

貨幣數量學說的競爭性：遺傳規畫在 知識發掘上的應用*

陳樹衡** 葉佳炫***

壹、緒論

實証研究工作者在從事資料分析中，所要面臨的問題，雖然是千頭萬緒，但是一個最根本的問題仍然是：如何在現有的資料中去做好知識發掘 (knowledge discovery) 的工作。¹ 這個問題不僅構成了社會科學與統計學一個交匯的領域，同時也是人工智慧與統計學中一個交匯熱絡的領域。很遺憾地，以上兩個交匯彼此之間似乎卻缺乏再交匯。舉個例子而言，以人工智慧與統計學會 (Society for Artificial Intelligence and Statistics) 1995年的年會所發表的62篇論文來看，幾乎所有有關應用方面的論文，都是分散在生物統計、醫療計量、語言學…等領域裡。應用在社會科學方面的論文，可以說是鳳毛麟角。而本文的目的就是希望能夠對此交匯有所貢獻。換句話說，我們希望藉著這篇文章，能讓社會科學中的實証工作者體認到：以資料庫為導向的知識發掘工作，是一個橫跨社會科學、統計學與人工智慧，一種高

* 本文承蒙管中閔教授以及兩位匿名評審提出寶貴的意見，在此一併致謝。

** 政治大學經濟系副教授

*** 政治大學經濟系博士班研究生

度科技整合性的工作，如果我們能更深入地融入在這個整合的領域中，或許將有助於社會科學實証工作的提昇。

有鑑於此，本文的構想如下：首先，我們將在第二節中，介紹一個在人工智慧領域中，最新發展的工具，即遺傳規畫 (Genetic Programming)。遺傳規畫在晚近人工智慧與統計學就有關以資料庫為導向的知識發掘工作中，扮演著一個很有潛力的角色。其次，我們要問的就是，遺傳規畫作為一個發掘知識的工具，它的功能性如何？對這樣的問題要有一個一般性的答案，在現階段而言，似乎並不容易。因此，本文的著眼點只是一個個案性的評估。我們希望能在一連串個案經驗的累積中，增加我們對遺傳規畫在發掘知識功能性上的認識，而奠定未來更一般性理論發展的基礎。

本文所使用的個案，就是在經濟學上歷久不衰的貨幣數量學說。我們的作法是在第三節中，建立一個由與貨幣數量學說相關變數所組成的小型資料庫。然後在「假裝」我們對該資料庫中缺乏任何知識的情況下，來測試遺傳規畫發掘此一知識的能力。如果我們對它的這個能力有信心，那麼將來在我們對某一資料庫「真正」缺乏知識時，遺傳規畫才會被有信心地使用作為發掘知識的工具。因此針對這點，我們的個案研究所要問的是：如果貨幣數量學說基本上是隱藏在我們的資料庫中，則遺傳規畫能不能發現此一學說。

第二，當我們對資料庫中的某一知識雖然掌握正確，但卻不是很完全時，那麼遺傳規畫是否能發掘並完備該知識。譬如說，我們雖然知道經濟資料庫中，擁有貨幣數量學說的知識，但是當我們並不確定流通速度的形式時，遺傳規畫在發掘出貨幣數量學說的同時，是否能告訴我們流通速度的形式？這點是要了解遺傳規畫對知識完備性上的能力。

第三，遺傳規畫所發掘(或完備)出來的「知識」究竟能不能算知識？譬如說，當我們資料庫中並不蘊涵貨幣數量學說，此時遺傳規畫

可能不會發現或完備該一學說。相反地，它可能為我們找尋出它認為對的知識。但是這個知識對我們有沒有用呢？這個問題在知識發掘的領域中，已逐漸受到重視 (Lubinsky, 1995)。此乃是因為人工智慧所使用的工具對其所發掘知識的表達形式，經常缺乏可理解性。因此，從這個理解的角度而言，這些知識並沒有用。但是，若就預測的眼光而言，則這些不易消化的知識，往往卻有益於提高預測能力。所以，從預測的角度上來說，這些知識又顯得有用。因此，要怎麼樣在可理解與可預測之間決定知識的有用性，並非一件容易的事。本文會在後面繼續對這點加以探討。

在這樣的構想下，我們將如下安排本文的章節。在第二節中，我們將對遺傳規畫做一個簡單的介紹。在第三節中，我們將敘述我們的經濟資料庫及在該資料庫中一個歷史悠久的經濟知識——即貨幣數量學說。貨幣數量學說若從洛克 (John Locke) 算起，已有三百年的歷史。其中歷代經濟學者的貢獻，已大大豐富它的內容。本文所用來代表貨幣數量學說的貨幣數量方程式 (Quantity Equation)，只是貨幣數量學說中的一種版本。第四節裡，我們將開始探討遺傳規畫作為一種在資料庫中發掘知識的工具，它能否發掘出此一學說。第五節中我們將以預測績效來衡量遺傳規畫所發掘出來的知識。最後一節則是結論。

貳、什麼是遺傳規畫？

要了解遺傳規畫之前，我們先要了解什麼是演化計算 (Evolutionary Computation)。顧名思義，演化計算是將生物學中達爾文「物競天擇，適者生存」的進化論，運用到「計算」的領域來。由於自人工智慧領域正式誕生之前，學者對於「智慧」(intelligence) 的概念幾乎就與「計算」平行。最有名的例子就是 Turing 檢定 (Turing Test) (Crevier, 1993)。所以，就某個意義而言，演化計算可以視為生物學與人工智慧一個整合性研究下的產物。然而，更基本的問題是為

什麼這樣的整合有意義？要回答這個問題之前，就必須先了解什麼是計算？計算可以視為一個尋找答案的過程。² 這個尋找是要在解答空間 (solution space) 中，去找尋正確的答案。然而，一般而言，由於解答的數目可能相當龐大，靠著一股盲勁 (brutal force) 去找，或是隨機亂找，都是不切實際 (impractical) 或不可行 (infeasible) 的。因此，程式學 (Theory of Algorithms) 的目的，就是希望能研究出「有效」尋找的途徑。而演化型計算就是將演化的運作，明顯地使用到尋找答案的設計上來。這些運作包含了複製、交配與突變。而最重要的是它們創造了一個競爭性的環境，並彰顯了「弱肉強食，適者生存」的原則。

演化型計算概念的提出已有30年的歷史。剛開始時，它是出現在不同的領域，由不同的人用來處理不同的問題，在文獻上也分別有不同的名詞，如 Rechenberg (1965) 的演化策略 (Evolutionary Strategies)，Fogel 等 (1966) 的演化規畫 (Evolutionary Programming)，Holland (1975) 的遺傳程式 (Genetic Algorithms) 及 Koza (1989) 的遺傳規畫。早年這些支派的整合性很低。直到最近，對它們彼此之間關連性的研究才起步。而演化型計算的名詞的出現，也是象徵這一個綜合性研究的開始。誠如前面提到的，本文所使用遺傳規畫就是演化型計算下的一個支派。從淵源上來講，遺傳規畫是遺傳程式的後續發展，但是也正因為此，就它的格局而言，它似乎是目前演化性計算各種支派中，最具一般性者。

遺傳規畫基本上是將整個解答空間中的每一個解答，用一個電腦程式 (computer program) 來代表。而又因為如果每個電腦程式都用 LISP (List Programming) 語言來表示的話，則它們都具有 LISP 的一種樹狀展開表現法。³ 所以，在 LISP 語言下，每一個解答都表現成一棵樹，而整個解答空間就是一個森林。遺傳規畫的任務，就是要在這個森林中找尋一棵最好的樹。它的作法首先是透過一個隨機程序產生一個原始森林。這個原始森林像一般生物一樣，具有繁衍的能力，也就是原始森林能繁衍出第一代森林，而第一代森林又能繁衍出第二代森

林。繁衍的方式也不外是複製 (reproduction) 與交配 (crossover)。問題是並不是所有的樹都有一樣的機會能使用複製或交配去繁衍它的下一代。相反地，複製與交配的機會是有選擇性的。凡是對環境有良好適應力 (fitness) 的樹，自然就會較那些適應力較差的樹有更多的機會被複製，或被選來與其他樹交配而繁衍後代。除此之外，在繁衍的過程中，基因也可能發生突變。至於複製、交配與突變的實際執行，就構成了遺傳規畫程式設計中的細節，這些將會在附錄A中詳細交代。至於遺傳規畫概觀性的簡略流程圖，可以參考 Koza (1992a) 一書第五章中的圖 5.1。

根據以上的描述，遺傳規畫可以說是將進化原則應用到一個以電腦程式所組成的社會中。在優勝劣敗的競爭性環境下，自行繁衍與成長。將這個方法用到知識發掘的領域裡，則每一棵樹就代表一種假說或說是一個模型。而森林就是各種假說 (或模型) 的集合。原始森林則是最起初假說 (或模型) 的集合。而用以決定每個假說 (或模型) 複製或交配機會的適應力，則可定義為該假說 (或模型) 對資料庫中資料的解釋能力。複製的意義在這裡很明顯。至於交配，則可視為兩種假說 (或模型) 的結合。突變則是某一假說 (或模型) 中的某一部份，在未經交配的情況下，自動性地變化。而在這樣的對比下，知識發掘的工作就自然可視為一個「物競天擇，適者生存」的競爭性過程。這個過程將淘汰掉不好的假說，而保留並尋找好的假說。這個過程也是使遺傳規畫能成為知識發掘工具的一個基本條件。至於成功的交配與突變，則可視為對現存優良假說的一個完備或補強。

除了知識的發掘與完備之外，從另一個角度來講，遺傳規畫也提供對任何假說一個檢試的環境。一個假說若要能夠被接受，必須是它有強大的競爭或存活力。遺傳規畫透過參數的設計，如競爭的模型數 (森林規模)，或競爭的代數 (時間)，可以增加或減弱一個環境的競爭程度。例如，森林規模愈大時間愈長的競爭愈是激烈。而在愈激烈的

環境中，能存活下來的模型，其競爭力便愈強，而可接受性便也就愈高。

從以上的說明中，我們也可以看出遺傳規畫和一般統計模型的關係。一般的統計模型無論是母數或是無母數在估計它的一些參數時，都會使用到某種數值方法作為尋找過程的設計依據。但是，這些尋找方式基本上而言仍是「單點到單點」(single-point-to-point)的尋找，而遺傳規畫(或遺傳程式)則是「多點到多點」(multiple-points-to-points)的尋找。因此，在克服區域最適(local optimum)的陷阱上，後者應有較佳的效率。⁴然而，這並不是意謂遺傳規畫與一般統計處於對立競爭的地位。比較合宜的看法應是遺傳規畫是提供一種新的尋找典範，這種典範可以應用來提高處理一般統計模型中尋找上的效率。譬如說，它可以用改進逐步迴歸中的尋找效率，也可以用來改良神經網路(Neural Networks)中的尋找績效(Angeline, Saunders and Pollack (1994))。

參、遺傳規畫與貨幣數量方程式

Koza (1992b) 是遺傳規畫在經濟學上最早的應用。在他的那篇文章中，Koza 強調遺傳規畫可以用來發現潛在的知識。他並用貨幣數量方程式 $P = \frac{MV}{T}$ 做例子，說明這樣一個蘊藏在資料中的知識，可以透過遺傳規畫重新發現。本文以下將重新執行 Koza 的研究，這樣做有幾個目的，第一、遺傳規畫是個隨機尋找過程，每次尋找的答案未必是一樣的。然而，Koza 對這點似乎沒有提到。在他整篇文章中，他只展示了「一次」執行的結果。這使人難以了解這次究竟是「一次中的一次(100%)」，還是許多次中的一次。畢竟，我們所關切的是：遺傳規畫是不是每一次都能學到貨幣數量方程式。如果不是，則學到的機率或困難有多高？這些都是我們在評價遺傳規畫發掘知識的能力

時，不可或缺的了解。第二、Koza 所執行的遺傳規畫不僅成功地發現了貨幣數量方程式，同時，還得到流通速度是常數的結果。只是這個結果經過驗証，似乎並不正確，Koza 的結論是根據他文中70頁(圖7)的展示的樹狀程式推導而來。但是，這個樹狀程式所代表的一個如下的函數：

$$P = \frac{M2[1.234366 + h(.)]}{GNP}$$

其中，

$$h(.) = \frac{M2(0.47 * GNP + 0.00235)}{0.47 * GNP^2 - 0.3749 * GNP + 0.0003008}$$

很明顯地，這個函數並無法簡化成一個常數。Koza 似乎想用此結果來說明遺傳規畫所發現的與他所熟悉的計量文獻「竟然」完全一致。我們認為這裡有點誇張的嫌疑。因此，重新執行他的作業，也是想再度確認這點。最後，我們也想了解 Koza 的這套方法執行在臺灣的資料會得到些什麼結果。爲了確定我們所使用的方式與 Koza 相同，重新再執行 Koza 的作業是有必要的。而且只有在我們執行的方式與 Koza 相同時，遺傳規畫在美國資料與臺灣資料對貨幣數量方程式發掘過程的比較才會有意義。

所以，基於以上這些理由，我們將在第四節中重新執行 Koza 的遺傳規畫，然後在確定我們的遺傳規畫與 Koza 是相同的版本後，再將這套程式執行在臺灣。同時，並比較它與美國經驗的異同。在此之前，讓我們先對資料來源做一簡單的說明。

