DEC04 在胺基酸序列上與 PROSITE、Foldzilla 比較

圖 5-4 PROSITE、Foldzilla、PRODEC 在蛋白質 2SAS 所找到的 EF-hand

左-PROSITE、中-Foldzilla、右-PRODEC

DEC04 和 Foldzilla 是完全重疊，五個區域結構中有四個是 PROSITE 也有找到的。從結構圖我們可以發現 PROSITE 找到的 EF-hand 少了一個 Helix，而

Foldzilla 及 PRODEC 找到的是完整的 EF-hand。

圖 5-5 及圖 5-6 則為 DEC07 在序列及結構上的示意圖。

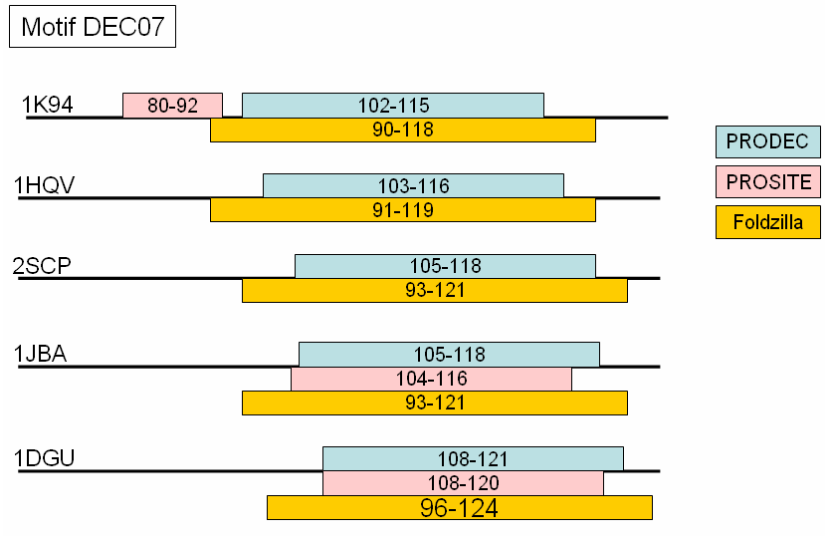


圖 5-5 DEC07 在胺基酸序列上與 PROSITE、Foldzilla 比較

圖 5-6 PROSITE、Foldzilla、PRODEC 在蛋白質 1DGU 所找到的 EF-hand

左-PROSITE、中-Foldzilla、右-PRODEC

DEC07 和 PROSITE 及 Foldzilla 的重疊部份也很多，尤其是和 Foldzilla，只不過找到的區域結構較短。從結構圖上來看，三者都有包含 EF-hand 的兩個 Helix 及中間的 Loop，不同的長度主要是導因於找到不同 Helix 的長度。

圖 5-7 及圖 5-8 則為 DEC011 在序列及結構上的示意圖。

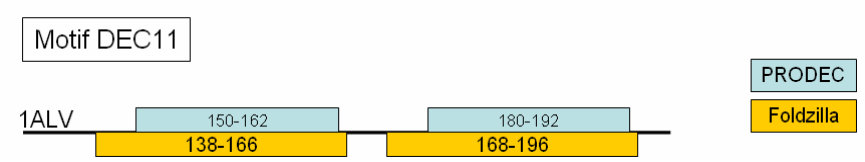


圖 5-7 DEC11 在胺基酸序列上與 PROSITE、Foldzilla 比較

圖 5-8 PROSITE、Foldzilla、PRODEC 在蛋白質 1ALV 所找到的 EF-hand

左-Foldzilla、右-PRODEC

DEC11 所包含的 EF-hand 在 PROSITE 中並沒有被標註，但卻有被 Foldzilla 標明，不過因為 PDB 檔案中有缺漏值，部份連續的胺基酸沒有空間座標可以視覺化，因此呈現的結構圖是不連續的。但，不論是從胺基酸序列或是結構上來看，DEC11 找到的區域結構被包含在 Foldzilla 裡面，因此長度也較短。

從以上關於 EF-hand 相關的四個相似結構元的深入探討，我們可以發現除了 DEC02 外，其餘三個結構相似元相較於其他兩者，雖然包含的區域較小，但

還是都能有效的將 EF-hand 中兩個 Helix 及一個 Loop 給囊括。至於 PROSITE 或是 Foldzilla 有找到的區域 PRODEC 卻沒辦法包含,原因是那些區塊的結構變異性是比較大的(ACC<0.95),所以不管是在基因規劃模型或是後置作業中,都無法被納入相似結構元中。

