

植基於粒子群演算法之大專院校派課系統

游峰碩
崑山科技大學資訊管理系
助理教授
fsyu@mail.ksu.edu.tw

王子夏
崑山科技大學資訊管理系
研究生
u9011023@mail.ndhu.edu.tw

吳昀珊
崑山科技大學資訊管理系
研究生
S096002353@stumail.ksu.edu.tw

摘要

本研究運用粒子群演算法(Particle Swarm Optimization, PSO)探討大專院校教師派課問題 (Faculty-Course Assignment Problem, FCAP)。教師派課問題不同於一般文獻上所探討的排課問題(Timetabling Problem, TP)。所謂的排課問題是指如何將下學期所欲開設之課程根據不同的資源限制指派至不同的時段，重點在於各項資源與時段之排程。而本研究探討的派課問題是指一個科系如何將下學期所欲開設之課程分派給系上之授課老師；研究的重點著重在派課時如何兼顧到「教」與「學」兩方面：在「教」的方面，研究重點在於派課的公平性、教師對於派課結果的滿意度；在「學」的方面，研究重點在於如何將適才適所的教師指派至相關課程以提升學生對於教學的滿意度。

關鍵詞：粒子群演算法、派課問題。

Abstract

In this research, we study the faculty course assignment problem (FCAP) with Particle Swarm Optimization (PSO). The FCAP is different from the timetabling problem (TP) discussed in the literature. The timetabling problems deal with how to assign courses in a semester to certain time-slots satisfying different resources constraints, and the faculty-course assignment problems deal with how to assign course to teachers regarding to teachers' preference, the fair distribution of teaching load and teachers' qualifications.

From literature, FCAP is always been considered as constraints in TP study or not at all, only a few papers study it independently. In this research, we apply genetic algorithm to search the optimal solution for FCAP. Using the study of timetabling problem as a basis, and focusing on the robustness and global search mechanism

provided by PSO as the research focus. The goal is to maximize the course assignment and teaching assignment satisfactory. Traditionally, genetic coding constantly involves repair mechanism after certain genetic operation which greatly degrades the performance of genetic algorithm. In this study, we propose a coding schema to prevent it so as to enhance the performance of the process.

Keywords: particle swarm optimization, course assignment.

1. 前言

大專院校之教師派課問題可視為一般排課問題的特殊例子，但又不同於一般文獻上所探討的排課問題。所謂排課問題針對如何將下學期所欲開授之課程根據目前系所的資源限制指派至不同的時段，其重點在於各項資源與時段之排程，而本研究探討之派課問題是指一個科系如何將下學期所欲開授之課程分派給系上之授課老師。現今大專院校之派課方式大部分皆為人工派課，在作業上不免因為個人因素而對整體派課結果產生影響，採用人工派課不僅費時費力，派課結果的不公平性也可能導致組織內部的不和諧。另外，不同院校間的派課規定彼此不同，甚至同校中不同科系的派課規定也不盡相同，其派課規定可能根據教學需求或是體制上的規範而有所改變。而在排課的過程中，對於派定的課表一但有任何教師因為某些因素無法履行授課責任而需調整課表時，常會有牽一髮而動全局的困擾。基於上述種種理由，我們可以看出派課方式之缺乏效率以及彈性。

本研究希望以粒子群演算法為基礎，建構一大專院校派課系統，此系統可以根據教師對於課程的喜好度及教師的專長產生出派課結果，讓有長才的教師能夠被分配到適任的課目，以提升學生的學習品質。本系統更可以針對不同院校間對於派課的需求做調整，如部分校系希望作業時滿足教師對於派課結果的滿意度，派課方向注重教師對於課目的喜好度，

以期提升教師教學的熱忱，而部分校系希望派課結果中各個科目的授課教師皆有其科目之專長或是相關專業認證，以維持整體的教學品質，及提高學生對於授課師資的滿意程度。使用者可以利用此大專院校派課系統，依照不同的派課需求做出以教師喜好或是教師專長為導向之調整，亦可產生兼顧兩者的派課結果。

以下各章節內容摘要分述如下。第二節對粒子群演算法與派課問題做相關的文獻探討。第三節描述本研究之派課問題並且建立所需要之數學模型。第四節描述本研究的研究步驟及權重值之設定。第五節對本研究之實驗結果做一歸納與結論。