本文中所使用的資料是從 EPS 資料庫中取得，相關資訊請參見表1與表2。除了貨幣供給量與臺灣的利率是月資料外，其餘皆是季資料。而貨幣供給量與臺灣利率的季資料，是以3個月的平均數作爲其季資料。美國的資料是從 1963: 1 到 1992: 4，而臺灣的資料則是從 1962: 1 到 1991: 4。因此兩國皆有120個觀察值。

表 1：美國資料

代號	意義	位址
PGNP	GNP 平減指數	USCEN
GNP87	實質 GNP (基期 = 1987)	USCEN
MNY2	M2 貨幣供給	USCEN
60c	國庫券利率 (三個月)	IMF 之 IFS

表 2：臺灣資料

代號	意義	位址
PGNP	GNP 平減指數	NIAQ
GNP86	實質 GNP (基期 = 1986)	NIAQ
M2	M2 貨幣供給	FSM
RMTD3@1ST	第一商業銀行存款牌告利率一定定期存款 (三個月)	FSM

由於貨幣數量方程式是以 $MV = PT$ 來表示，當我們在等號兩邊皆取 \log 後，可以得到下列的線性關係：

$$\log(P) + \log(T) = \log(M) + \log(V) \quad (1)$$

而方程式 (1) 又可改寫成

$$\log(P) = \log(M) - \log(T) + \log(V) \quad (2)$$

因為在我們資料庫中 V 不是直接可觀察的，所以，遺傳規畫在執行中，就會自動用我們所提供的其他變數來代替。此時，遺傳規畫實際上所要發掘的知識應是方程式 (3)。

$$\log(P) = \log(M) - \log(T) + k(M, T, IR) \quad (3)$$

其中「 IR 」表示利率。方程式 (3) 等號的右邊可以分成兩部份，其中一部份是 $\log(M)$ 與 $\log(T)$ 的線性部份。而另一部份則是原來 $\log(V)$

的替代變數， k 。如上所述， k 一般而言是資料庫其他變數的函數。至於，這個函數的形式並沒有任何限制。一般而言，它可以是非線性。在方程式 (3) 的基礎上，如果遺傳規畫成功地發現了方程式 (3) 中的線性部份，我們便說遺傳規畫發掘到了貨幣數量學說的知識。在這種情況下，我們也可以說貨幣數量學說在我們資料中具有競爭性。而此時的 $k(\cdot)$ 則視為遺傳規畫對貨幣數量學說中流通速度函數形式的補充。反之，若是遺傳規畫的學習並沒有發現方程式 (3) 中的線性部份，則我們便說遺傳規畫並沒有發掘到貨幣數量學說的知識，或者說貨幣數量學說在我們的資料中，並沒有競爭性。以下我們便根據表1與表2的資料來比較貨幣數量學說在美國與臺灣的競爭性。亦即在我們的分析中，我們是用「PGNP」代表 P ，「GNP87」(臺灣則以「GNP86」) 代表「 T 」，「 $M2$ 」代表「 M 」，三個月國庫券利率(臺灣則以第一商業銀行三個月定期存款牌告利率) 代表「 IR 」。

表 3：線性迴歸模型估計的係數 (美國)

參數估計				
變數	估計值	標準差	t-值	p-值
Intercept	-0.941802	0.39298943	-2.397	0.0182
$\log(GNP)$	-0.773090	0.06621536	-11.675	0.0001
$\log(M2)$	0.930928	0.02095407	44.427	0.0001
$\log(IR)$	0.045321	0.00672047	6.744	0.0001

$$SSE=0.07001, R^2=0.9977, \overline{R^2}=0.9977, DW=0.152$$

肆、實證結果

首先，我們對 $\log(P)$ 做迴歸分析，結果列於表 3 中。而 $\log(GNP)$ 、 $\log(M2)$ 和 $\log(IR)$ 為自變數，其中適合性 (fitness) 定義為誤差平方和 (SSE)。此處迴歸分析得到的 SSE 為 0.07。在當代計量經濟

學發展中，當我們去跑這樣一個簡單的線性迴歸時，有許多問題要注意。其中，在近來文獻發展上較引人矚目的就是 Granger 與 Newbold (1974) 所提出的假迴歸 (spurious regression)。假迴歸的發展到了 80 年代在計量經濟學中，產生了一些重要的概念，尤其是單根 (unit root) 與共整合 (cointegration)。除了這些之外，發展於 70 年代早期，而在晚近已與共整合納入一個體系的 Granger causality 都會使得我們所使用的簡單迴歸方程式呈現地太簡單。所以，在這裡我們必須指出我們不是要用這條簡單迴歸方程式來「檢定」貨幣數量學說。我們只是用這條迴歸方程式中所產生的 SSE 來決定遺傳規畫這種自動尋找機能所得到的適合性是否可以接受。如果遺傳規畫的表現連我們可順手撿來的線性知識都不如的話。它的可接受性無疑地將受到很大的保留。至於如何將遺傳規畫配上當代的計量工具來使用，則自是未來應再研究的。⁵ 本研究基本上是平行於 Koza (1992b)，所以在希望能與 Koza 的結果做一個直接比較的前提下，對更複雜卻更嚴謹的層面暫且省略。

接下來，我們便按照第二節的敘述，用遺傳規畫在表 1 所代表的資料中，發動一個發掘知識的假說 (或模型) 競爭環境。表 4 與表 5 是該環境的參數設定。表 4 與表 5 的差別，只在於函數集合的不同。在表 4 中，函數集合包含了 \sin 與 \cos 函數，然而在表 5 中卻不包含這兩個函數。這些參數基本上可讓我們選擇演化 (競爭) 的規模及演化的過程。在規模方面包含了競爭對手的多寡 (族群個體總數)，及開放競爭的時間 (子孫代數)。而在過程方面則已包含了基本結構 (函數集合與終點集合)、創造 (完整法與成長法)、及演化運作元 (複製、交配、突變與新生嬰兒) 的進行方式。雖然遺傳規畫的結果會受設定參數的影響，但是這方面的研究尚無進展。⁶ 所以目前尚沒甚麼理論能告訴我們一個參數的設定是太高還是太低。所以，我們對參數值的設定基本上是依照 Koza (1992a, 1992b) 的方式。

表 4：模擬 a1-a5 的參數設定

子空間個體總數	500
原始子空間深度等級	2, 3, 4, 5, 6
每一等級完整法創造個體的數目	50
每一等級成長法創造個體的數目	50
函數集合的元素	{+, -, ×, %, exp, Rlog, Sin, Cos}
終點集合的元素	{GNP, M2, IR, R}
複製的個數	50
新生嬰兒的個數	50
突變的個數	100
突變的機率	0.2
個體長度的限制	17
終點被選為交配點的機率	0.5
子孫代數	100
配適值標準	SSE

對於表 4 與表 5，我們分別執行 5 次的模擬，其結果列於表 6 中。表 6 分別陳列各次模擬其 *SSE* 在第 0 代、50 代與 100 代的表現。其中 a1 到 a5 是根據表 4 所得到的結果，而 b1 到 b5 則是根據表 5 所得到的結果。從表 6 中，我們可以看到，雖然進化的速度不同，但進化的現象在每次的模擬中，都明顯地陳列。此外，我們也可以看到，在經過 100 代的演化後，依據表 4 所執行的 5 次模擬中，有兩次的結果優於線性迴歸所得到的結果。模擬 a2 與 a3 的 *SSE* 分別是 0.066 與 0.060。而在這十次的模擬中，模擬 a3 所選擇的模型是最具有競爭力的。其函數形式列於方程式 (4) 與 (5) 中。至於其它各次模擬所選擇的最佳模型，則列於附錄 B.1 中。為了便於比較，我們將模擬 a3 與 OLS 對 $\log(P)$ 的配適殘差值列於圖 1。

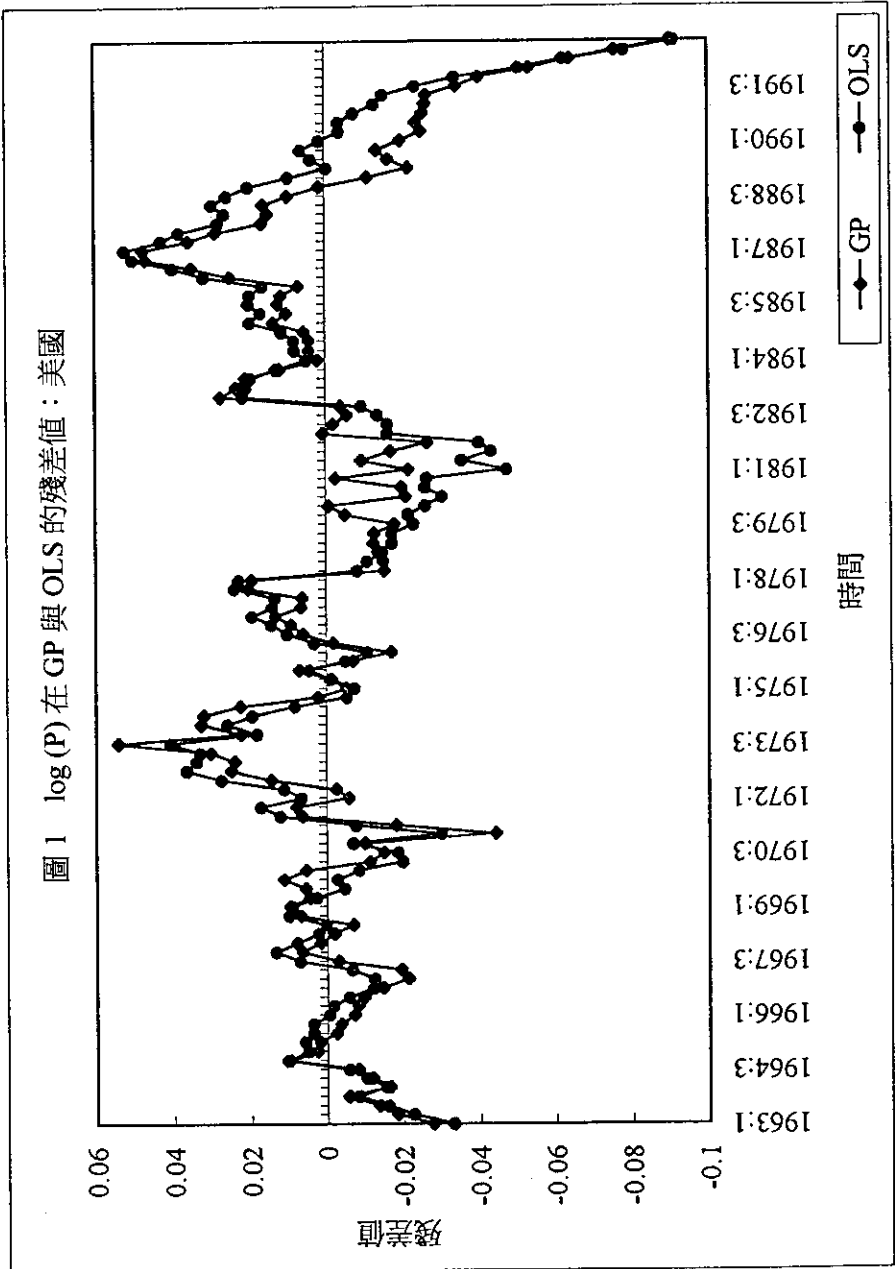
表 5：模擬 b1-b5 的參數設定

子空間個體總數	500
原始子空間深度等級	2, 3, 4, 5, 6
每一等級完整法創造個體的數目	50
每一等級成長法創造個體的數目	50
函數集合的元素	{+, -, ×, %, exp, Rlog}
終點集合的元素	{GNP, M2, IR, R}
複製的個數	50
新生嬰兒的個數	50
突變的個數	100
突變的機率	0.2
個體長度的限制	17
終點被選為交配點的機率	0.5
子孫代數	100
配適值標準	SSE

表 6：各次模擬在 Gen= 0,50 及 100 時的 SSE (美國)

模擬	GEN = 0	GEN = 50	GEN = 100
a1	16.6014356945	1.7560194444	1.5877132321
a2	4.5280953476	0.0943477508	0.0669074593
a3	18.4662807460	0.0662950995	0.0607039044
a4	14.1192882072	1.9335507499	1.5578049443
a5	9.7651900811	2.6750861048	1.6254611716
b1	24.2477124200	0.1209872472	0.0950912770
b2	7.4295429800	0.1248445692	0.1222076943
b3	6.7823155886	0.6041591037	0.5166354455
b4	24.7274992134	0.5533928640	0.1564576940
b5	24.4970863778	0.5991962402	0.2972428306

圖 1 $\log(P)$ 在 GP 與 OLS 的殘差值：美國



$$\log(P) = \log(M2) - \log(GNP) + k(GNP, M2, IR) \quad (4)$$

其中

$$k(GNP, M2, IR) = ((gnp / (((((gnp / ((m2 - gnp) - gnp)) + gnp) - (-0.56)) + ((gnp + (ir * (((sin(m2 / gnp) + gnp) * sin(-9.54)) - (sin(m2 / ((gnp * (-0.56 - gnp)) * (gnp / gnp)))) + gnp)))) * sin((ir / gnp) * (gnp - m2)))) - (-9.54)))) \quad (5)$$