除了上述四個相似結構元外,其他一些相似結構元的部份區域結構也和 PROSITE 所標註的區域結構有所重疊,上一章紀錄相似結構元的表格有予以標註。由於這些被包含的蛋白質功能區域屬性是非常普遍且長度短的,因此就不再就蛋白質功能面討論。

蛋白質分類專一性域

相似結構元除了在功能方面外,本研究也計算了每個相似結構元在蛋白質分類上的專一性(Fold-Specificity)。在 12 個相似結構元中,有 7 個的分類專一性大於 50%,若把低於 50%但相較於其他 Fold 仍為最高的則有 10 個,分類專一性過低的通常都為較短的相似結構元,往往出現在其他為數眾多的蛋白質分類中,此種相似結構元在實用性及意義上就比較少,例如 DEC02 及 DEC12。

本研究的目標是找結構相似的相似結構元,結構元的篩選標準是以結構相似度為條件,但相似的結構並未必有相同的功能、具有相同功能的結構也未必一定很相似,因此以結構相似的前提下所找到的子結構中,就容易發生只有部分子結構擁有蛋白質功能區域,或是同一個結構卻擁有不同的功能,這情形也同時發生在 PROSITE 及 Foldzilla 的相似結構元中。

本研究結果雖然在結構相似度上優於其他的資料庫,但所包含的蛋白質或

是子結構卻是四者最低的，也就相似結構元在 Fold 中的普遍性相較之下是最低的。主要的原因是模型中對於蛋白質結構相似度的門檻值設定較嚴苛(ACC >= 0.95)，所包含的子結構數目自然就會較少、普遍性較低。門檻值的設定是很自由的，端看使用者的需求。若是使用者希望能找到普遍性較高且結構不需要太相似的相似結構元，只需把門檻值降低到適當值便能找到想要的相似結構元。

另外一個造成普遍性過低的原因是在系統的後置作業子模型中，依據胺基酸上發生的位置來篩選過濾，因此擁有相似結構但發生位置差太多的區域結構就非常容易被過濾掉，造成最後相似結構元只包含了較少的子結構。當然，不同的篩選機制會產生不同的相似結構元，但不管使用者的目標和所設定的篩選條件為何，在候選相似結構元眾多的子結構支持下，都會有不錯的結果。

雖然在後置作業模型中是以區域結構在胺基酸上發生的位置來篩選，理當每個相似結構元都會有其各自的區域，不過由結果我們可以發現每個相似結構元發生的位置不是個定值，而是一個範圍。這符合了蛋白質演化過程中，結構相似的區域會因為胺基酸的插入(insertion)和消失(deletion)所造成發生位置的位移。另外，細看每個相似結構元發生位置的區段，可以發現到其實是有部分重疊的，但若再加上蛋白質資訊，重疊的部份都是發生在不同的蛋白質上，也就是沒有任何區域結構是重疊，這又再次說明了蛋白質演化過程中的自然現象，也說明了本研究之第三階段的分群過程是可信任的，

5.2.1 資訊科學上的貢獻

本研究模型中的基因規劃及後置作業兩個子模型所包含的概念及作法，以資訊科學角度來看，和傳統的分群方法想解決的問題、解決的方法極為相同。圖 5-9 為傳統分群方法和本研究模型方法的比較流程圖。