2. 文獻探討

2.1 粒子群演算法

粒子群演算法由 Kennedy 和 Eberhart[7] 於 1995 年所提出，其主要概念源自於鳥類與魚群之群體行為的觀察。在其最佳化之過程中，每單一粒子會根據其本身的經驗以及整個粒子群之經驗作移動，使粒子能朝全域最佳解之路徑移動，意指每個粒子皆知道群體中哪個粒子離最佳解最近，也知道該粒子之位置，而粒子群之群體行為會驅使所有粒子跟隨離最佳解最近的粒子作移動。粒子群演算法初期隨機產生群體與計算適應值的步驟與基因演算法相當類似，但是兩者不同之處在於粒子群演算法後期會利用粒子間的群體行為與競合關係取代基因演算法的遺傳運算元(genetic operators) [10]。粒子群演算法之運作流程如下圖所示。

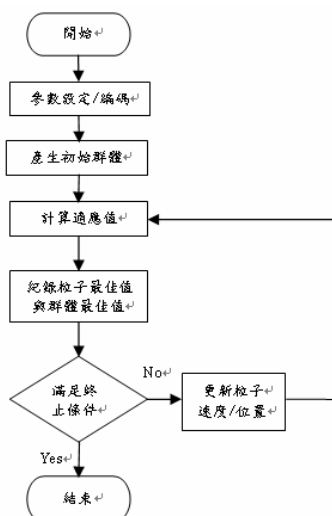


圖 1 粒子群演算法運作流程

由於許多的最佳化問題皆為離散型空間，如部份非連續解的排程問題等，而傳統的粒子群演算法只能針對問題解為連續實數的研究範疇，於是 Kennedy 和 Eberhart[6]於 1997 年提出離散式粒子群演算法(Discrete Particle Swarm Optimization, DPSO)，用以解決非連續性的最佳化問題，證實離散式粒子群演算法於 De Jong[4]之五項最佳化測試方程式中獲得良好的成效。過往亦有數位學者針對此一演算法進行改善，Yang 等學者[10]利用不同的方法更新速率，而提升演算法之執行效率。Al-kazemi 和 Mohan[3]在速率更新部份使用個體最佳值與群體最佳值擇一的方式，代替原本兩者同時使用之方式。

2.2 派課問題

在過去的文獻中，派課問題鮮少被獨立出來研究。Hultberg[5]從古典的運輸問題(transportation problem)觀點來研究教師的派課問題，並且將問題表示成混合整數規劃問題(mixed integer program)。對於派課問題，Hultberg 只考量了一個單純的條件：如何將指派給教師之不同科目的數量最小化。Hultberg 的研究將派課問題轉化成基數分割(cardinality partitioning)的極大化問題，並且藉由此模式來證明派課問題是一個強 NP-困難(NP-hard in the strong sense)的問題。文中亦使用一個利用分支限界(branch-and-bound)的演算法來求解其派課模型。

另一方面，有些學校的派課問題牽涉到教師與管理者，因此，從教師以及管理者兩個不同使用者的角度來考量，Ozdemir 等學者[6]使用 0-1 非線性多目標規劃(0-1 nonlinear multiobjective programming)求解派課問題的最佳化。在派課問題中，Ozdemir 的作法是把每位教師的需求當成是一項需要滿足的目標，並以此作為問題的限制式；在問題上，不僅需要考量到各位教師們的需求，也必須考量到管理者的需求。Ozdemir 採用層級分析法(Analytic Hierarchy Process, AHP)來綜合簡化問題中之多目標函數成為單一函數，並且有效地求得其問題的最佳解。Parthiban 等學者[9]也採用層級分析法並且提出一個以喜好為基礎之教師派課決策模型。本研究[1][2]過往也曾採用基因演算法(Genetic Algorithm, GA)建構大專院校之派課系統，並可依照系所之需求調整權重值以產生合適的派課結果。

3. 問題描述

本研究所探討的派課問題是指一個科系如何將下學期所欲開設之課程分派給系上之授課老師。派課問題必須考慮到諸如教師喜好，教師專長，行政職務等因素，各因素有其輕重，因此，如同排課問題一般，我們將限制式區分為兩大類來考慮，分別為：硬性限制(Hard Constraint, HC)以及軟性限制(Soft Constraint, SC)。所謂的硬性限制是指因為資源限制，學校規定或班級課程不能衝堂之性質，所必須要滿足的派課條件，是不可避免的條件。而所謂的軟性限制則是問題中必須儘量去滿足的條件。大部份的軟性限制都是人為因素的考量。