且

$$gnp = \log(GNP) \quad m2 = \log(M2) \quad ir = \log(IR)$$

由於(4)在線性的部份完全符合貨幣數量學說的形式，因此我們基本上認為遺傳規畫發掘到了貨幣數量學說的知識。這點與 Koza (1992b) 的發現是一致的。我們也必須注意到：遺傳規畫並不是在每一次都學到了貨幣數量學說，這點是 Koza (1992b) 所沒有指出的。事實上根據附錄 B.1，在我們所做的十次模擬中，只有模擬 a3、b1 與 b2 三次成功地學到貨幣數量學說。換句話說，遺傳規畫在美國資料中，成功發掘貨幣數量學說的機率是30%。

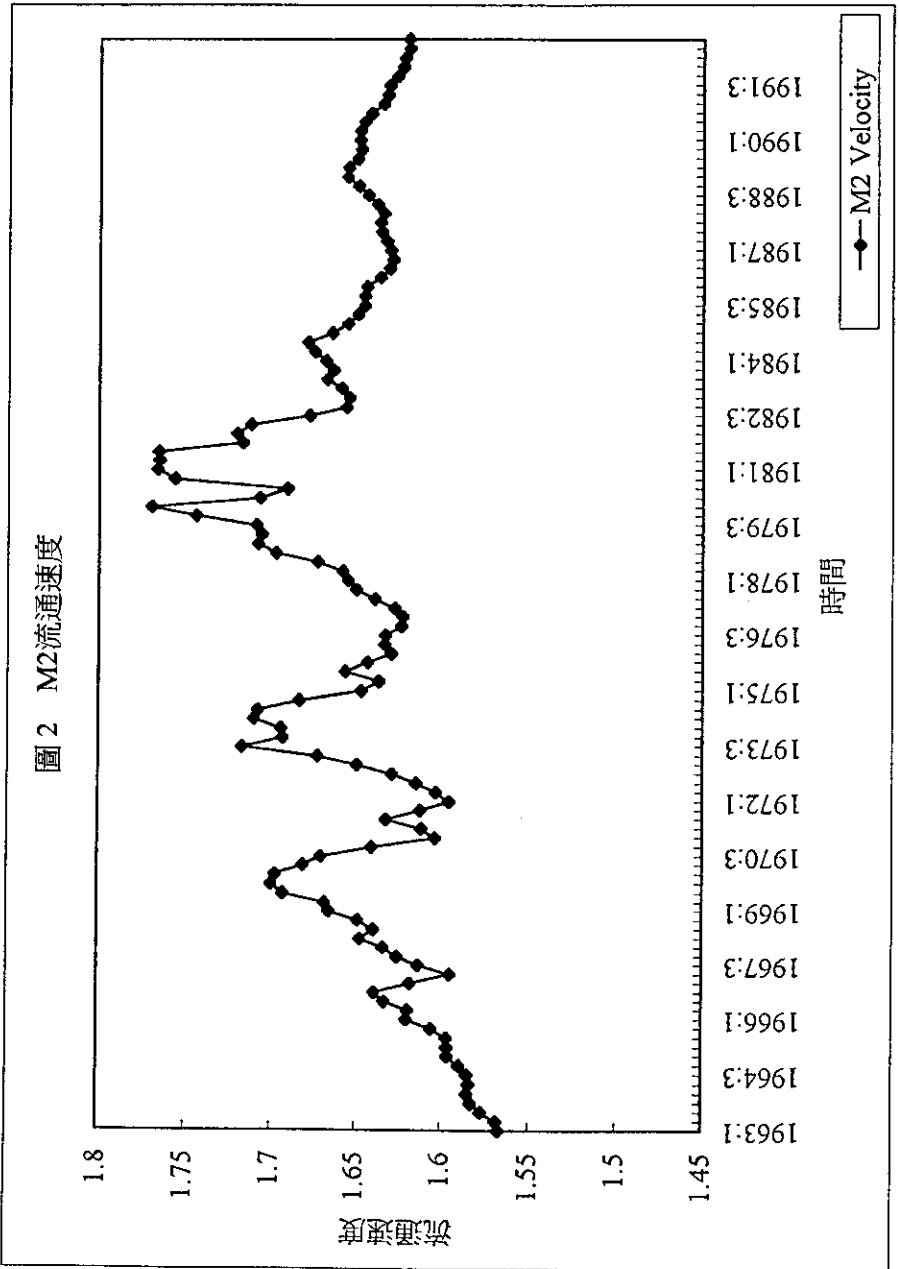
此外，若是將貨幣數量學說中流通速度的函數形式分為常數與非常數，則我們所得到的結果與 Koza 並不一致。Koza 得到的流通速度是一個常數項，即 1.6527。而流通速度為常數的貨幣數量學說，則只是我們三個學到貨幣數量學說模擬中的一個，即模擬 b2。根據附錄 B.1，模擬 b2 所學到的正好就是流通速度為常數 1.6610 的貨幣數量學說。問題是根據表 6，模擬 b2 所產生的最佳模型，在十次模擬中並不是最具有競爭力的模型。相反地，在最具競爭力的模擬 a3 中，我們所得到的流通速度則如方程式(5)所示，它是一個利率、GNP 與 M2 的非線性函數，而非一個常數。

在圖 2 中，我們將方程式 (5) 所估計出的流通速度陳列出來。我們可以看出雖然流通速度不是一個常數，但它是相當穩定地在 1.5 與 1.8 之間波動。而它的平均數是 1.6508，與 Koza 的也非常接近。Hallman et al. (1989) 與 Humphrey (1989) 均認為美國在戰後 M2 流通速度是相當穩定的。我們遺傳規畫上的發現，基本上是支持他們的觀點。

然而這個結果在其他的個案是否一樣會成立呢？我們以相同的程序測試臺灣的資料。線性迴歸的結果列於表 7 中，此時的 *SSE* 為 0.706。同樣地，我們分別依據表 4 與表 5 的參數設計對遺傳規畫進行 5 次的模擬。表 8 則是各次模擬中，最佳模型在第 0 代、50 代與 100 代的 *SSE*。其中 a1 到 a5 是在表 4 下所得到的結果，而 b1 到 b5 則是在表 5 下所得到的結果。與執行美國資料的結果相似，在 100 代的演化後，表 4 中的參數所執行的 5 次模擬中，有兩次的結果優於線性迴歸所得到的結果。模擬 a4 與 a5 的 *SSE* 分別是 0.634 與 0.506。最具競爭力的模型是由模擬 a5 所得到的，其函數形式列於方程式 (6)，而每次模擬所選擇的最佳模型則列於附錄 B.2 中。同樣爲了比較，我們將模擬 a5 與 OLS 對 $\log(P)$ 的配適殘差值列於圖 3。

$$\begin{aligned} \log(P) = & (\log(((gnp + ((\log(((((-7.19 - gnp) - (((m2 - gnp) - (gnp \\ & + gnp)) - gnp) * ir)) / gnp) - ir) - (m2 * m2)) + gnp) - \log((((gnp + gnp) \\ & - m2) - ((m2 - gnp) * (((gnp + ir) + ir) - (m2 - m2)))))) / gnp) - \\ & ir))) + gnp) - (m2 * m2)) - \exp.\log.\sin.\log((m2 - gnp) - (((m2 - \\ & \cos(m2 + (m2 - (m2 - m2)))) - \cos(m2 + (m2 - (m2 - (-4.25)))))) * \\ & ((-4.25 + ((gnp + gnp) - (gnp * m2))) - (((m2 + ir) + ir) - (m2 - gnp)) * \\ & m2)))))) \end{aligned} \quad (6)$$

圖 2 M2流通速度



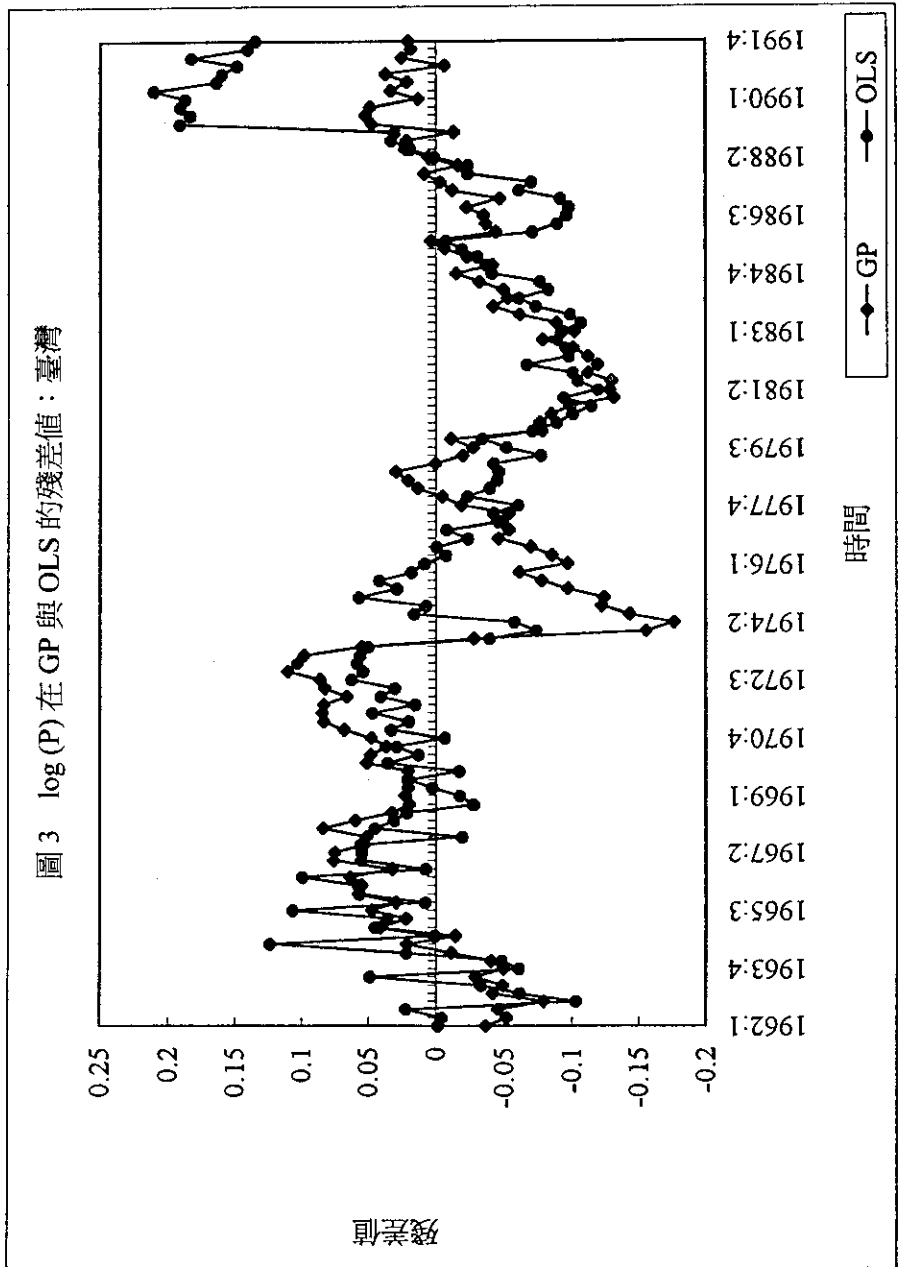


表 7：線性迴歸模型估計的係數 (臺灣)

參數估計				
變數	估計值	標準差	t-值	p-值
Intercept	5.150583	0.94570982	5.446	0.0001
$\log(GNP)$	-0.753168	0.12869377	-5.852	0.0001
$\log(M2)$	0.615813	0.05337688	11.537	0.0001
$\log(IR)$	0.245289	0.02838309	8.642	0.0001

$$SSE = 0.70620, R^2 = 0.9821, \bar{R}^2 = 0.9816, DW = 0.283$$

表 8：各次模擬在 $Gen = 0, 50$ 及 100 時的 SSE (臺灣)

模擬	$GEN = 0$	$GEN = 50$	$GEN = 100$
a1	39.4751786496	1.6001926406	1.6001926406
a2	41.3487536682	0.7455070149	0.7315891104
a3	26.8567646949	1.8955262774	1.1959025085
a4	33.0949060660	0.8526333263	0.6346563918
a5	22.6192243478	1.2386248062	0.5062178911
b1	32.3602415962	0.8990328900	0.8445017865
b2	35.7581759220	1.2207678276	1.0602512173
b3	12.4748416357	1.3310489601	1.3310489601
b4	22.4495660415	1.5862027886	1.3700185785
b5	43.4232369828	0.8667889255	0.7646361182

方程式 (6) 整個來講，是個相當複雜的非線性函數。它並沒有方程式 (3) 所需要的線性部份。所以，十次模擬中的最佳模型並不代表貨幣數量方程式。至於，其他九次的模擬又如何呢？從附錄 B.2 中，我們發現其他九次模擬所產生的最佳模型也沒有一個具有方程式 (3) 的線性部份。從這點來講，遺傳規畫在臺灣資料中，成功發掘貨幣數量學說的機率是零。所以不像美國資料，貨幣數量學說在臺灣的資料中並不具競爭力。這點發現也許不太令人意外。事實上，臺灣貨幣數量

成長率與通貨成長率之間一個不成比例的關係，早已受到學者普遍地注意，有關這點最近的研究，可參考 Yang & Shea (1994)。而遺傳規畫所發掘的只是再度將這個問題的面貌呈現出來。

