

粒子群演算法區域搜尋能力之探討問題

張家豪¹ 藍國桐^{1*} 藍俊雄²

德明財經科技大學 醒吾科技大學

資訊科技學系¹ 企業管理學系²

E-mail: ktlan@takming.edu.tw

摘要

本論文討論進化演算法與粒子群演算法的不同之處，並且提出一可以改善粒子群演算法易陷入局部最佳處的缺點。進化演算法(evolutionary algorithm; EA)認為一個適應性系統應該是一個族群，其中又包含了很多個個體(individuals)，這樣的一個族群不斷的演化(evolution)以尋求最能適應環境情況的個體或族群。演化的世代交替並不是按部就班的探索著各種遺傳組合的可能性，而是各種嘗試齊頭並進，這就是所謂族群式(population-base)的搜尋。採用族群對族群的方式是要利用多方向探索的方式(exploring the search space)，並且捨棄一步一步在可能的解空間中循序搜尋的傳統方法，改採重組個體、大膽改變個體的方式，以跳離局部最佳解。粒子群演算法也採用族群對族群的方式，利用群體中每個個體根據本身經歷過的最佳解，以及目前族群中的最佳解，調整個體的位移(movement)。它會向較佳同伴的方向來搜尋解空間，這種是一種採掘式(exploiting the best solution)的搜尋方法。雖然它以快速向最佳解靠近為其優點，但由於複雜問題的搜尋空間往往具有非線性變化現象，加上在搜尋空間中存有相當多的局部最佳點，這些局部最佳點常常會成為搜尋上的陷阱。但是進化演算法的缺點就是搜尋時間過於冗長，無法被應用到即時性的系統上。本研究提出一個粒子群演算法的改善版本，藉由加入演化的技巧，我們可以提高粒子群演算法的探索能力，但又不影響其快速搜尋能力。

關鍵詞：進化演算法，粒子群演算法，族群式搜尋，探索式搜尋，採掘式搜尋。

一、前言

無論是工程上，經濟行為上，科學研究，甚至日常生活上，我們經常要面臨最佳化(optimization)的問題，如時間最少、速度最快、效率最好等等。面對這類的問題，一般都需要先對問題及解決方案(solution)之間作一個評估(evaluate)，所以會得到一個目標函數(objective function, cost function, fitness function)，根據這個函數所得到的函數值(fitness value)我們可以評估這個解決方案的好壞並作為改善此方案或是另謀他途的依據。傳統求最佳解的方法乃是根據這個目標

函數作梯度(gradient)或其他高階(high-order)統計方法，以求得一個搜尋的方向，並循此方向改善解決方法，這種方法我們統稱為登山式搜尋(hill-climbing)。由於這些登山式搜尋(hill-climbing)的方法常常會有收斂在局部最佳點(local optima)的問題[1]。利用演化計算(evolutionary computation)可以解決此類的局部收斂問題(又稱為早熟問題，premature)。它的演算過程中具有突變(mutation)及天擇(selection)兩種機制可以對複雜崎嶇(rugged)的地形作有效的搜尋(這種搜尋並不是以局部的地貌資訊來作為搜尋的依據)。突變會導致所產生的下一代個體的內在或外形發生隨機性的變異，它是族群中的個體嘗試向未知的空間探索的重要機制，而天擇則會產生一個天擇壓力(selection pressure)來驅趕族群個體邁向適合度更高的狀態。

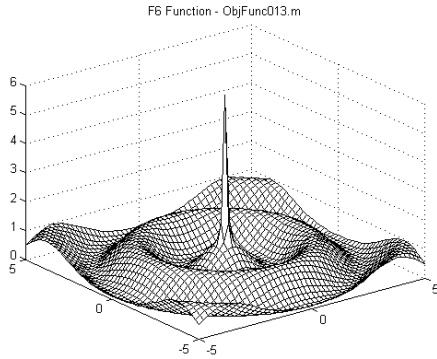
雖然進化演算法具有跳脫局部最佳的能力，但是它最大的缺點就是冗長的演化時間，這點不利於需要作即時控制的系統，粒子群演算法就具有即時計算的優點[2-4]。利用族群之中的目前最佳個體與本個體的差距，並再利用本身目前的狀況與歷史的最佳狀況之間的差距，來調整本身的位置以達到搜尋整個空間的目標。由於它是一個類似登山法的族群搜方式，因此快速尋得最佳點是它的優點，但是和一般登山式搜尋一樣，早熟問題仍然是它的問題之一。

本論文針對如何設計一個具有探索能力的粒子群演算法作為研究目標。在探討進化演算法與粒子群演算法之後，我們將突變的機制放到粒子群演算法之中，在對一複雜搜尋空間的鑑別的問題上，這研究的結果應該可以讓粒子群演算法的應用更為廣泛，並解決演化計算的冗長計算問題。本論文以下的章節規劃成：第二節比較分析兩種演算法的差異列出本論文針對圖樣設計所適用的進化演算法中演化個體的設計，第三節則是設計一個粒子移動過程中的突變機制，第四節是我們針對幾個典型的複雜搜尋空間，利用本演算法的實驗結果分析，第五節作一個總結。