3.1 硬性與軟性限制

本研究之硬性限制與軟性限制列舉如下：

硬性限制：

- HC1 同一門課程只能有一名教師被指派。
- HC2 符合系所之學分數限制
- HC3 所有班級的必修課程必須有指派教師。

軟性限制：

- SC1 被指派之課程的領域必須盡量與教師之專長領域相符合。
- SC2 同一教師被指派之學分數差異度越小越好。
- SC3 同一教師被指派之相異課程數目越少越好。
- SC4 教師的授課偏好應該儘量被滿足。

其中 SC1 軟性限制是為了達到適才適所之目的；而 SC2 軟性限制之設定主要是因為多數的教師不喜歡所被指派的學分太多，此軟性限制可以降低教師的負擔，進而給學生較好的學習品質；而 SC3 軟性限制主要目的在於期望相同課目最好由同一教師擔任。

對於所探討之派課問題所牽涉之軟性限制，我們必須事先收集所需之資料以利將來演算法之運算。這些資料包含「教師授課喜好表」、「課程領域分析表」、「教師專長分析表」等三項表格，分述如下：

「教師授課喜好表」：為一份包含該學期所有課程之課程表，對於一個學期所要開設之

課程，每位教師對於課程提供其喜好度，喜好度等級區分為 10 等，其中 10 代表最滿意，1 代表最不滿意。每位教師需提供 10 個科目的喜好度於此表格上，如表 1 所示。

表 1 教師授課喜好表

科目名稱 \ 喜好度	教師一	教師二	教師三	...
課程一	5		7	
課程二	4			
課程三		8	2	
課程三		8	2	
課程四	10	3		

「課程領域分析表」：對於全系之課程結構做一綜合性分析並且建構課程之領域結構圖。一個課程可能只屬於某特定領域，但也有可能隸屬於不同的領域，表 2 顯示課程 1 屬於領域 C，課程 2 屬於領域 A 與 B，課程 3 則屬於領域 A 與 D。

表 2 課程領域分析表

課程 \ 領域	課程一	課程二	課程三	...
領域 A		v	v	
領域 B		v		
領域 C	v			
領域 D			v	

「教師專長分析表」：這是一份由教師所填寫之專長分析表，用以顯示其所屬之專長領域，其目的在於執行派課演算法時做為 SC1 計算之用，如表 3 所示。

表 3 教師授課喜好表

領域 \ 教師	領域 A	領域 B	領域 C	領域 D
教師 1	v		v	v
教師 1		v	v	
教師 3		v	v	v
教師 4	v			v

4. 研究方法

4.1. 粒子編碼(particle coding)

根據粒子群演算法之執行步驟，首先定義本研究所要採用之粒子的編碼方式。一個粒子代表一個對於問題的解，因此，對於派課問題

所定義出的粒子則代表一個派課的結果。對於派課結果之粒子編碼型態，我們定義如下：令粒子 X 代表一派課結果， X 由 n 個子粒子所組成，其中 n 為這學期所決定開設的課程數量，每個課程包含 m 個 0 或 1 的值，其中 m 為教師數量，所以每個子粒子代表各個課程所排定的派課結果。粒子編碼如圖 2 所示。

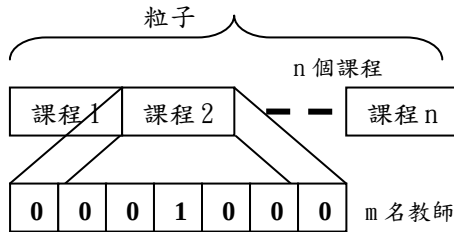


圖 2 粒子編碼示意圖

令 $C = \{C_1, C_2, \dots, C_n\}$ 表示所有課程所成之集合， $T = \{T_1, T_2, \dots, T_m\}$ 表示所有教師所成之集合。定義決策變數 x_{ij} 如下：

- $x_{ij}=1$ 假如教師 T_j 被指派到課程 C_i ；
 $x_{ij}=0$ 假如教師 T_j 沒有被指派到課程 C_i ；
 $1 \leq j \leq m, 1 \leq i \leq n$ 。