伍、遺傳規畫是否會產生過度配適？

在上一節中，我們用美國與臺灣的資料，說明了遺傳規畫在發掘知識上所發動的假說競爭過程。透過大規模（在我們的例子是五十萬次）的演化尋找中，遺傳規畫找出這之間最有競爭力的模型。然而，在這整個過程上一個更基本的問題則是：到底遺傳規畫所發掘出來的模型究竟能不能算知識？譬如說，以美國的個案而言，遺傳規畫在模擬 a3、b1 與 b2 中，均發掘出貨幣數量方程式。但是從附錄 B.1 中，我們可以清楚地看見，除了 b2 以外，a3 與 b1 在理解上都有相當的困難。因此，在這種情況下，我們面臨到的問題是：這樣成堆而難以理解的數學方程式能不能算知識。這個問題不僅僅是遺傳規畫的問題，也是其他一些人工智慧自動尋找方法所同樣面臨的問題。這個問題也許並不是一個容易解答的問題，但是，解決的前提是我們必須正確了解它的本質。

這個問題的本質就是複雜。用 LISP 語言所表現的樹狀展開來講，樹的層級（或深度）就是一個複雜的概念。籠統地說，愈深的樹（或枝葉愈茂密的樹）所表現的就是愈複雜的模型。然而，在資訊理論中，複雜（或複雜度）並不是一個「繁瑣」或「冗長」的概念。相反地，它和經濟學中成本函數中的成本有一個非常相似的地方，亦即它們都是衡量在「最有效」的臨界邊（frontier）上。在經濟學中的成本是為生產某一單位的產量，所必須付出的最少投入（最有效的投入）。在程序資訊理論中（Algorithmic Information Theory），複雜度（或是程序複雜度（Algorithmic Complexity））則是說，某一物件（object）在萬能圖靈機器（Universal Turing Machine）展現下最短程式的長度（最有效程式的長

度)。⁷ 在這樣的定義下，複雜度的概念和我們日常用語的方式有些出入。最重要的，除非我們願意犧牲一點精確度或做「差不多推理」(Approximate Reasoning)，複雜度不是任意可以降低的。⁸ 因此，從這個角度來講，如果複雜度是造成我們理解上的障礙，那麼要增加理解度的代價就是要犧牲精確度。或者說，如果知識的前提必須是它能用語言、文字直覺來理解，而此理解又受到了複雜度的影響時，那麼精確的假說(或模型)就不太容易成為知識。

這樣一個認識是否能幫我們解決問題？以美國的例子來講，當我們很重視理解能力時，只有模擬 b2 所發掘的算是一個知識。但是，這並不代表當我們很重視精確度時，模擬 a3 所發掘的就能算是知識。因為這裡還牽涉到另外一個問題，即什麼是「精確」？以模擬 a3 而言，它的 SSE 最小，但是「過度精確」的背後，往往是統計學家所擔心的「過度學習」或「過度配適」(overfitting)。所謂過度配適，就是當我們用模型來解釋過去時，它可以解釋的天衣無縫，但是要用它來預測未來時，它的表現卻是很差。由於過度學習或過度配適所展現的另一面就是模型的過度複雜。因此統計學中對過度配適的處理基本上是將模型的複雜化視為一個要「懲罰」的對象。在實際操作上，則是建立一個懲罰函數 (penalty function)。而在目前文獻裡，幾乎所有的懲罰函數都可溯其基礎到資訊理論上。所以，由這些懲罰函數所構成的模型選擇判準又稱為「資訊判準」(Information Criterion)，如 AIC (Akaike Information Criterion), BIC (Bayesian Information Criterion), FIC (Fisher Information Criterion)。此外，MDLP (Minimum Description Length Principle) 也是種不折不扣的資訊判準。在真實模型存在的前提下，對這些資訊判準的大樣本研究已有很多。如 Scolve (1994) 與 Wei (1992)。但在小樣本的情況下又如何呢？Lauritzen (1995) 在比較這些準則時，曾下了這樣的結論：

No small sample results of any significance are known to me. (p.26)

除了小樣本是個限制外，截至目前為止，我們也不很清楚如何將資訊判準應用到遺傳規畫的架構中來，從本文的立場，這自是一個值得深入的研究方向。因此，在以下的討論中，我們仍沿用一般常用的「樣本內」(in-sample)「樣本外」(out-of-sample)的配適比較。亦即，遺傳規畫中所產生的複雜模型要成為知識之前，必須說明它的精確度絕非過度配適下的過度精確。也就是不僅它的 *SSE* 很低，而且在預測未來時，它的 *SSPE* (sum of squared prediction errors) 也應低得合理。依循 Koza (1992b) 的方式，我們將全部的資料分成兩部份，即樣本內與樣本外兩部份。樣本內的資料，是用來開啓演化的過程，而樣本外的資料，是用來測試演化的績效。在我們的實驗中，將120季的資料中的前80季作為樣本內的資料，亦即美國資料從1963: 1 到 1982: 4，臺灣資料從 1962: 1 到 1981: 4。而後40季則作為樣本外資料，美國資料從 1983: 1 到 1992: 4，臺灣資料從 1982: 1 到 1991: 4。

利用美國前80個季資料所產生的線性迴歸結果，列於表 9 中。至於遺傳規畫方面，我們則仿照前面的作法，根據表 4 與表 5 的參數分別對美國前80季的資料執行5次模擬，其結果列表 10 中。表 10 的第二欄是各次模擬在進化到第100代時，最佳模型的 *SSE*。第三欄則是該最佳模型在後樣本的 *SSPE*。根據 *SSE*，模擬 a4 所選擇的最佳模型是最具有競爭力的。其函數形式列於方程式 (7) 中。

$$\log(P) = \log(M2) - \log(GNP) + k(GNP, M2, IR) \quad (7)$$

其中

$$k(GNP, M2, IR) = \sin(2.72 + (((ir + (ir + \sin(-1.48))) * (-1.48)) / ((4.33 + \sin((gnp - (\sin(gnp / ((m2 * m2) / ((m2 * (-1.48)) + gnp))) - gnp)) / (((\sin(m2) / m2) * \sin(gnp)) - ((\cos(m2) - (ir * (gnp + m2))) + m2)))) * m2)))$$

表 9：線性迴歸模型估計的係數 (美國)：1963: 1 — 1982: 4

Model-1 參數估計				
變數	估計值	標準差	t-值	p-值
Intercept	0.635123	0.28049390	2.264	0.0264
$\log(GNP)$	-1.036667	0.04699530	-22.059	0.0001
$\log(M2)$	1.003368	0.01539021	65.195	0.0001
$\log(IR)$	0.078958	0.00686719	11.498	0.0001

$SSE = 0.01667$, $SSPE = 0.07574$, $R^2 = 0.9984$, $R^2 = 0.9983$, $DW = 0.72$

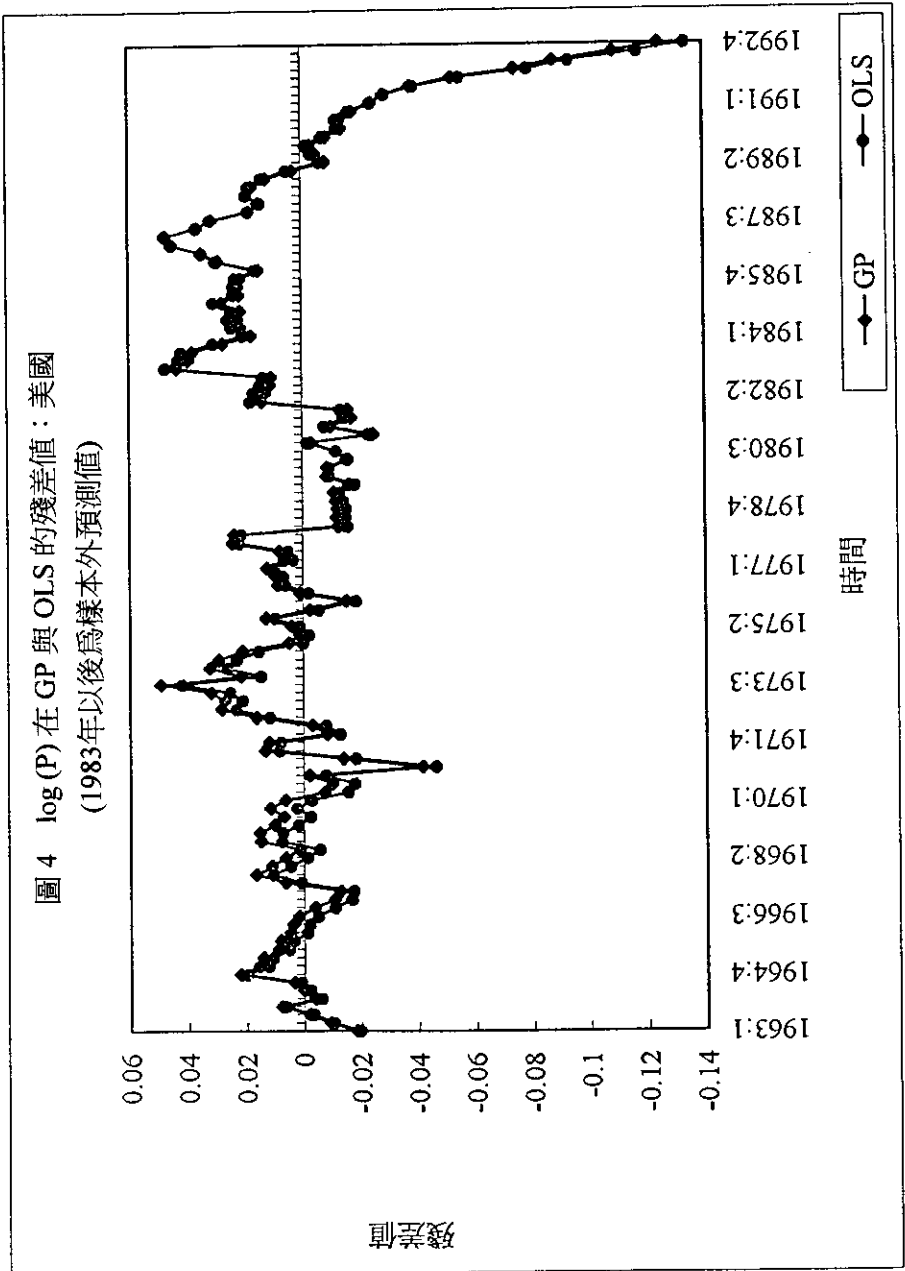
表 10：各次模擬在 Gen = 100 時的 SSE 及 SSPE (美國)：1963: 1 — 1982: 4

模擬	SSE	SSPE
a1	1.5999298828	1.7465597347
a2	1.6255382926	1.8543774187
a3	0.0717859184	2.1047432318
a4	0.0192405101	0.0672607563
a5	0.0284926297	0.0831830847
b1	0.1670751781	3.0964667583
b2	0.9946509059	15.3993457101
b3	0.0821156711	0.2780130295
b4	0.3717427121	2.2893105894
b5	0.0861279756	0.0349934493

表 11：線性迴歸模型估計的係數 (臺灣)：1962: 1 — 1981: 4

Model-1 參數估計				
變數	估計值	標準差	t-值	p-值
Intercept	4.479476	0.62668134	7.148	0.0001
$\log(GNP)$	-0.704005	0.08648681	-8.140	0.0001
$\log(M2)$	0.608745	0.03811986	15.969	0.0001
$\log(IR)$	0.325145	0.02979381	10.913	0.0001

$SSE = 0.15509$, $SSPE = 0.70262$, $R^2 = 0.9899$, $R^2 = 0.9895$, $DW = 0.895$



由於方程式 (7) 在線性的部份完全符合貨幣數量方程式 (3) 中的線性部份，因此和前面用120個觀測值的結果一樣，遺傳規畫再次在美國樣本內的資料中，發現了貨幣數量學說。至於，複雜的 $k(\cdot)$ 函數是否能視為一種「知識」，或是否有必要將流通速度視為變數而非常數，則取決於方程式 (7) 有無過度配適的嫌疑。我們將用方程式 (7) 來做樣本外40季的條件性預測，⁹ 並將此結果與線性迴歸模型相比。根據表 10 中的第三欄，我們可以看見，方程式 (7) 的 $SSPE$ 為 0.067，較線性迴歸的 0.075 為佳。¹⁰ 為了進一步了解兩種方法在條件預測上的差異，圖 4 中同時展現 GP 與 OLS 對 $\log(P)$ 的預測誤差。從圖 4 中，我們發現 GP 在 1983 年以後的 40 季中，其對 $\log(P)$ 的預測，有四分之三以上的時間都較 OLS 為佳。

另外，為了更嚴謹起見，我們也依照 Diebold and Mariano (1995) 中所提 S_2 統計量，在假設損失函數是絕對誤差的情況下，對 GP 與 OLS 預測的準確度進行近似檢定 (asymptotic test)。為了確定 S_2 統計量使用的方式，我們先對兩個預測的損失誤差進行 Box-Pierce X_2^2 檢定。結果並無法拒絕原始假設 ($X_{12}^2=6.227$)，亦即序列不相關的假設成立。然後，我們再進行 S_2 檢定。因為 $S_2 = 10$ ，所以在 0.05 型一誤差的雙尾檢定下，我們拒絕原始假設，而 GP 的預測準確度因而顯著優於 OLS。