二、演算法差異化比較

針對一個複雜無梯度資訊的非線性空間，本論文以進化演算法(Evolutionary Algorithm; EA)及粒子群演算法(Particle Swarm Optimization; PSO)來

搜尋其複雜的目標函數地貌(landscape)。在實作演算法中，我們必須對每個族群中的個體(individual)指定一個適合度(fitness value)給它，這個適合度函數端視環境的要求而定。在本研究中我們使用一個 Schaffer 函數來作為欲作搜尋的目標函數地貌，如圖一。



圖一 Schaffer 目標函數的地貌曲線圖

除了如上述的目標外，我們在演算法初始狀態時，將族群分佈限定在解空間的某個小範圍內，這種作法可以測出演算法的搜尋能力。因為在實際上我們所面臨的最佳化問題中，它的解空間範圍(feasible region)都相當大，我們都是從一個小範圍開始對整個解空間作搜尋。

2.1 進化演算法(Evolutionary Algorithm; EA)

進化演算法是用來搜尋其目標函數地貌(landscape)。在演化計算中，對每個族群中的個體(individual)而言，我們必須指定一個適合度(fitness value)給它，這個適合度函數 $f:R^m \rightarrow R$ 端視環境的要求(即我們希望演化出來的目標)而定。對每個個體 \bar{x}_i 而言，它是由 m 個實數分量所構成的， $\bar{x}_i = [x_i(1) \ x_i(2) \ \dots \ x_i(m)]$ ，當它執行突變的動作時，每個實數分量都加入一個隨機數 Δx ，這個隨機數一般都是採用高斯隨機數(Gaussian distribution)或是歌西隨機數(Cauchy distribution)，即 $\Delta \bar{x} = \Delta r \cdot \bar{N}(0,1)$ 。由[5, 6]的結論，我們已經了解歌西式突變的優勢，因此本研究將直接採用。

D. B. Fogel 綜合各種進化演算法的特性，提出來一個完整的進化演算法，它所採用的方式是重複下列步驟[7, 8]：

1. (族群初始分佈; initialization) 由 m 維的向量空間中任選 n 個候選向量當作族群的個體，即

$$\bar{x}_i = [x_i(1) \ x_i(2) \ \dots \ x_i(m)],$$

$$i=1, 2, \dots, n$$

2. (複製及突變; reproduction & mutation) 首先

由這 n 個父代， \bar{x}_i ，個別複製一個子代。再利用歌西式的隨機數：平均值為 0 且預設一個標準差 $\Delta r = \sigma$ ，對每個子代作突變的動作。即若 \bar{x}_i 的子代為 \bar{x}_i' 則

$$\bar{x}_i' = \bar{x}_i + \bar{R} = \bar{x}_i + \Delta r \cdot \bar{R}$$

隨機變數向量 \bar{R} 是由 m 個歌西隨機數 C 所構成：

$$\bar{R} = [C \ C \ \dots \ C]$$

3. (排序式天擇; ranking selection) 比較父代及子代共 $2n$ 個個體的適合度值， $f(\bar{x}_i)$ 及 $f(\bar{x}_i')$ ，並選出 n 個適合度最高的個體保留至下一代。
4. 回到步驟 2，除非已經找到適當的解答。

2.2 粒子群演算法(Particle Swarm Optimization; PSO)

K. Kennedy 提出來的一個粒子群演算法(Particle Swarm Optimization; PSO)，它所採用的方式是重複下列步驟[2-4]：

1. (族群初始分佈; initialization) 由 m 維的向量空間中任選 n 個候選向量當作族群的個體，即族群大小(population size)為 n ：

$$\bar{x}_i = [x_i(1) \ x_i(2) \ \dots \ x_i(m)],$$

$$i=1, 2, \dots, n$$

2. (個體移動; movement) 首先針對每個個體， \bar{x}_i ，找出其和族群中的目前最佳個體 \bar{p}_i 的

距離： $d_i = \|\bar{x}_i - \bar{p}_i\|$ ，以及其和個體 \bar{x}_i 的歷史最佳記錄 \bar{p}_i 的距離：

$d_{best} = \|\bar{x}_i - \bar{p}_i\|$ 。利用這兩項資訊，對個體 \bar{x}_i 作調整(我們稱為 movement)。即

2. 個體它只會向 \bar{P}_i, \bar{P}_g 移動。
3. \bar{P}_i 的更動只能靠個體 x_i 在變動中，獲得在 x_i 的函數地貌附近的較好位置。
4. \bar{P}_g 的更新只能靠著族群中任何一個個體在搜尋過程中，獲得在函數地貌附近的較好位置。

由於粒子群演算法只和體族群中的目前最佳個體與歷史最佳記錄作比較，因此會喪失有關跳脫局部最佳解的能力。因此我們在步驟 3 加上一個不定時的突變機制，利用這個突變(mutation)的機制來對族群中除了最佳者 (\bar{P}_g) 外的所有的個體作隨機式的變動。

其演算法程式碼(algorithm pseudo code)如下：

```

DO
Initialize the population.
For i=1 to {
Update  $\bar{P}_i, \bar{P}_g$ 

```

其中：

φ_1, φ_2 兩個位於 [0,1]的隨機正數.