所以對於派課之結果可以用一個矩陣 $X_{n \times m}$ 來表示。所以本研究以矩陣的形式來表示一個粒子。另外，我們將前置作業所收集之「教師授課喜好表」轉換成喜好度矩陣 $F_{n \times m}$ 。喜好度矩陣中之每一列(row)代表所有教師對於該課程之喜好度，每一行(column)代表一名教師對於所有課程之喜好度。

4.2. 適應度函數(fitness function)

為評估粒子的好壞，本研究訂定一適應度函數，透過適應函數的計算，當粒子的適應值越佳，表示其所代表的解越能滿足問題的要求。本研究之適應度函數由「派課總體喜好度(Total Average Preference, TAP)」以及「成本函數(Cost Function)」兩大部分組合而成，令 X 表示一個粒子，則其適應度函數定義如下：

$$\text{fitness}(X) = \frac{\text{TAP}(X)}{1 + \text{cost}(X)} \quad (1)$$

由上之公式可以看出，一個相當符合絕大多數教師喜好之派課結果所計算出之適應值會趨

近於 1。這是因為對於該粒子 X ，我們會有 $\text{TAP}(X) \rightarrow 1$ 以及 $\text{cost}(X) \rightarrow 0$ 。下列小節分述各組成部分的定義。

4.3. 派課總體喜好度(TAP)

「派課總體喜好度」代表全體教師對某一派課結果的平均喜好值。TAP 之計算由派課結果與「教師授課喜好表」矩陣 $F_{n \times m}$ 之轉置相乘而得。令 $I = \{1, \dots, n\}$ 代表課程數目， $J = \{1, \dots, m\}$ 代表教師數目， $P = \{P_1, \dots, P_n\}$ 代表喜好值所成的集合， n 代表課程總數，則 TAP 定義如下：

$$\text{TAP}(X) = \frac{\sum_{i \in I} \sum_{j \in J} x_{ij} p_{ji}}{\max(P) \cdot n} = \frac{\text{tr}(X_{n \times m} F_{n \times m}^T)}{\max(P) \cdot n} \quad (2)$$

其中 $F_{n \times m}^T$ 表示喜好度矩陣 $F_{n \times m}$ 的轉置， $\text{tr}(A)$ 表示矩陣 A 的 trace， $\max(P)$ 代表集合 P 中之最大值。根據此定義，一個滿意的派課結果(指派給教師之課程皆滿足其最大喜好度)其派課整體喜好度之值將會趨近於 1。

4.4. 成本函數(Cost Function)

成本函數的責任在於決定一個粒子的好與壞，讓演算法可以根據此資訊接續著相關操作。對於一個粒子 X ，定義其 Cost 函數如下：

$$\text{Cost}(X) = \sum_{i=1}^3 w_i \times d_i(X) \quad (3)$$

其中 w_i 為 $d_i(X)$ 之權重值。採用動態權重值，我們可以根據不同的需求來產生合適之課表。

在 Cost 函數中， $d_1(X)$ 代表違反 SC1 之 cost， SC_1 表示被指派之課程的領域必須盡量與教師之專長領域相符合。為了讓派課結果儘量符合適才適所的目的，達到最大的專長符合度，對於任意染色體 X ，函數 d_1 定義如下：

$$d_1(X) = \sum_{j \in J} \sum_{i \in I} x_{ij} (a_i - T_{ji}) \quad (4)$$

其中 a_i 代表課程 i 所需要的專長領域數目， T_{ji} 代表教師 j 在課程 i 所擁有的專長數。 a_i 與 T_{ji} 這兩部分的資料來自於表 2 與表 3。 $d_1(X)$ 之設計考量主要是對於派課之結果計算出是否