所以，這點也代表了方程式中的 $k(\cdot)$ 並沒有過度配適 M2 的流通速度。因此，綜合前一節的結果下來，我們可以做這樣一個結論：遺傳規畫運用在美國的資料中，所發現的不僅是一個具競爭力的模型，而且也是個具競爭力的知識。這個模型 (或知識) 就是貨幣數量方程式 (學說)。它的流通速度，是隨著時間而變動的，並非常數。

同樣地，我們將相同的程序運用在臺灣的資料上。表 11 與表 12 分別是線性迴歸與遺傳規畫的結果。從表 12 「 SSE 」一欄中，我們可以看出最具競爭力的模型是由模擬 a5 所選出的最佳模型。其 SSE 值為

0.056，較線性迴歸模型的0.15509為低。而它的函數形式則列於方程式(8)。

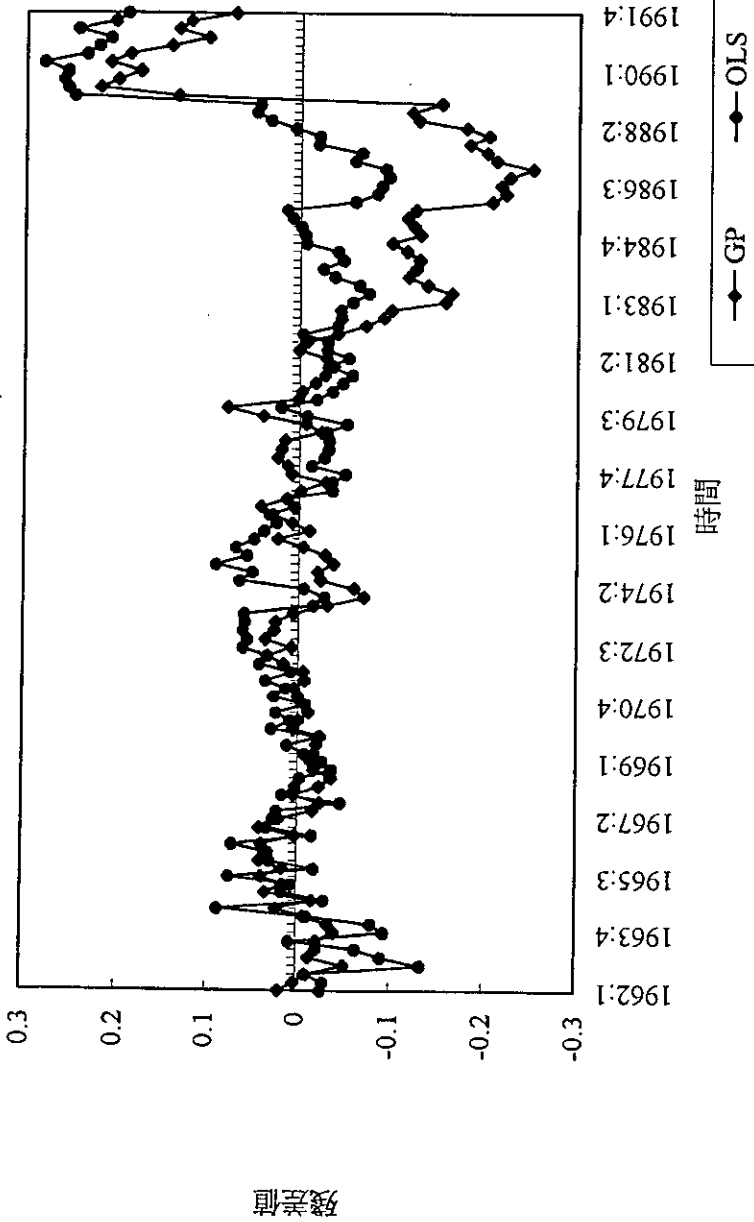
$$\log(P) = \log(((\sin(\sin(\log(\exp(\log(\exp(\sin(m2)) + \sin(\log(\exp(\log(\exp(\sin(m2))) + \sin(gnp))) * \log(((\log(m2) * gnp) * \log(\exp(m2)) + \exp(\exp(\log(\exp(\sin(m2)) + (\sin(\sin(gnp + \sin(m2))) + \sin(m2))) * \log((((gnp + \sin(m2))) * m2) * \log(\exp(\log(\exp(m2)) + \exp(1.72 + \sin(1.72)))))))))) - (-7.69)) + (m2 / \sin(\log(\log(\exp(\log(\exp(m2) + (((4.86 / \log(\exp(gnp + \sin(\sin(gnp + \sin(m2)))) + ir) + ir))))))))) \quad (8)$$

表 12：各次模擬在 Gen=100 時的 SSE 及 SSPE (臺灣)：1962:1 — 1981:4

模擬	SSE	SSPE
a1	0.1924329652	7.4097310516
a2	0.5120331812	5.3415378651
a3	0.1634411429	0.7357428760
a4	0.7196929084	0.6155675044
a5	0.0567330853	1.0089169895
b1	0.6988117015	31.8570062237
b2	0.6229769314	3.7542788341
b3	2.8936244352	4.2058834105
b4	0.3024963573	0.9325835964
b5	0.3027790330	3.9627926138

方程式(8)與前一節的方程式(6)有兩個相似點。第一、它們都是相當複雜的非線性函數。第二、它們都沒有方程式(3)所需要的線性部份，所以貨幣數量方程式在臺灣樣本內的資料中，並沒有被發掘到。除此之外，在其他九次模擬中，也沒有任何一次發掘到貨幣數量

圖 5 $\log(P)$ 在 GP 與 OLS 的殘差值：臺灣
(1982 年以後為樣本外預測值)



方程式。¹¹唯一比較接近的是模擬 b3。模擬 b3 所得到的是一個線性函數(方程式(9))，但是它並沒有方程式(3)線性部份所要的 m_2 。而且從表 12 中，也可以看出它的 SSE 是十次模擬中「特別」差的。所以，整體來講，不像美國，貨幣數量方程式在臺灣資料中，並沒有被遺傳規畫發掘到。

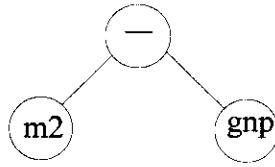
$$\log(P) = (gnp - 8.66) \quad (9)$$

至於模擬 a5 所發現的模擬能不能算知識，有沒有過度配適？從表 12 的 $SSPE$ 一欄中，我們可以看到模擬 a5 對樣本外 40 個季資料的 $SSPE$ 是 1.008，比線性迴歸模型的 0.7026 高出一些。¹² 爲了進一步了解這個差異，圖 5 中我們比較 GP 與 OLS 在預測 $\log(P)$ 上的差異。圖 5 可告訴我們從 1982 年第一季以後，GP 對 $\log(P)$ 的預測較之於 OLS 而言，一直都處於劣勢，直到 1989 第二季後的 13 個季，GP 才顯出它的優勢來。另外，我們也就 Diebold 與 Mariano 的 S_2 進行近似檢定。首先，Box-Pierce 的 X^2 檢定並無法拒絕序列不相關假定 ($X^2_{12} = 4.8799$)，所以就正常程序而言， S_2 是 29。在 0.05 型一誤差的機率下，我們拒絕原始假設。也就是方程式(8)所代表的 GP 預測是劣於 OLS 的預測。這點雖然不利於方程式(8)成爲一種知識，但是，由於十次模擬中的九次都傾向非線性的模型。而且，其中表現最佳的模擬 a3，其 SSE 優於 OLS，而且它的 $SSPE$ 與 OLS 相差無幾。所以，就「非線性」這點而言，遺傳規畫所發現的仍是有知識的特性。因此，我們的結論是遺傳規畫運用在臺灣資料中，所發現「非線性」現象，不僅是一個有競爭力的模型，也是個有競爭力的知識。這個發現使得貨幣數量學說在臺灣的資料不具競爭力。

六、結 論

將遺傳規畫應用在以資料庫爲導向的知識發掘裡，在社會科學中仍是一個相當新的研究工作。因此，在這個嘗試性的起步裡，本文藉

著具體的應用來試圖說明遺傳規畫如何可能成爲一個知識發掘的有力工具。而在這個說明裡，我們考慮了貨幣數量學說一種最簡單形式，即數量方程式。用 LISP 的樹狀結構而言，這個等於是在檢視遺傳規畫能否發現以下這個簡單的樹：



由於這棵樹的深度只有2，在遺傳規畫的植樹過程中，應是相當容易被創造出來的樹。這棵樹在美國的資料中出現了，而且能存活下來。而在臺灣的資料中也可能是出現了，但卻沒有存活下來。所以，這其間的差別就說明了貨幣數量方程式在美國資料中比在臺灣資料中較具競爭力。當然，這並不意謂著貨幣數量學說在臺灣資料中得不到支持。也許臺灣資料中存在著某種更複雜形式的貨幣數量學說。譬如說，具有「共整合」關係下的動態貨幣數量學說。由於共整合牽涉到落差元的運作，而要将落差元的運作融入遺傳規畫的程式設計中，尚有一些困難。若能克服這些困難，則遺傳規畫除了能發掘隱藏的非線性關係外，還可被用來發掘隱藏的動態結構。這將是本文未來繼續深入研究的方向。

註 釋

- 1 在計量經濟學正統的訓練中，這樣一個基本的工作，卻有一個惡名昭彰的名稱，那就是「資料濫墾」(data mining)。有關這點，可參考 Charemza 與 Deadman (1992)，第二章。

- 2 這只是從本文應用的觀點來思考「計算」，而並不是對「計算」的一個定義。有關對計算精確的定義，請參考 Davis (1978)。
- 3 LISP 是用在1956到1962年間由數學家 McCarthy 所創造。它與 Prolog (programming in Logic) 同是人工智慧中最常用到的兩種語言。有關 LISP 的歷史，可參考 McCarthy (1981)。
- 4 自遺傳程式裡，Rudolph (1994) 已經證明，在加入「保留精英」(elitist election) 的運作元後，用來代表 GA 馬可夫鏈中唯一的滯點 (Absorbing States) 就是全域性最適點。所以，不論原始條件為何，任何尋找途徑最終都能收斂到全域均衡。至於在收斂速度上的實証研究，請參考 Gelsema (1995)。
- 5 暫且不論遺傳規畫，就以其動態模型較遺傳規畫為簡單的遺傳程式而言，目前對參數值設定的理論依據，在研究成果上也非常有限。Shapiro, Prugel-Bennett and Rattray (1994) 是作者所僅知的文獻。在那篇文章中，他們用統計力學的方式探討了複製、交配與突變之參數值對配適度分配各階動差的影響。雖然，它們對最適複製參數值的研究有較一般的結果，但是其對交配與突變的研究仍只針對了特定的問題而並不具一般性。
- 6 有關這點一些例子，可參考陳樹衡、林祖嘉與葉佳炫 (1995)。
- 7 程序複雜度在程序資訊理論中，也稱為 Kolmogorov 複雜度 (Kolmogorov Complexity)。它是整個程序資訊理論中，最重要的概念。Cover 與 Thomas (1991) 一書中，對此有專章 (第七章) 介紹。
- 8 有關這點，也請參考 Martin (1995) 的討論。
- 9 詳細結果，請參見陳樹衡與葉佳炫 (1996)。
- 10 條件性預測即指在使用方程式 (7) 來對 $\log(P)$ 進行預測時，我們是在方程式 (7) 右邊的 $M2$ 、 GNP 與 k 均已為確知的情形下進行的。也就是，我們是將 $M2$ 、 GNP 與 k 的真實值代入式 (7)，來對 $\log(P)$ 進行預測。
- 11 但是依照 Hochberg-Tamhane 的檢定 (Hochberg and Tamhane (1987), Table 7)，在 $a1$ - $a5$ 與 OLS 六個模型一起檢定的結果，其差異並不具顯著性 (顯著性水準 0.05)。

12 但在統計上而言，這差異亦不顯著。

參考資料

陳樹衡、林祖嘉、葉佳炫

1995 〈空屋率的模型選擇及其穩定性：遺傳規畫的應用〉，《住宅學報》3: 73-98。

陳樹衡、葉佳炫

1996 〈貨幣數量學說的競爭性：遺傳規畫在知識發掘上的應用〉，Working Paper Series No. 9604，政治大學經濟系。

Angeline, P. J., G. M. Saunders and J. B. Pollack

1994 “An Evolutionary Algorithm that Construct Recurrent Neural Networks,” 5(1): 54-65.

Charezma, W. W. and D. F. Deadman

1992 *Nem New Directions in Econometric Practice: General to Specific Modelling, Cointegration and Vector Autoregression*. Edward Elgar.

Chen, Shu-Heng and Chia-Hsuan Yeh

1996a “Toward a Computable Approach to the Efficient Market Hypothesis: An Application of Genetic Programming,” Forthcoming in *Journal of Economic Dynamics and Control*.

1996b “Bridging the Gap Between Nonlinearity Tests and the Difficient Market Hypothesis by Genetic Programming,” in *Proceedings of the IEEE/IAFE 1996 Conference on Computational Intelligence for Financial Engineering*, IEEE Press (ISBN: 0-7803-3236-9), pp.34-40.

1996c “Genetic Programming and the Efficient Market Hypothesis,” in J. Koza et al. (eds.), *Proceedings of the 1996 International Conference on Genetic Programming*. Cambridge: MIT Press.