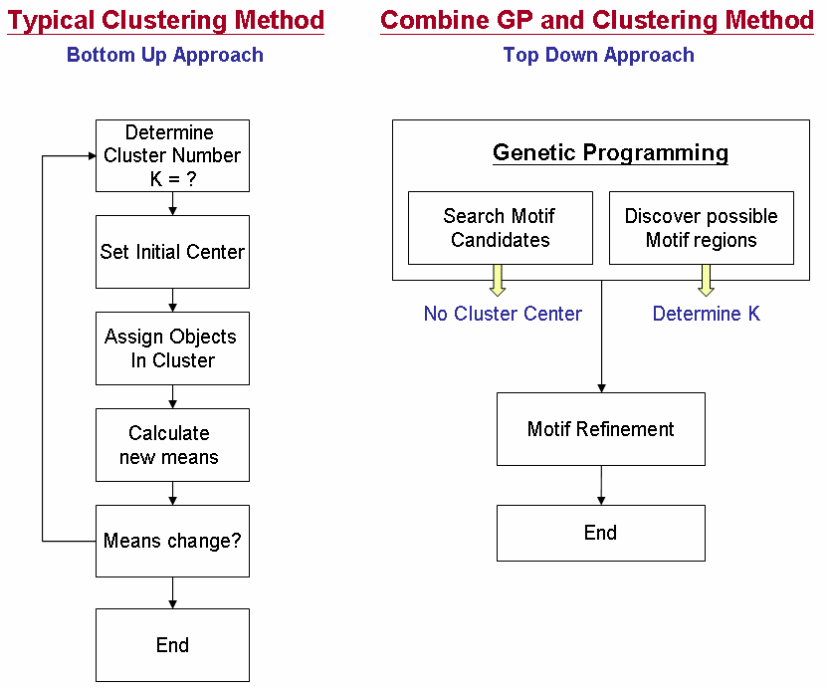


圖 5-9 典型分群法和本研究流程比較圖

一般的分群方法必須事先人為去訂定分群的群數(k)、及選擇每一群的起始中心點(center)，不同的預設值會大大影響最後的分群結果，這也是目前大部分分群法希望能改善之處。本研究結合了基因規劃法自我學習的優點，透過適應性函數讓系統能自動地學習並找出分群數，免除掉人為憑直覺設定所產生的缺點。表 5-3 為傳統分群法和結合基因規劃法的分群法比較表：

	Typical Clustering Method	Combine GP and Clustering
# of Clusters	Try and error	Learning
Cluster Center	Determine by random	No center
Order of input object	Sensitive	Insensitive
Noise Data	Sensitive	Insensitive

表 5-3 典型分群法和本研究模型比較表

5.2.2 生物學上的貢獻

5.2.2.1 解決過去研究上的問題及瓶頸

一般尋找蛋白質相似結構元的方法中，多重結構比對時的比對順序、比對標準的選定(reference)，或是相似結構元中胺基酸突變允許的範圍等等老問題，目前為止還沒有真正可以徹底解決的好方法，也一直深深困擾著此領域的研究學者。本研究在不同的模型階段中，提供了一個可以解決了上述傳統老問題的方法。

多重結構比對時會因為比對順序不同而有不同的比對結果。本研究後置作業的「產生代表性相似結構元」步驟中，從數個相似但不同的結構元所包含的區域結構中，依照區域結構在胺基酸序列上發生的位置挑選出可能的區域結構組合，若此組合不符合相似結構元的標準再逐一刪去最不像的區域結構。有別於過去方法是判斷子結構是否相似，相似的再歸類為同一相似結構元，本研究以反向思考，先找出可能相似的一群子結構，再逐一刪去最不相像的，因此比對區域結構順序的不同便不會構成影響最終結果的因素。

在上述步驟中也同時解決了選定比對標準的困難。比對過程中是以整體的區域結構為標準來找出不像的，不管最後哪個子結構被淘汰，比對標準始終是整體區域結構，正好解決了不同比對標準而有不同結果的問題。

相似結構元中胺基酸突變情形也是個棘手問題。過去方法中大多是依據胺基酸的化性、物性、或是生物經驗來猜測哪些不同胺基酸可能會有相似的結構，方法簡單但很容易忽略掉特殊狀況，且常常因為考慮系統複雜性而將條件設定的非常嚴苛，忽略掉的資訊會更多。而本研究以系統隨機產生胺基酸突變的可

能組合，再透過適應性函數引導學習，慢慢淘汰不可能的組合。雖然沒辦法將各種可能組合一網打盡，但相較過去的方法是能包含到更多種的胺基酸突變組合。表 5-4 為本研究模型解決過去問題的整理：

Typical Problem	Solving Method	Model
Alignment Order	Filtering the dissimilar structure one by one	Post-processing
Refernce Motif	Choose an optimal motif from some candidates	Post-processing
Amino Acid Mutation	Generated by the system and measured by fitness function	Genetic Programming

表 5-4 本研究模型解決過去尋找相似結構元方法整理表

在第二章文獻探討中曾提到，I-sites 以及 Tendulkar 是以 Clustering 的演算法來搜尋相似結構元。I-sites 先是將胺基酸序列打散，再將這些子序列(subsequence)加以分群，分群好後再以結構再做一次分群，Tendulkar 則是選擇 56 個特性來分群。上述兩種方法都會面臨到不知道分幾群才適合、選擇群中心的問題。本研究則利用基因規劃法學習的優點，讓相似結構元彼此競爭的自然方式而非人為來找出最適當的分群數，改善了過去分群法在生物資訊應用領域上的問題。