3. 回到步驟 2，除非已經找到適當的解答。

其演算法程式碼(algorithm pseudo code)如下：

```

DO
Initialize the population.
For i=1 to {
Update  $\bar{P}_i, \bar{P}_g$ 

```

```

Calculate the fitness value:  $f(x_i)$ 
}
Until termination criterion is met.

```

φ_1, φ_2 are positive random numbers between 0 and 1.

\bar{P}_g is the global best term so far for the whole population.

\bar{P}_i is called the best term encountered this far for the individual i .

三、粒子移動過程中的突變機制的設計

在粒子群演算法中，針對每個個體的移動(movement)是該演算法能夠探索整個可行解空間的能力。檢視其作法，我們會發現：

1. 它只跟個體族群中的目前最佳個體與歷史最佳記錄 \bar{P}_i, \bar{P}_g 有關。

```

Calculate the fitness value:  $f(x_i)$ 
}
For i=1 to {

```

```

if (
 $\bar{x}_i' = \bar{x}_i + \bar{w} = \bar{x}_i + \varphi_1 R$ 
)

```

```

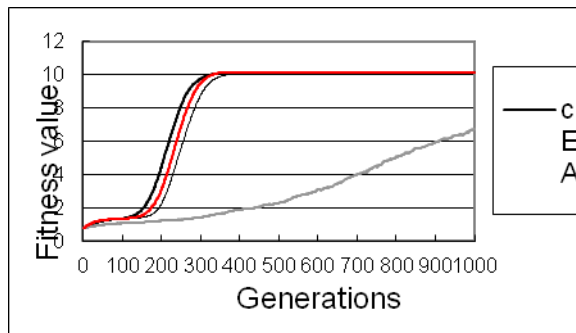
}
Until termination criterion is met.

```

四、實驗結果與分析

針對 Schaffer 測試函數，整個解空間為

，而族群初始分佈的空間定義為，和整個解空間比較起來，族群初始分佈空間可說非常的小。在本研究的實驗中，我們定義各個測試函數均為 2 個自變數，也就是個體字串是由 2 個實數所構成的，族群個體數目均為 10，演化代數以 1000 代為主。



圖二 佈建基地為 10 X 10 X 10，使用 60 顆人工漁礁限制下的最佳設計，FD=2.3843

五、結 論

本研究利用粒子群演算法的優點，並加上演化計算中突變所帶來的探索能力，來設計一個具有跳脫局部收斂解的粒子群演算法。由於複雜搜尋地貌空間具有非線性變化，加上搜尋空間隨維度呈指數形式的變大(即規模問題(scaling))，因此一般傳統登山式搜尋演算法或是傳統的粒子群演算法並無法成功得到最佳的搜尋結果。利用跳脫能力較佳的突變方式(如歌西式的突變方式)引入到粒子群演算法中，可以讓粒子群演算法具有比較好的空間搜尋能力，也不易陷入局部最佳解上。未來將以這個改良式的粒子群演算法應用到一些具有非線性複雜動態系統的即時控制領域上。

參考文獻

- [1] 藍國桐，競爭式進化演算法理論分析及其在非線性動態系統之應用，台灣科技大學電機系博士論文，2000年。
- [2] J. Kennedy, "The particle swarm: Social adaptation of knowledge," in *Proc. 1997 Int. conf. Evolutionary Computation*, Indianapolis, IN, Apr. 1997, pp.303-308.
- [3] M. Clerc and J. Kennedy, "The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex space," *IEEE Trans. Evol. Comput.*, vol.6, no.1, pp.58-73, 2002.

- [4] Y. D. Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. G. Harley, "Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems," *IEEE Trans. Evol. Comput.*, vol.12, no.2, pp.171-195, 2008.
- [5] 藍國桐，藍俊雄，王李吉，"應用進化演算法於圖樣設計問題"，in *Proc. of the 11th Artificial Intell. and Applications*, (第十一屆人工智慧及其應用會議論文集) Kaohsiung, Taiwan, 2005。
- [6] K. T. Lan and C. H. Lan, "Notes on the Distinction of Gaussian and Cauchy Mutations," to be appearing in *Proc. of the 8th Intell. Syst. Design and Applications*, Kaohsiung, Taiwan, 2008
- [7] D. B. Fogel, "An introduction to simulated evolutionary optimization," *IEEE Trans. Neural Net.*, vol.5, no.1, pp.3-14, 1994.
- [8] D. B. Fogel, *Evolutionary Computation: Towards a New Philosophy of Machine Intelligence*, Discataway, NY: IEEE Press, 1995.