Chen, Shu-Heng, J. Duffy and C. H. Yeh

- 1996 "Genetic Programming in the Coordination Game with a Chaotic Best-Response Function," in P. Angeline, T. Back, and D. Fogel (eds.), *Proceeding of the Fifth Annual Conference on Evolutionary Programming EP'96*. Cambridge: MIT Press, MA.

Cover, T. M. and J. A. Thomas

- 1991 *Elements of Information Theory*. John Wiley and Sons.

Crevier, D.

- 1993 *AI: The Tumultuous History of the Search for Artificial Intelligence*. BasicBooks.

Davis, M.

- 1978 "What is a Computation?" in L. A. Steen (ed.), *Mathematics Today: Twelve Informal Essays*. Springer-Verlag.

Diebold, F. X. and R. S. Mariano

- 1995 "Comparing Predictive Accuracy," *Journal of Business and Economic Statistics* 13(3): 253-264.

Dworman, G., S. O. Kimbrough and J. D. Laing

- 1996 "Bargaining by Artificial Agents in Two Coalition Games: A Study in Genetic Programming for Electronic Commerce," in J. Koza et al. (eds.), *Proceedings of the 1996 International Conference on Genetic Programming*. Cambridge: MIT Press.

Fogel, L. J., A. J. Owens, and M. J. Walsh

- 1966 *Artificial Intelligence through Simulated Evolution*. John Wiley.

Gelsema, E. S.

- 1995 "Abductive Reasoning in Bayesian Belief Networks: Using a Genetic Algorithm," in D. Fisher (ed.), *Proceedings of the 5th International Workshop on Artificial Intelligence and Statistics*, pp.245-251.

Granger, C. W. J. and P. Newbold

- 1974 "Spurious Regressions in Econometrics," *Journal of Econometrics* 35: 143-159.

Hallman, J. J., R. D. Porter, and D. H. Small

- 1989 "M2 Per Unit of Potential GNP as an Anchor for the Price Level," *Board of Governors of the Federal Reserve System*. Washington, DC. Staff Study 157.

Hochberg, Y. and A. Tamhane

- 1987 *Multiple Comparison Procedures*. New York: Wiley & Sons.

Holland, John H

- 1975 *Adaptation in Natural and Artificial Systems*. University of Michigan Press.

Humphrey, T. M.

- 1989 "Precursors of the P-star Model," *Economic Review*, Federal Reserve Bank of Richmond, July-August, pp.3-9.

Koza, J. R.

- 1989 "Hierarchical Genetic Algorithms Operating on Populations of Computer Program," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*. Morgan Kaufmann.
- 1992a *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. The MIT Press.
- 1992b "A Genetic Approach to Econometric Modelling," in Bourguine, P. and B. Walliser (eds.), *Economics and Cognitive Science*, Pergamon Press.

Lauritzen, S.

- 1995 "Statistical Methods for Inducing Discrete Graphical Models from Data," AIS Tutorial in the *Fifth International Workshop on Artificial Intelligence and Statistics*.

Lubinsky, D.

- 1995 "Tree Structured Interpretable Regression," in D. Fisher (ed.), *Proceedings of the 5th International Workshop on Artificial Intelligence and Statistics*, pp.331-341.

Martin, J. K.

- 1995 "Evaluating and Comparing Classifiers: Complexity Measures," in D. Fisher (ed.), *Proceedings of the 5th International Workshop on Artificial Intelligence and Statistics*, pp.331-341.

McCarthy, J.

- 1981 "History of LISP," in R. Wexelblat (ed.), *History of Programming Languages*. Academic Press.

Rechenberg, I.

- 1965 "Cybernetic Solution Path of an Experimental Problem," *Royal Aircraft Establishment Translation No. 1122*, Ministry of Aviation, Royal Aircraft Establishment.

Rudolph, G.

- 1994 "Convergence Analysis of Canonical Genetic Algorithms," *IEEE Transactions on Neural Networks* 5(1): 96-101.

Sclove, S. L.

- 1994 "Small-sample and Large Sample Statistical Model Selection Criteria," in P. Chesseman and R. Oldford (eds.), *Selecting Models from Data: AI and Statistics IV*. Lecture Notes in Statistics 89, Springer-Verlag.

Shapiro, J., A. Prugel-Bennett and M. Rattray

- 1994 "A Statistical Mechanical Formulation of the Dynamics of Genetic Algorithms," in T. C. Fogarty (ed.), *Evolutionary Computing*. Springer-Verlag. pp.17-27.

Stark, W. R.

- 1989 *LISP, Lord and Logic*. Springer-Verlag.

Wei, C. Z.

1992 "On Predictive Least Squares Principles," *The Annals of Statistics*, 20(1): 1-42.

Yang, Y and J. Shea

1994 "Money and Prices in Taiwan in the 1980s," 開放總體經濟問題研討會，臺北：中央研究院經濟研究所主辦。

附 錄 A

A 遺傳規畫程式設計簡介

A.1 尋夫空間

遺傳規畫是一種隨機尋找程式 (random search algorithm)。要了解它程式執行的梗概，首先要了解的是它的尋找空間 (search space)。遺傳規畫的尋找空間是由程式 (programs) 所組成的空間。更明確的說，它是由 LISP 樹所組成的空間。爲了要了解這點，我們先要有一個簡單而基本的認識，即一個程式就是一棵 LISP 樹。爲了要建立這個認識，我們假設物價水準 (P) 是 M2 供給量與 GNP 的函數，且函數形式爲 $G(\cdot)$ ，

$$G(M2, GNP) = 1.65 * \left(\frac{M2}{GNP}\right) \quad (A.1)$$

如果我們將 G 這個函數寫成 LISP 的 S 表現 (LISP-S expression)，則它的形式如下：

$$G = (* 1.65 (% M2 GNP)) \quad (A.2)$$

若再進一步將 G 函數的 LISP-S 表現用有根有方向的分離樹 (a rooted, directed parse tree) 來表現，則可將其繪成如圖A.1的形式。

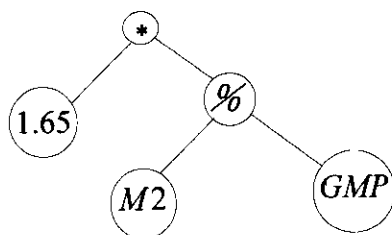


圖 A.1

所以，從方程式 (A.1), (A.2) 到圖 (A.1)，我們可以了解到為甚麼一個函數或一個程式就是一棵 LISP 樹(對這點更詳盡的介紹，可參考 Stark (1989))。

而由這些 LISP 樹所組成的空間就是遺傳規畫的尋找空間。任何一個有關物價決定的學說，若是能表現成一個函數，它就有一個 LISP-S 表現，也就有一個相對應的 LISP 樹表現。所以，凡是能被函數或程式所表現的物價學說或假設，也就自動成為 GP 尋找空間中的一員。

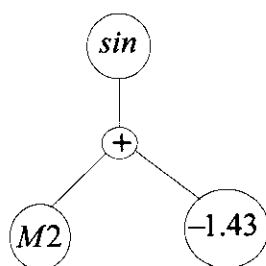


圖 A.2

但是，在實際應用上，並不是所有的 LISP 樹都可以出現在我們的尋找空間。要認識這點，我們必須再回到圖 A.1 中。在圖 A.1 中，我們

可以看到一棵 LISP 樹的組成有幾個部份。首先是所謂的函數部份，譬如說圖A.1中的 * 與 %。這些可以被允許出現在 LISP 樹中的函數的集合，就稱之為函數集合。以本文為例，我們使用的函數集合有兩類。兩者之差在於一者有 *sin* 與 *cos*，一者則沒有 (請參見與表 4 與表 5)。所以，以圖A.2所示的這棵樹而言，它可以被允許出現在表 4 的尋找空間中，但是卻不被允許出現在表 5 的尋找空間裡。因為在表 5 的尋找空間中，並沒有 *sin* 函數。當然，這裡所謂的不被允許是指不能以「直接」的形式出現。但是，一個函數如 *sin* 或 *cos* 都有它的多項式展開。所以，即使它的直接形式不允許，其間接形式 (展開逼近) 仍是有可能的。也因為此，函數集合中的函數也稱為初始函數 (initial function)。而其它的函數都是由這些初始函數所衍生而來的。

其次是所謂的常數部份，如圖A.1中的 1.65。常數部份通常是給定一個初始的浮點域。如在本文中，此一浮點域是設定在由 -9.99 到 9.99 之間所有到小數點兩位 (含兩位) 的有理數。至於其它的有理數可以再透過初始函數的運作而產生。例如，2 與 3 都在初始浮點域中，但是 $(\% 2 3) = \frac{2}{3}$ 就不在初始浮點域中，而是由初始浮點域所衍生出來的。

最後，是變數部份，如圖A.1中的 M2 與 GNP。變數部份可以允許使用 (include) 的變數也必須事先設定。如在本文中，我們設定的變數有 GNP，M2 與 IR。不在變數集合中的變數，自然不可能被用來組成 GP 尋找空間中的 LISP 樹。

一棵 LISP 樹便是由這些事先設定的初始函數，初始浮點域與變數所建構或衍生出來。由於這三者 (初始函數、初始浮點與變數) 是構成 LISP 樹的基本元素，所以它們又統稱之為原子 (atom)。另外，由於每一個 LISP 樹要有意義，必須滿足封閉的條件，也就是不能以函數集合中的元素做為 LISP 樹任一分枝展開的終點。例如圖A.3所示的就不是有意義的 LISP 樹。這樣的 LISP 樹自然不能納入我們的尋找空

間。換句話說，尋找空間的任何 LISP 樹都必須以常數或變數做為其分枝的終點。也正因為此，常數與變數集合又合稱之為終點集合。

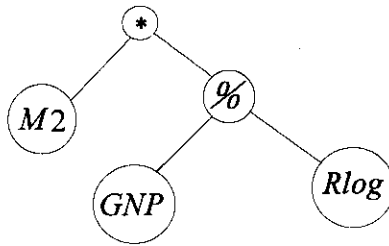


圖 A.3

A.2 尋找目標

一旦函數集合與終點集合設定，GP 的尋找空間便已確定。接下來的便是尋找的目標。尋找的目標通常視要解決的問題而定。以本文為例，我們的尋找目標是找一個模型，它的誤差平方和 (SSE) 為最小，亦即我們要找的對象是 G^* ：

$$G^* = \arg \min_{G \in \Omega(F, T)} SSE(G)$$

其中 $\Omega(F, T)$ 為由函數集合 F 及終點集何 T 所衍生的 LISP 樹的集合。換言之， Ω 就是尋找空間。而

$$SSE(G) = \sum_{i=1}^n (G(GNP_i, M2_i, IR_i, R) - P_i)^2$$

其中 $P_i, GNP_i, M2_i,$ 與 IR_i 分別代表樣本內 (in-sample) 中的第 i 個現象值。此處的 $P_i, GNP_i, M2_i,$ 與 IR_i 就是表 1 與表 2 所列的經濟資料。從 $SSE(G)$ 方程式中，我們可以看出這些經濟資料分成解釋變數與被解釋變數，透過 $SSE(G)$ 扮演著決定一個方程式良窳的標準。

A.3 尋找過程

在尋找空間與尋找對象皆已給定後，GP 所要做的便是對此一尋找對象發動一系列的隨機尋找過程。這套過程簡單來講是先是一個規模可能相當龐大的尋找空間 Ω 中，創造一個原始子空間 $\Sigma_0 (\Sigma_0 \subset \Omega)$ 。然後，再利用象徵達爾文進化論中的三個基本運作元集複製、交配與突變來移動 Σ_0 ，而產生一個序列的子空間 $\Sigma_1, \Sigma_2, \dots (\Sigma_i \subset \Omega, i \in N)$ 。對這一個過程，我們下面要分幾個步驟來說明。

首先是子空間的規模。子空間的規模可以是固定的，也可以是隨著一些條件而改變的。在目前，大多數 GP 的文獻中，子空間的規模都是固定的。本文也是如此。我們子空間的規模設定在 500 (請參見表 4 與表 5)。也就是說，我們每一次或每一代尋找的結果都是一個由 500 棵 LISP 樹所組成的森林。而這個規模固定的空間就是所有不同程式所想要競相爭奪的資源。

其次是原始子空間 Σ_0 的形成。 Σ_0 的形成是透過 Koza (1992a) 所謂的交叉對分法 ("ramped half-and-half" generative method) 所產生。這套方法也為相當多的 GP 文獻所使用。本文限於篇幅的考量，在此只做一個簡單的說明。這種方法是從 LISP 樹的根開始從函數集合與終點集合隨機選項來建立一棵樹。然而這棵樹不能任意結束，其必須滿足前面提到的封閉性。譬如說，如果我們一開始是隨機選到了終點集合中的 M2，那麼由於 M2 已經是終點集合，所以一棵 LISP 樹已經完成。但是，假設我們一開始是隨機選到了函數集合中的 *。則由於 * 函數後面必須接兩個原子 (atom) 才有意義，因此，這個植樹過程就必須繼續。在繼續的過程中，若是第一個原子抽到 1.35，第二個原子抽到 M2，那麼由於 1.35 及 M2 都是終點，所以此時，這棵樹便完成了 (圖 A.5)。但是若第二個原子抽到函數 +，則由於函數 + 並不能做終點，而且 + 必須後接兩個原子。因此，植樹的過程便還要繼續。如到圖 A.6 所示才能停止。在這樣的過程中，LISP 樹的分枝每往下一個層