5.2.3 研究方法的限制

本研究採用演化式計算中的基因規劃法為主要模型，而演化式計算很重要的部份是隨機化的演化過程，無法完全掌握最後的解。本模型也同樣面臨到隨機化的問題，不同的搜尋開始點會導向不同的結果，只要改變一開始的搜尋點，最後的結果就會不盡相同。幸運的是，蛋白質結構中存在結構相似區域是固定的，相似和不相似兩者之間又很好區別，可以大大降低因為隨機化的不同而造

成最後結果不同的問題。但，本系統雖能找到結構相似的區域，該相似結構區域到底包含了多少區域結構則是非常依賴隨機的過程，例如：一個確實應該被包含的區域結構會因為隨機演化過程中並沒有隨機挑到他而被忽略掉；或是不是那麼相像的區域結構，比真正相像的區域結構先被選中而挑入成為相似結構元的一部分。

為了降低隨機化所造成的解的不穩定性，增加了後置作業中分群的步驟，期望就算是隨機被劃分在不同相似結構元、或是根本不該出現的區域結構，能透過分群的步驟整合並予以計算、逐一淘汰最不像的區域結構，來降低隨機化所帶來的不穩定性。雖然，經過如此層層關卡篩選，但也僅是降低隨機化的影響而非徹底剔除，這也是演化式計算最典型也最根本難解的問題。

由於上述隨機化的問題，影響了本研究最後找到的相似結構元中所包含的區域結構。以 EF-hand 為例，PRODEC 的四個相關相似結構元並不能偵測到完整的功能區域，除了認定結構相似度較嚴謹外，隨機過程也是影響重大的原因之一，或許任一個相似結構元踢除掉或增加另一個新的區域結構就能偵測到完整的 EF-hand，這完全端看隨機化的過程中如何發展。當然，若是已經有標準答案可供演化過程中的學習，隨機化的負面效應能因此大大降低，但至今還沒有公認最完整的相似結構元資料庫，因此難以透過適應性函數納入錯誤答案等方式來增加系統穩定度。期望未來能有相關正確且完整的資料庫出現，便能大大幫助此演化式計算模型使用者追求系統穩定度。

另外，本研究仍不能解決蛋白質演化中棘手的 Insertion、Deletion 等問題，設計的適應性函數並沒有針對這兩種情況的評分方式，所以在演化學習過程中就無法解決此兩個問題。

5.2.4 模型的應用

本研究提出的模型是非常具有彈性的且能運用在其他複雜的生物領域中。依據每次研究不同的主題，前置作業模型中給予不同的實驗資料，基因規劃模型中設計不同的適應性函數引導模型找到預定的目標，後置作業中給予不同的篩選條件而達到最後的目的。每階段都是非常有彈性，很容易套用至所需要的研究領域中。

5.3 未來展望

首先，實驗資料上我們是採用相似度低於 40%的蛋白質序列，是為了避免基因規劃學習的過程中被太類似的結構给牽引，導致找到的相似結構元普遍性不夠，換言之，普遍性不夠的這個缺點不是自然界蛋白質特性所造成，而是基因規劃系統中的適應性函數沒辦法解決而造成的。因此，若是能設計出另一個函數能更精確的評量相似結構元的普遍性並加入原有的適應性函數中，並以蛋白質資料庫中所有的蛋白質為實驗對象，我們相信一定能有效改善普遍性不夠的這個問題。

實驗資料另一個可以改進的地方是資料庫的選用，目前我們僅採用蛋白質分類資料庫(SCOP)中的資料，將來可以用其他分類資料庫，例如 CATH 等等來驗證模型的可靠性及可用性。

另外，本研究的實驗對象僅為 a.49 這個蛋白質分類，雖然結果還不錯，但不足以說明本模型的穩定性。未來我們可以把 SCOP 中所有的 Fold 當做實驗資料，以得到的大範圍、完整的實驗結果來更精準的說明研究模型的特性，便能更清楚知道模型的優缺點及適合應用的領域。