有將適任之教師指派到相關課程。此計算值期望越小越好，才能符合適才適所的目標。

$d_2(X)$ 代表違反 SC2 之 cost，SC2 表示教師被指派之課程數差異度越小越好。亦即希望每位教師被指派之學分數能夠平均分配，而避免某些教師之授課學分數過多或過少的情況。為了計算此軟性限制式之成本，對於每位教師，我們首先計算其授課學分數，然後，將此結果做為一個成本增量函數的輸入以計算出其成本，最後再將其累加起來，因此，函數 $d_2(X)$ 定義如下：

$$d_2(X) = \sum_i CI(\sum_j x_{ij} c_i) \quad (5)$$

其中 c_i 代表課程 i 之學分數， $CI(n)$ 代表上述之成本增量(Cost Increment, CI)函數。其 $CI(n)$ 定義如(6)。

$$CI(n) = \left[\left| n - \frac{\sum_i C_i}{i} \right|^2 \right] \quad (6)$$

從成本增量函數之定義上我們可以看出，當一位教師所分派之學分數與平均學分數差異越大，將會使此函數呈幾何成長，造成限制式(5)的成本越大。

$d_3(X)$ 代表違反 SC3 之 cost，其中 SC3 表示同一教師被指派之相異課程數目越少越好之限制式。因此，函數 $d_3(X)$ 定義如下：

$$d_3(X) = \sum_j CI(\#(\bigcup_i G(x_{ij})) - [\frac{k}{m}]) \quad (7)$$

其中 $G(x_{ij})$ 代表課程 i 所在之課組的集合， $\#(S)$ 代表集合 S 中相異元素的個數， k 代表相異之課程數， m 代表教師人數，記號 $[x]$ 表示不大於 x 之最大整數。所謂的課組是指由課程名稱相同之課程所成的集合。例如：數學系負責全校微積分課程的開授，因此，該系開設四門微積分課程以供全校學生選修，如表 4 所示。雖然說實質上有四門課程，但其皆屬於同一課程，因此將這四門微積分課程歸類於同一課組，此一課組即為集合 $\{\text{Calculus_001}, \dots, \text{Calculus_004}\}$ 。 $G(x_{ij})$ 之計算方式以下表為例做說明。對於教師 1，雖然說他/她被指派到

三門微積分課程，但是實際上這三門課是同一課程。

表 4 課組範例表

		教師 1	教師 2	...
1	Calculus_001	v		
2	Calculus_002	v		
3	Calculus_003		v	
4	Calculus_004	v		

因此， $\bigcup_i G(x_{i1}) = \{\text{Calculus}\}$ ，所以 $\#(\bigcup_i G(x_{i1})) = 1$ 。對

於其他的教師，依此類推計算並且加總起來作為 $d_3(X)$ 的值。利用此計算方式，很顯然地可以看出 $d_3(X)$ 的值代表派課結果中教師被指派之相異課程數目，因此期望此數目越小越好。

4.5. 粒子群演算法(PSO)

粒子群演算法主要透過粒子間彼此分享經驗，達到更新粒子位置的目的，每個粒子會保存各自目前最好的適應值與位置，稱為粒子最佳值(Particle best value, Pbest);以及目前群體中最好的適應值與位置，稱之為群體最佳值(Globe best value, Gbest)。每個粒子會根據粒子最佳值與群體最佳值更新粒子速度，其粒子移動與速度更新公式如下：

$$V_{ij}^{t+1} = W \times V_{ij}^t + c_1 \times r_1 \times (Gbest_{ij} - X_{ij}^t) + c_2 \times r_2 \times (Pbest_{ij} - X_{ij}^t) \quad (4)$$

$$X_{ij}^{t+1} = X_{ij}^t + V_{ij}^{t+1} \quad (5)$$

其中 V_{ij} 為粒子速度， r_1 及 r_2 為介於 0 與 1 之間的隨機變數， c_1 及 c_2 為學習因子， W 為權重值，本研究採用 Shi 和 Eberhart[12]所提出之線性慣量遞減權重(Linear inertia Reduction)，其公式如下：

$$W = W_{\max} - \frac{W_{\max} - W_{\min}}{iter_{\max}} \times iter \quad (6)$$

其中 $W_{\max}$ 為權重最大值， $W_{\min}$ 為權重最小值， $iter$ 為演算法執行代數，而 $iter_{\max}$ 為執行代數最大值。利用線性遞減慣性權重的概念，可使得演算法執行初期有較高的慣性權重值，藉以

保持全域搜尋的靈活性，並且隨著執行過程逐漸降低慣性權重值，進而轉入局部搜尋之動作，以加強粒子之局部搜尋能力。

由於粒子群演算法只能針對問題解為連續實數的研究範疇，而傳統離散式粒子群演算法又不適合本研究之粒子編碼，因此對於單一課程 C_i 本研究將公式(5)替換為公式(7)，其公式如下：

$$\text{if } j = \arg \max V_{ij} (j=1,2,\dots,m) \text{ then } X_{ij}^{k+1} = 1; \quad (7) \\ \text{else } X_{ij}^{k+1} = 0;$$