次深度就加深一層。譬如說，以圖A.4, A.5與A.6所示的 LISP 樹而言，其深度分別是 1, 2, 3。



圖 A.4

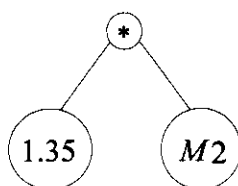


圖 A.5

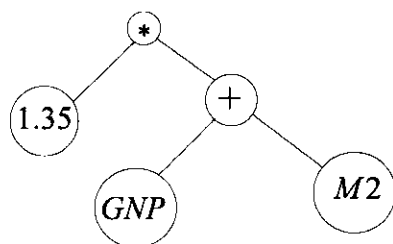


圖 A.6

至於，在植樹過程中，每一個出現的結點(即 LISP 樹中的圓圈)，其內容該如何決定則完全是隨機的。至於隨機的方式，基本上有兩種。第一種方法叫做成長法 (grow method)。成長法基本上是設定一個樹的深度 K ，然後在這個深度以內讓樹自由的成長。在這個情況下，當樹的深度還不到 K 時，任何出現的結點可以由函數來填補，也可以由常數或變數來填補。至於由何者來填補，則是以均勻分配的方式來決定。譬如說，在本文表 4 中，函數集合總共有 8 個元素，而終點集合共有 4 個元素，因此函數集合聯集終點集合一共有 12 個元素。所以，每

一個元素被選上的機率都是 $\frac{1}{12}$ 。可是當 LISP 樹的深度已經達到 K 時，則植樹過程必須要做一了結，所以此時終點集合只允許由終點集合中的元素依均勻分配的方式來填補。以本文為例，我們的終點有四個元素，所以每一個被選上的機率均為 $\frac{1}{4}$ 。

第二種方法則叫做完整法 (full method)。完整法是要要求 LISP 樹的深度一定要長到設定的深度 K 。深度沒有到 K 時，植樹的過程不能停止。基於這樣的限制，當樹的深度小於 K 時，所有的終點都必須由均勻分配的方式從函數集合中產生。也就是，在本文中每一個函數都有 $\frac{1}{8}$ 的機率被選上。而當 LISP 樹的深度到 K 時，植樹過程必須結束。所以此時就和前述的成長法一樣，所有在第 K 層的結點都是由均勻分配的方式從終點集合中產生。

在以上所描述的隨機選樣的過程中，若選上的是常數。則我們還要依照均勻分配的方式由初始浮點域 (即本文中的 $[-9.99, 9.99]$) 中去產生一個確定的數值。此外，每一個原子在進行隨機選樣時，彼此之間是獨立。而在 Σ_0 中，每一棵 LISP 樹的建立過程彼此之間也是獨立的。

至於交差對分法就是將成長法與完整法等比重地使用於給定的各種深度中。譬如說，在本文中我們依照 Koza，給定 2,3,4,5,6 五種等級的深度。而由於 Σ_0 的規模是 500。所以，在每一個深度中，用完整法創造的 LISP 樹與用成長法創造的 LISP 樹各為 50 棵 (請對照表 4)。

在初始空間 Σ_0 建立後，遺傳規畫接下來的工作就是要從 Σ_0 中去產生一個規模相等的空間 $\Sigma_1(\Sigma_0 \rightarrow \Sigma_1)$ ，然後再從 Σ_1 去產生 $\Sigma_2(\Sigma_1 \rightarrow \Sigma_2)$ ，依此類推，循序漸進。然而在任何一個 $\Sigma_{t+1}(t \geq 0)$ 產生之前，我們先要對 Σ_t 中的所有的 LISP 樹進行一個配適度 (fitness)

的衡量。直覺上來講，也就是我們要知道那些是「好樹」，那些是「壞樹」。至於「好」「壞」的標準，則自然隨著問題的性質或尋找的對象而不一。以本文為例，我們在前面已經提到，我們尋找的對象是一個能使誤差平方和 (*SSE*) 最小的樹。所以，在好壞認定的標準上，一個很自然的選擇就是 *SSE* (請參見與表 4 表 5)。因此，在 Σ_{i+1} 產生之前，我們必須先將 Σ_i 中所有 LISP 樹的 *SSE* 值算出來。為了討論方便，我們令 SSE_{ij} 為 Σ_i 中第 j 個樹的 *SSE*。

在 $SSE_{ij} (j = 1, 2, \dots, 500)$ 決定後，遺傳規畫下面要進行的工作就是根據 $\{SSE_{i,j}\}_{j=1}^{500}$ 對 Σ_i 去形成一個機率分配 $\{p_{ij}\}_{j=1}^{500}$ ，其中 $0 \leq p_{ij} \leq 1 (\forall j)$ 且 $\sum_{j=1}^{500} p_{ij} = 1$ 。這個機率分配不僅在 Σ_i 到 Σ_{i+1} 的過程中扮演很重要的角色，而且也是遺傳規畫能不能發揮其適當功能的一個關鍵因素，這點我們在後面還會提到。

$\{p_{ij}\}_{j=1}^{500}$ 的形式基本上是在考慮到了 LISP 樹的相對優越性後所給予它們的一個適當的權數。對於比較好的 LISP 樹而言，我們賦與其較高的權數 (機率值)，對於較差的 LISP 樹而言，我們則賦予其較低的權數。在不違反這個規則下，有很多種方式可以將 $\{SSE_{i,j}\}_{j=1}^{500}$ 轉換到 $\{p_{ij}\}_{j=1}^{500}$ 。在本文，我們使用 Koza (1992a) 在 "Symbolic Regression" 一節中所使用的轉換機制，即

$$p_{i,j} = \frac{a_{ij}}{\sum_{j=1}^{500} a_{ij}},$$

$$\text{其中 } a_{ij} = \frac{1}{1 + SSE_{i,j}}, j = 1, \dots, 500.$$

在文獻上，我們將 $SSE_{i,j}$ 稱為原始適合性 (raw fitness)，而將 a_{ij} 稱為調節適合性 (adjusted fitness)，並將 $p_{i,j}$ 稱為正規適合性 (normalized fitness)。

在 $\{p_{ij}\}_{j=1}^{500}$ 一旦形成後，遺傳規畫接下來就是要分三個步驟來進行 Σ_{i+1} 的建構。這三個步驟分別是由三個運作元所進行，即複製、交配及突變。這三個運作元將各自分擔式地創造出 Σ_{i+1} 中的一部份。這個分擔的比例是外生設定的。在本研究中，我們使用的分擔比例是由複製運作元負責10% (即50棵 LISP 樹)，交配運作元負責70% (即300棵 LISP 樹)，突變運作元則分擔10% (50棵 LISP 樹)，而另外剩下的10%是由所謂的「新生嬰兒」(new birth) 運作元來完成，新生嬰兒運作元也稱為「移民」(immigrants) 運作元，它在遺傳規畫中扮演的角色，相對於複製、交配與突變而言，是屬於比較次要、輔助性的。它的功能主要是在保持 $\Sigma_{i+1} (i \geq 0)$ 中 LISP 樹的歧異度 (diversity)，避免產生所謂的早熟型的收斂 (premature convergence)。

以下我們就分別對這幾個運作元的運作方式做一簡單的說明。首先是複製運作元。複製就是將 Σ_i 中的某一棵 LISP 樹原封不動的拷貝到 Σ_{i+1} 。所以，每一次複製都要先在 Σ_i 中選一棵 LISP 樹，而這選取的標準就是我們前面所提到的機率分配 $\{p_{ij}\}_{j=1}^{500}$ 。也就是 Σ_i 中的每一棵 LISP 樹 j 都有一個相對應的機率 p_{ij} 被選來用做複製的正本。複製要重覆的進行直到完成它應負擔的配額。以本文為例，複製需要進行50次，然而每一次複製的隨機取樣都是獨立的。

當複製完成後，我們的下一個工作就是交配。交配可以視為遺傳規畫中最核心的一個運作元。交配運作的方式是先從 Σ_i 中依據 $\{p_{ij}\}_{j=1}^{500}$ 獨立地進行兩次選樣，每一次選樣都會產生一棵 LISP 樹來，所以每一次交配運作都會先產生兩個 LISP 樹來。這兩棵 LISP 樹便為稱爲「配偶」或「父母」。以本文為例而言，假設在某次交配運作中，第一次選樣的結果方程式 (A.3)，

$$p = 3.18 + M2 (M2 * IR) \tag{A.3}$$

而第二次選樣的結果是方程式(A.4)，

$$p = 0.86 IR + \ln \left| \frac{M2}{GNP} \right| \tag{A.4}$$

則這兩條方程式寫成 LISP 程式後分別是方程式 (A.5) 及 (A.6)。

$$(+ 3.18 (* (M2 (* M2 IR)))) \tag{A.5}$$

$$(+ (* 0.86 IR) (Rlog (% M2 GNP))) \tag{A.6}$$

再將這兩棵 LISP 程式編成 LISP 樹便如圖A.7。

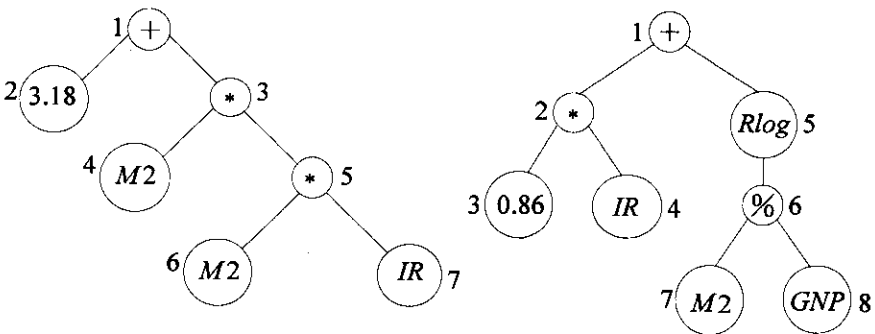


圖 A.7

圖A.7中的兩棵 LISP 樹便是父母。父母在 GP 中的責任，便是透過交配行為而產生下一代。從交配到產生下一代的過程可以分幾個步驟來說明。第一個步驟是選擇交配點 (crossover point)。這個步驟是在一棵 LISP 樹上指定一個節點，用這個節點做根，從原來的 LISP 樹中創造出一棵分離的子樹 (subtree)。這個被選定的結點便是交配點。以圖A.7為例，假設在左邊的這棵樹，我們選上了結點3做為交配點，而在右邊我們選上了點5做為交配點，則以上所說的兩棵子樹便如圖A.8所示。

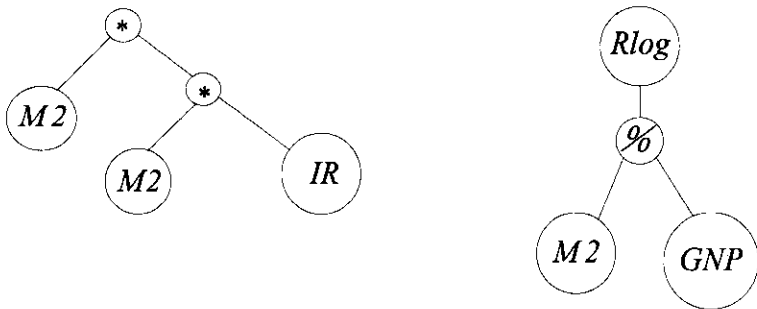


圖 A.8

至於交配點的決定也是隨機的。這個隨機的方式是先以白努力實驗 (機率1/2) 來決定交配點是終點 (常數或變數) 還是函數。一旦這個結果出來後，再進一步依據均勻分配來決定由那一個終點 (或函數點) 來做交配點。在以圖 7 為例，圖 7 右邊的 LISP 樹一共有四個終點 (即點 2,4,6,7)，另有三個函數 (即點 1,3,5)。若白努力實驗的結果是由終點做交配點，則點 2,4,6,7 將各有1/4的機率被選為交配點。若白努力實驗的結果是由函數點做為交配點，則點1,3,5將會有1/3的機率被選為交配點。

一旦交配點產生後，交配的下一個步驟便是產生下一代。產生下一代的方式便是將由交配點所決定的兩棵子樹互換至配偶子樹分離的位置。以圖A.7為例，左邊的 LISP 樹選定點3做交配點，而右邊的樹選擇點5做交配點。所以左樹中由點3所創造的子樹 (即圖A.8的左子樹) 將移接至右樹中點5以下的位置，而右樹中由點5所創造的子樹 (圖A.8的右子樹) 將移植到左樹中點3以下的位置。如此而形成的兩棵樹將如圖A.9所示。