附錄 1 –1 PROSITE 相似結構元

No	Motif	RMSD	ACC	RRMSD	No. of Proteins	No. of Fragments	Length	Annotation
PS00001	N-{P}-[ST]-{P}	6.6702	0.5276	0.46	18	23	4	N-glycosylation site
PS00002	S-G-x–G	N/A	N/A	N/A	1	1	4	Glycosaminoglycan attachment site
PS00003	Rule	3.7330	0.8208	0.42	8	9	15	Tyrosine sulfation site
PS00004	[RK](2)-x-[ST]	1.5414	0.7431	0.49	10	11	4	cAMP- and cGMP-dependent protein kinase phosphorylation site
PS00005	[ST]-x-[RK]	3.6731	0.5169	0.53	45	79	3	S or T is the phosphorylation site
PS00006	[ST]-x(2)-[DE]	3.4205	0.6541	0.54	48	155	4	S,Casein kinase II phosphorylation site
PS0007	[RK]-x(2)-[DE]-x(3)-Y or [RK]-x(3)-[DE]-x(2)-Y	4.4816	0.7912	0.23	2	2	8	Tyrosine kinase phosphorylation site ,Y is the phosphorylation site
PS00008	G-{EDRKHPFYW}-x(2)-[STAGCN]-{P}	4.5562	0.7795	0.29	27	46	6	G is the N - myristoylation site],N-myristoylation site
PS00009	x-G-[RK]-[RK]	2.229	0.7371	0.48	2	2	4	Amidation site
PS00015	Rule	N/A	N/A	N/A	1	1	17	Bipartite nuclear targeting sequence

附錄 1－2 PROSITE 相似結構元

No	Motif	RMSD	ACC	RRMSD	No. of Proteins	No. of Fragments	Length	Annotation
PS00016	R-G-D	N/A	N/A	N/A	1	1	3	Cell attachment sequence
PS00018	D-x-[DNS]-{ILVFYW}-[DENSTG]-[DNQGHRK]-{GP}-[LIVMC]-[DENQSTAGC]-x(2)-[DE]-[LIVMFYW]	6.6702	0.5790	0.39	33	51	23	EF-hand calcium-binding domain
PS00029	L-x(6)-L-x(6)-L-x(6)-L	5.4818	0.8143	0.46	2	2	22	Leucine zipper pattern
PS00039	[LIVMF](2)-D-E-A-D-[RKEN]-x-[LIVMFYGSTN]	N/A	N/A	N/A	1	1	9	DEAD-box subfamily ATP-dependent helicases signature
PS00303	[LIVMFYW](2)-x(2)-[LK]-D-x(3)-[DN]-x(3)-[DNSG]-[FY]-x-[ES]-[FYVC]-x(2)- [LIVMFS]-[LIVMF]	5. 6306	0.7778	0.41	6	6	22	S-100/ICaBP type calcium binding protein signature
PS00342	[STAGCN]-[RKH]-[LIVMAFY]	0.6717	0.9766	0.17	2	2	3	Microbodies C-terminal targeting signal
PS00613	F-P-x-R-[IM]-x-D-W-L-x-[NQ]	N/A	N/A	N/A	1	1	11	Osteonectin domain signature 2

附錄 2 – Foldzilla 及 Wangikar 相似結構元

No	RMSD	ACC	No. of Proteins	No. of Fragments	Length	Fold Specificity	Annotation
MTF00052	4.3782	0.8680	20	33	29	100%	EF-hand superfamily (a.39.1) Calcium binding region I
MTF00053	2.7114	0.9340	7	7	17	71.43%	EF-hand superfamily (a.39.1) Calcium binding region II
MTF00054	1.0142	0.9801	6	6	13	8.33%	None
MTF00055	0.8580	0.9908	7	7	13	6.03%	None
Wangikar	N/A	N/A	36	63	8	N/A	EF-hand：calcium binding loop