根據上式利用此一公式可確保粒子移動後不會違反硬性限制，破壞演算法之粒子結構。

5. 研究成果

本節中，我們模擬一個六名教師、二十門課程的大學科系，利用本系統進行派課問題之研究，其演算法執行次數設為 10000 代，而粒子群大小設為 60 組。而表 5 為教師們所填寫對於二十門課程之教師授課喜好表，包含各教師對於欲開設課程的喜好度。

表 5 教師授課喜好表表

教師與教師編號 課目與課程編號		教師 1	教師 2	教師 3	教師 4	教師 5	教師 6
		t1	t2	t3	t4	t5	t6
統計學-1	01-1	3	6	9		6	
統計學-2	01-2	3	6	9		6	
統計學-3	01-3	3	6	9		6	
知識管理	02	9			1	5	6
電子商務	03	5	3	8			9
資料庫管理系統-1	04-1		8		6	4	8
資料庫管理系統-2	04-2		8		6	4	8
供應鏈管理	05	8	5	7			5
企業資料通訊	06		4		7	10	1
基礎網路管理	07	4		10	3		4
系統分析設計-1	08-1		7	1	4		2
系統分析設計-2	08-2		7	1	4		2
文書處理軟體	09		1	2	8	3	
行銷管理	10	2				8	10
伺服器網頁設計	11	1		5	10	1	
程式設計-1	12-1	7	9	6	2		
程式設計-2	12-2	7	9	6	2		
資料探勘	13	10	10	3		7	3
企業資源規劃	14		2		9	2	7
專業英文選讀	15	6		4	5	9	

表 6 為課程領域分析表，資料蒐集部分需對於所有課程的課程結構做一綜合性分析。一個課程可能只屬於某特定領域，但也有可能隸屬於不同的領域析。

表 6 課程領域分析表

		領域 A	領域 B	領域 C	領域 D	領域 E	領域 F
統計學-1	01-1		✓				
統計學-2	01-2		✓				
統計學-3	01-3		✓				
知識管理	02	✓					
電子商務	03	✓		✓			
資料庫管理系統-1	04-1	✓			✓		
資料庫管理系統-2	04-2	✓			✓		
供應鏈管理	05	✓					
企業資料通訊	06			✓		✓	
基礎網路管理	07	✓		✓		✓	
系統分析設計-1	08-1	✓			✓		
系統分析設計-2	08-2	✓			✓		
文書處理軟體	09					✓	
行銷管理	10	✓					
伺服器網頁設計	11			✓	✓	✓	
程式設計-1	12-1				✓	✓	
程式設計-2	12-2				✓	✓	
資料探勘	13	✓	✓			✓	
企業資源規劃	14	✓				✓	
專業英文選讀	15						✓

表 7 為教師專長分析表，包含所有教師的專長分析，其與課程領域分析表將作為計算 SC1 的主要依據。

表 7 教師專長分析表

專長領域 教師與教師編號		領域 A	領域 B	領域 C	領域 D	領域 E	領域 F
教師 1	t1	✓	✓				✓
教師 2	t2	✓		✓	✓	✓	
教師 3	t3				✓	✓	✓
教師 4	t4			✓	✓	✓	
教師 5	t5		✓	✓			✓
教師 6	t6	✓	✓		✓		

以下為本研究將上述表格輸入系統後，經過粒子群演算法操作所產生之派課結果，我們可以觀察到輸入不同的權重值對於整體派

課結果之影響:圖 5 之派課結果以被指派之課程的領域必須盡量與教師之專長領域相符合作為其派課依據,其權重值偏向 SC1 以期望派課結果會盡量符合教師之專長,圖 5 中 SC1 的違反數量為 2, SC2 的違反數量為 6, SC3 的違反數量為 5,顯示出派課結果均以教師專長是否能配合課程領域作為其主要派課考量。

	t0	t1	t2	t3	t4	t5	t6
000	0	0	0	0	0	0	0
01-1	0	0	0	0	0	0	0
01-2	0	0	0	0	0	0	0
01-3	0	0	0	0	0	0	0
02	0	0	0	0	0	0	0
03	0	0	0	0	0	0	0
04-1	0	0	0	0	0	0	0
04-2	0	0	0	0	0	0	0
05	0	0	0	0	0	0	0
06	0	0	0	0	0	0	0
07	0	0	0	0	0	0	0
08-1	0	0	0	0	0	0	0
08-2	0	0	0	0	0	0	0
09	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0
12-1	0	0	0	0	0	0	0
12-2	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0