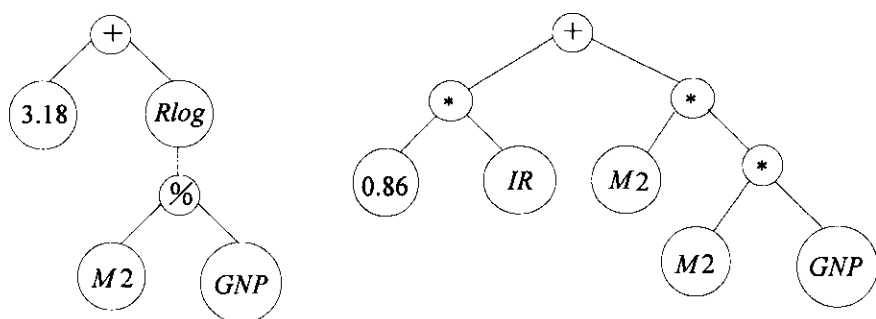


圖 9

圖A.9中的兩個子樹寫成代數形式，便是方程式 (A.7) 與 (A.8)

$$p = 3.18 + \ln \left| \frac{M2}{GNP} \right| \quad (A.7)$$

$$p = 0.86 IR + M2 (M2 * GNP) \quad (A.8)$$

而寫成 LISP 形式，便是方程式 (A.9) 與 (A.10)。

$$(+ 3.18 (Rlog (\% M2 GNP))) \quad (A.9)$$

$$(+ (* 0.86 IR) (* M2 (* M2 IR))) \quad (A.10)$$

當然，我們對樹的深度沒有任何限制，則在交配元運作不斷的運作下，樹有可能會愈長愈大(深)。太多太深的樹對整個 GP 的運算速度有相當負面的影響。因此，在本文，我們仿照 Koza 的做法將樹中的深度上限定為17(請參見表 4與表 5)。如果在交配的結果中，有任何一棵樹，其深度超過17，則我們將以1/2的機率從其父母中選擇一個做為替代，若兩個都超過17，我們將以父母做為替代。

如此，一次交配的運作便已完成，而一次交配運作的結果便是誕生兩棵 LISP 樹。這兩棵的 LISP 樹便加入到 Σ_{i+1} 中。和複製一樣的，交配必須反覆運作到完成其應用的配額。以本文為例，由於交配運作元一共要負責產生300棵樹，所以本文要重覆執行交配150次。然而每一次的隨機選樣(父母)都是獨立進行的。

交配之後的工作便是突變。突變的進行是先從 Σ_i 中依 $\{p_{ij}\}_{j=1}^{500}$ 機率分配隨機選樣產生一棵樹。然後，這棵樹上的任一結點都有一固定的機率會產生改變。在本文中，此一機率設為0.2(請參見表 4與表 5)。當一終點要變化時，我們將從終點集合中按照均勻分配的方式產生一個變數或常數來替代原有之變數或常數。至於函數點，我們則分為兩種情形來處理。其中 $\{+, -, *, \%\}$ 中的任何一函數可以被該一集合中的任一元素依均勻分配的方式所替代。至於 $(\sin, \cos, \exp, Rlog)$ 中的任何一函數也可以被該集合中的任一元素依均勻分配的方式所替代。但是，這兩個集合中的元素不可以彼此去替代另一個集合的元素。這樣子做的目的也是只在保護程式不失去其數學意義。至於，同一棵樹每一個結點其發生突變的機率在彼此之間是互相獨立的(mutually independent)。像複製與交配一樣，突變也一直要進行到它完成的配額為止。在本文中，突變要獨立的進行50次。

以上我們所介紹的三個遺傳運作元是遺傳規畫的主要運作元。它們一起完成了 $\sum_{i+1}(i \geq 0)$ 中450棵 LISP 樹的產生。在這三種運作元之外，在遺傳規畫中有時為了防止早熟型的收斂 (premature convergence) 的產生，通常還有些輔助型的運作元。其中，「新生嬰兒」(new birth) 便是一種常用的運作元。而本文也使用此一種運作元。新生嬰兒的運作方式是在 $\sum_i \rightarrow \sum_{i+1}$ 中，利用前面所談的成長法 (grow method) 在以深度不超過上限 (在本文中，此一上限設為17) 的前提下，創造一個「新」樹。這個樹稱為「新樹」是因為它的產生完全是由創造而來，而不是從 \sum_i 中衍生而來。同樣地，新生嬰兒運作元要反覆獨立地執行，直到完成其承擔的配額。在本文中，此一配額設為50 (請參見表 4與表 5)。

透過上述的四個運作元，整個序列 $\sum_0, \sum_1, \sum_2, \dots$ 便得以一個接著一個地產生。如果有一個最佳模型是 G^* ，我們希望的是存在一個 m ，使得 $G^* \in \sum_i (i \geq m)$ 。當然，如果我們不做任何額外的限制，這個尋找過程可能不會結束。所以，在本文中，如果遇到以下兩種情況，尋找過程就告結束。第一種情況是，如果存在 $G (G \in \sum_i)$ 且 $SSE(G) = 0$ ，則尋找過程便在第 i 代結束。第二種情況是，設定一個演化代數的上限，如在本文中，這個代數上限設為100 (參見表 4與表 5)。

以上便是我們對整個遺傳規畫運作過程的說明。對經濟學家而言，遺傳規畫仍是一個相當陌生的工具。限於篇幅，本文的說明可能仍然不夠詳盡。因此，我們除了推薦 Koza (1992a) (特別是第六章) 作為進階外，也提供以下幾篇目前 GP 在經濟學上僅有的應用作為參考：Chen and Yeh (1996a, 1996b, 1996c), Chen, Duffy and Yeh (1996), Dworman, Kimbrough and Laing (1996)。

附 錄 B

B 遺傳規畫模擬結果

B.1 在第100代表現最佳的個體：美國資料(1963: 1 — 1992: 4)

模擬 a1 :

$$\log(P)=\sin(((\text{gnp}-(-9.67))-m2)+\sin.\sin.\sin(((m2-9.67)-\sin.\log \\ (((\text{gnp}-(-9.67))-m2)+\text{gnp})+(-0.77-((\text{gnp}-m2)+\log(((m2-\text{gnp})-m2) \\ -(-9.67)))))))+\cos.\log((m2-(((m2-(-9.67))-m2)+\sin((\text{gnp}-(-9.67))* \\ 9.67)))/\text{gnp})))$$

模擬 a2 :

$$\log(P)=((\cos(-7.82*6.29)+\sin(m2-\text{gnp}))*\log(\log(\text{gnp}*((m2-\text{gnp}) \\ +(\text{gnp}*(m2*\exp(\cos.\cos(m2*6.29)))))))+(m2*(m2*\exp.\cos(6.29*m2))) \\))*\exp.\cos((\log(\cos(-7.82*-7.82)-m2)+(\cos(-7.82*6.29)+\sin(\\ 6.29-\text{gnp}))))+m2)))$$

模擬 a3 :

$$\log(P)=((m2-\text{gnp})+(\text{gnp}/((((\text{gnp}/((m2-\text{gnp})-\text{gnp}))+\text{gnp})-(-0.56))+ \\ ((\text{gnp}+(\text{ir}*(((\sin(m2/\text{gnp})+\text{gnp})*\sin(-9.54)))-(\sin(m2/((\text{gnp}*(-0.56- \\ \text{gnp}))*(\text{gnp}/\text{gnp}))))+\text{gnp}))))*\sin((\text{ir}/\text{gnp})*(\text{gnp}-m2))))-(-9.54))))$$

模擬 a4 :

$$\log(P)=((8.23-m2)/(m2/(-3.37)))$$

模擬 a5 :

$$\log(P)=\cos((\text{gnp}-\cos(\text{gnp}-\cos((\text{gnp}-\cos(3.47/((m2/3.47)))/(\text{gnp}/\cos.\cos((\text{gnp}-\text{gnp})+(3.47-(-4.25+m2))))))))+(3.47-(\text{gnp}+m2)))$$

模擬 b1 :

$$\log(P)=(((m2-\text{gnp})+((-0.66-(((\text{gnp}-(((0.39-(-0.66)))-(((m2-\text{gnp})+(m2-\text{gnp}))(-0.39))+(-0.66/(-5.69)))-(-0.39)))/m2)-ir))*(-0.39)/(-5.69))/(-5.69))-(-0.66-(-0.39)))$$

模擬 b2 :

$$\log(P)=((m2-\text{gnp})-(3.77/(-7.43)))$$

模擬 b3 :

$$\log(P)=(((\exp 6.80+((((((\exp(\text{gnp})/(m2-9.63))-\text{gnp})/((m2/\log.\exp(\text{gnp}/ir))+ir*(((\text{gnp}/ir)+(\exp(m2)/(gnp-\text{gnp}))/ir)/(m2-\text{gnp})))))-m2-\text{gnp})/m2)+\log(\text{gnp}))+(((\exp(m2)-\text{gnp})/(m2-\text{gnp}))/((m2/ir)+(\exp(\text{gnp})/(\exp m2-\text{gnp}))))+((m2-9.63)/((((\text{gnp}-\text{gnp})/((ir/\log((m2-6.80)/(ir-m2))))+(\exp(\text{gnp})/(m2-\text{gnp})))))-\text{gnp})-\text{gnp})+(\exp(m2/ir)/(gnp-(gnp/ir)))))))/(\exp(\text{gnp})/(m2-9.63)))$$

模擬 b4 :

$$\log(P)=\log(\log((m2+m2)-\log(m2/(5.62*(m2-\text{gnp})))))/((ir+(((5.62*(m2-\text{gnp}))/((m2/m2)*(m2/(5.62*(ir-((-0.64+m2)+m2)-\log((m2+ir)/(m2*$$

$$5.62)))))))/((m2/((m2-gnp)+\log((ir/(m2/m2))-m2))) * m2)-m2)) * (m2 - gnp)))$$

模擬 b5 :

$$\log(P) = (((m2 + (-8.39)) / (m2 + (((-9.22 - (\log(((\log(m2) - 9.36) / -6.19) / (gnp - m2)) / (gnp + (-8.39))) * gnp)) + (-8.39)) / gnp))) * \log(\log.\log.\exp(-9.22 - (((\log.\log(m2) - 9.36) / m2) / ((gnp - 6.19) / (m2 + (-8.39)))) * m2)) - gnp))$$

B.2 在第100代表現最佳的個體：臺灣資料 (1962: 1 — 1991: 4)

模擬 a1 :

$$\log(P) = (\exp.\sin.\sin.\sin.\sin.\exp.\log(gnp) + 2.83)$$

模擬 a2 :

$$\log(P) = ((\exp.\sin.\sin.\sin.\sin.\sin.\sin.\sin.\sin(6.73) / \sin.\sin.\sin.\sin.\sin.\sin.\sin(6.73)) + \sin.\sin(gnp))$$

模擬 a3 :

$$\log(P) = \log(m2 * ((-2.49 - \log(((m2 - ir) * (((\exp(\cos m2 - ((m2 - gnp) * ((ir - 3.86) / (-2.49)))) * (-2.49)) - 3.86) / ir)) + \log(((-2.49 + 3.86) - 3.86) + \log(((-2.49 + 3.86) * (-2.49) + m2)))))) + (m2 - ir))$$

模擬 a4 :

$$\log(P) = \log((\cos(gnp - ir) + ir) * (\log((m2 - \cos((\log(gnp - (-5.99 / \log(($$

$$\cos(m2)+ir)*(\log(ir)*m2))))*m2)-5.76))-(\cos(((gnp+(5.76-(\cos(ir-m2)+5.76-(m2/\sin(ir)))))/\log(gnp))*ir)-(\cos((ir-gnp)-m2)+ir))*m2))$$

模擬 a5 :

$$\begin{aligned} \log(P)=&(\log(((gnp+((\log(((((-7.19-gnp)-(((m2-gnp)-(gnp+gnp))-gnp)* \\ & ir))/gnp)-ir)-(m2*m2))+gnp)-\log((((gnp+gnp)-m2)-((m2-gnp)* \\ & (((gnp+ir)+ir)-(m2-m2))))/gnp)-ir))+gnp)-(m2*m2))-exp.\log.\sin. \\ & \log((m2-gnp)-(((m2-\cos(m2+(m2-(m2-m2))))-\cos(m2+(m2-(m2-(-4.25) \\ &)))*((-4.25+((gnp+gnp)-(gnp*m2)))-((((m2+ir)+ir)-(m2-gnp))*m2)))))) \end{aligned}$$

模擬 b1 :

$$\begin{aligned} \log(P)=&(((ir*ir)*(2.82+(-2.66)))*(((m2/ir)-(ir/(m2*(m2/ir))))- \\ & (((m2/ir)-((ir*m2)/(((m2/ir)-((m2*(m2/ir))/(m2*m2))))-(2.82/(\\ & ((m2/ir)-(((m2/ir)-((ir+ir)/ir))/(ir*m2))))*(m2/ir)-((ir*4.83 \\ & -ir/(-2.66))/(ir*ir)))))))/m2))) \end{aligned}$$

模擬 b2 :

$$\begin{aligned} \log(P)=&\log((\log((ir*m2)/(5.33/((m2-\log(gnp/(\log(((gnp/gnp)/gnp)*m2)/ \\ & ((\log(ir*ir)-m2)-\log((1.38/gnp)*\log(m2)))))))-\log(5.33/(m2-5.78)) \\ &))) * m2)/(5.33/(m2-5.33))) \end{aligned}$$

模擬 b3 :

$$\log(P)=(\log.\log.\log.\exp.\log.\exp(M2)*(\log.\log.\log.\exp.\log.\exp.\exp.\log(m2))$$

(1.80*log(gnp)))

模擬 b4 :

$$\log(P)=(((m2/gnp)/(gnp/m2))+(m2/4.33))$$

模擬 b5 :

$$\begin{aligned} \log(P)=\log(&(((gnp+\log((gnp*(((gnp+5.75)-(m2+m2))+gnp)*((-4.83 \\ &+((gnp-m2)+\log(m2))))+(((-4.83+gnp)-(m2+m2))+gnp)+\log.\exp \\ &\log(m2)))))+m2))-(m2+m2))*(\text{ir}*(-4.83+((gnp-m2)+\text{ir}))))+m2) \end{aligned}$$