參考資料

1. Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov and Philip E. Bourne (2000), The Protein Data Bank, *Nucleic Acids Research*, Vol. 28, No. 1 235-24
2. Higgins D., Thompson J., Gibson T.Thompson J.D., Higgins D.G., Gibson T.J. (1994), CLUSTAL W: improving the sensitivity of progressivemultiple sequence alignment through sequence weighting,position-specific gap penalties and weight matrix choice.*Nucleic Acids Research*, Vol. 22, 4673-4680.
3. Ramachandran N, Colman RF (1977), Evidence for a critical glutamyl and an aspartyl residue in the function of pig heart diphosphopyridine nucleotide dependent isocitrate dehydrogenase, *Biochemistry*, 16(8):1564-73
4. Frédéric Guyon*, Anne-Claude Camproux, Joëlle Hochez and Pierre Tufféry (2004), SA-Search: a web tool for protein structure mining based on a Structural Alphabet, *Nucleic Acids Research,* Vol. 32
5. Catherine Etchebest, Cristina Benros, Serge Hazout, Alexandre G. de Brever n (2005), A structural alphabet for local protein structures: Improved prediction methods, *Proteins: Structure, Function, and Genetic,* Vol. 59, 810 – 827
6. Tendulkar AV, Joshi AA, Sohoni MA, Wangikar PP (2004), Clustering of protein structural fragments reveals modular building block approach of nature, *J Mol Biol,* 338(3):611-29
7. Inge Jonassen, Ingvar Eidhammer, William R. Taylor (1999), Discovery of Local Packing Motifs in Protein Structures, *Proteins: Structure, Function, and Genetic,* Vol. 64, 206 – 219
8. Inge Jonassen, Ingvar Eidhammer, Darrell Conklin, William R. Taylor (2001) , Structure motif discovery and mining the PDB, *Bioinformatics*, Vol, 18, 362-367
9. Philip Bradley, Peter S. Kim, Bonnie Berger (2002), TRILOGY : Discovery of sequence-structure patterns across diverse proteins, *PNAS*, Vol. 99, 8500-8505
10. Christopher Bystreoff, David Baker (1998), Prediction of Local Structure in Proteins Using a Library of Sequence-Structure Motifs, *J. Mol, Biol,* Vol. 281, 565-577
11. Roman L. Tatusov, Stephen F. Altschul, Eugene V. Koonin (1994), Detection of conserved segments in proteins : Iterative scanning of sequence datatbases with alignment blocks, *PNAS*, Vol. 91, 12091-12095
12. Karen F. Han, David Baker (1995), Recurring Local Sequence Motifs in Proteins, *J. Mol. Biol.*, Vol. 251, 176-187
13. Jimmy Y. Huang, Douglas L. Brutlag (2001), The EMOTIF database, *Nucleic Acids Research,* Vol. 29
14. Gerard J. Kleywegt, Recognition of Spatial Motifs in Protein Structures (1999),

J. Mol, Biol, Vol. 285, 1887-1897
15. Christine A. Orengo, William R, Taylor (1993), A Local Alignment Method for Protein Structure Motifs, *J. Mol, Biol,* Vol. 233, 488-497
16. Christopher Bystroff, Kim Simons, Karen F Han, David Baker (1996), Biotechnology, Vol.7, 417-421
17. Lissa Holm, Chris Sander (1999), Protein folds and families : sequence and structure aligments, *Nucleic Acids Research,* Vol.27
18. John Moult (1999), Prediction protein three-dimensional structure, *Biotechnology*, Vol. 10, 583-588
19. Jessica Shapiro, Douglas Brutlag (2004), FoldMiner: Structural motif discovery using an improved superposition algorithm, *Protein Science*, Vol.13, 278-294
20. James O. Wrabl, Nick V. Grishin (2004), Gaps in Structurally Similar Proteins: Towards improvement of Multiple Sequence Alignment, *Proteins: Structure, Function, and Genetic,* Vol.54, 71-87
21. O. Dror, H. Benyamini, R. Nussinov, H. Wolfson (2003), MASS: multiple structural alignment by secondary sturcutres, *Bioinformatics*, Vol.19, 95-104
22. Cedric Notredame, Desmond G. Higgins, Jaap Heringa (2000), T-Coffee: A Novel Method for Fast and Accurate Multiple Sequence Alignment, *J. Mol, Biol,* Vol. 302, 205-217
23. 林順富、陳師瑩、顏瑞鴻、蕭慧美，生物化學，2001
24. Ashish V. Tendulkar, Anand A. Joshi, Milind A. Sohoni, Pramod P. Wangikar (2004), Clustring of Protein Structural Fragments Reveals Modular Building Block Approach of Nature, *J. Mol, Biol,* Vol. 338, 611-629
25. Laurent Falquet, Marco Pagni, Philipp Bucher, Nicolas Hulo, Christian J. A. Sigrist, Kay Hofmann, Amon Bairoch (2002), The PROSITE database, its status in 2002, *Nucleic Acids Research,* Vol.30
26. Ta-Tsen Soong, Cheng-Yu Chen, Ming-Jing Hwang (2004), Conserved Local Structural Motifs for Functional Annotation of Protein Families. (in preparation)