圖 5 派課結果(權重值=0.8, 0.1, 0.1)

圖 6 針對降低教師被指派之課程數差異度作為主要依據,所以權重值偏向 SC2 希望能夠平衡教師的授課數量,根據此派課結果可得知圖 6 中 SC1 的違反數量為 9, SC2 的違反數量為 2, SC3 的違反數量為 5,我們可觀察出圖 6 中各個教師之授課數皆位於課程平均數正負一個標準差的範圍之內,足見此派課結果期望降低教師被指派課程數的差異度,避免部分教師授課學分數過多或過少之情形。

	t0	t1	t2	t3	t4	t5	t6
000	0	0	0	0	0	0	0
01-1	0	0	0	0	0	0	0
01-2	0	0	0	0	0	0	0
01-3	0	0	0	0	0	0	0
02	0	0	0	0	0	0	0
03	0	0	0	0	0	0	0
04-1	0	0	0	0	0	0	0
04-2	0	0	0	0	0	0	0
05	0	0	0	0	0	0	0
06	0	0	0	0	0	0	0
07	0	0	0	0	0	0	0
08-1	0	0	0	0	0	0	0
08-2	0	0	0	0	0	0	0
09	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0
12-1	0	0	0	0	0	0	0
12-2	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0

圖 6 派課結果(權重值=0.1, 0.8, 0.1)

根據圖 7 可以觀察出來其權重值偏向 SC3, 意指期望同一教師被指派之相異課程數目越少越好,在派課結果上皆主要針對減少相異之課程數作為其派課依據,其圖 7 中 SC1 的違反數量為 10, SC2 的違反數量為 4, SC3 的違反數量為 3,在此派課結果中具有兩個以上相同課目之課程皆未指派給不同教師,此派課結果降低了同一教師的相異課程數量,有效降低教師備課負擔。

	t0	t1	t2	t3	t4	t5	t6
000	0	0	0	0	0	0	0
01-1	0	0	0	0	0	0	0
01-2	0	0	0	0	0	0	0
01-3	0	0	0	0	0	0	0
02	0	0	0	0	0	0	0
03	0	0	0	0	0	0	0
04-1	0	0	0	0	0	0	0
04-2	0	0	0	0	0	0	0
05	0	0	0	0	0	0	0
06	0	0	0	0	0	0	0
07	0	0	0	0	0	0	0
08-1	0	0	0	0	0	0	0
08-2	0	0	0	0	0	0	0
09	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0
12-1	0	0	0	0	0	0	0
12-2	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0

圖 7 派課結果(權重值=0.1, 0.1, 0.8)

圖 8 之派課結果的權重值皆取 0.33, 以產出符合所有限制式之均衡課表,圖 8 中 SC1 的違反數量為 5, SC2 的違反數量為 2, SC3 的違反數量為 3,我們可以觀察出此派課結果不但維持了教師專長與課程領域相符合的需求,將課程平均分派給教師,並且有效降低教師在授課前的備課負擔。

	t0	t1	t2	t3	t4	t5	t6
000	0	0	0	0	0	0	0
01-1	0	0	0	0	0	0	0
01-2	0	0	0	0	0	0	0
01-3	0	0	0	0	0	0	0
02	0	0	0	0	0	0	0
03	0	0	0	0	0	0	0
04-1	0	0	0	0	0	0	0
04-2	0	0	0	0	0	0	0
05	0	0	0	0	0	0	0
06	0	0	0	0	0	0	0
07	0	0	0	0	0	0	0
08-1	0	0	0	0	0	0	0
08-2	0	0	0	0	0	0	0
09	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0
12-1	0	0	0	0	0	0	0
12-2	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0

圖 8 派課結果(權重值=0.33, 0.33, 0.33)

下圖為本研究模型以權重值皆取 0.33，執行 100 次運算之適應值平均演化曲線、最佳演化曲線與最差演化曲線圖，可由圖中觀察得知平均演化曲線約在 4000 代後開始收斂，於 6000 代後趨近穩定。運算結果中世代最大適應值為 0.40013，世代平均適應值為 0.33091，運算結果世代最小適應值為 0.24049，平均每世代執行所需運算時間為 3 分鐘，證實此應用粒子群演算法之派課模型可使用於求解派課問題。

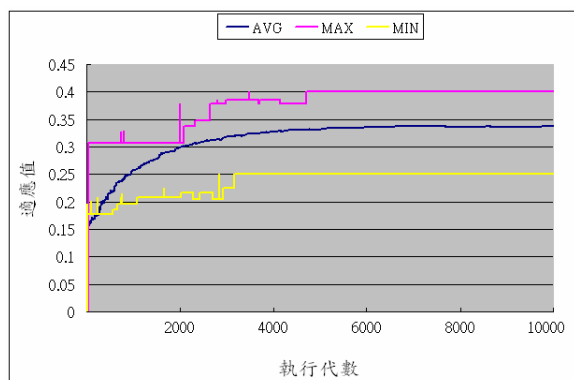


圖 9 適應值演化曲線圖

6. 結論

粒子群演算法為近十年來廣為討論的最佳化演算法，由於現今大專院校之派課方式大部分皆採用人工作業，不僅費時費力，更可能因人工派課的關係在作業時受人為因素之影響。因此，本研究希望藉由粒子群演算法建構一大專院校派課系統，利用電腦作業維持派課的公平性，並產生出適才適所且滿足教師期望之派課結果。本研究結合了決策變項之動態調整，將派課問題中的數個變項加入權重值，期望作業人員可依據其系所之授課規範亦或師資情況調整權重值以產生合適的派課結果。

未來發展方面希望與實務作結合，藉由了解各大專院校之派課方式設立更多的決策變數，以適應不同校系間之教學型態。並且探討不同演算法之運算方式對於系統執行的影響，以期建構一更高效之派課系統，亦可結合排課系統延續此研究理念，幫助作業人員減少每學期的負擔，以期提高作業效率並且提升師生對於課程之滿意度。

參考文獻

[1] 游峰碩,王子夏,運用基因演算法於大專院校教師派課問題之研究,2007 年資訊科技國際研討會,2007。

[2] 游峰碩,王子夏,運用基因演算法建構大專院校派課系統之研究,台灣作業研究學會 2007 年學術研討會,2007。

[3] B. Al-kazemi and C. K. Mohan, "Multi-phase discrete particle swarm optimization", in Proc. Intl. Workshop Frontiers in Evolutionary Algorithms, 2002.

[4] De Jong, K. A., "An analysis of the behavior of a class of genetic adaptive systems", Doctoral dissertation, University of Michigan, 1975.

[5] Hultberg, T. and Cardoso, D., The teacher assignment problem: A special case of the fixed charge transportation problem, European journal of operational research, vol.101, pp. 463-473, 1997.

[6] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm optimization", in Proc. IEEE Conf. Syst., Man, and Cybernetics, Orlando, FA, pp.4104-4109, 1997.

[7] J. Kennedy and R. Eberhart, "Particle swarm optimization", in Proc. IEEE Intl. Conf. Neural Networks, vol. 4, pp.1942-1948, 1995.

[8] Ozdemir, M. S., and Gasimov, R. N. The analytic hierarchy process and multiobjective 0-1 faculty course assignment, European Journal of Operational Research, 157, pp. 398-408, 2004.

[9] Parthiban, P., Ganesh, K., Narayanan, S. Dhanalakshmi, R., Preferences Based Decision-making Model (PDM) for Faculty Course Assignment Problem, In Proceedings of Engineering Management Conference, IEEE International, pp. 1338-1341, 2004.

[10] S. Yang, M. Wang, and L. Jiao, "A quantum particle swarm optimization", in Cong. Evolutionary Computing, Vol.1, pp.320-324, 2004.

[11] Y. Shi and R. Eberhart, "A modified particle swarm optimizer", in Proc. IEEE World Cong. On Computational Intelligence, pp.96-73, 1998.

[12] Y. Shi and R. Eberhart, "Parameter Selection in Particle Swarm Optimization," V. W. Porto, N. Saravanan, D. Waagen, and A. E. Eiben (eds), Lecture Notes in Computer Science, 1447, Evolutionary Programming VII, Springer, Berlin, pp. 591-600, 1998.