計算型智慧-作業三 RBFN參數PSO粒子群最佳化

授課教授：蘇木春　教授

學號：104522065

學生：張翔珳

**1.使用環境**

作業系統：windows 7 or windows 8.1 64位元

使用語言：C++

編譯平台：Dev C++ 5.11

**2.基因演算法架構**

目前測試結果較佳參數如表1所示。

表1 測試參數結果

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 參數項目 | 迭代次數 | 神經元數量 | 粒子數量 | 個體過去最佳參考係數 | 群體過去最佳參考係數 | 隨機加入雜訊大小 | 初始值 |
| 參數數值 | 150 | 20 | 500 | 0.8🡪0.0  隨迭代次數減少 | 0.2🡪1.0  隨迭代次數增加 | -0.1~0.1 | 1 |

類神經網路RBFN的部分是以20顆神經元(J=20)的架構，如圖1所示。

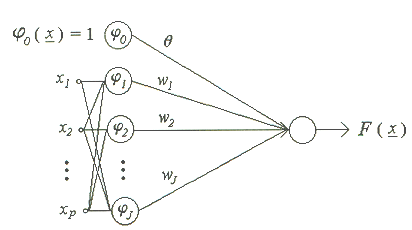


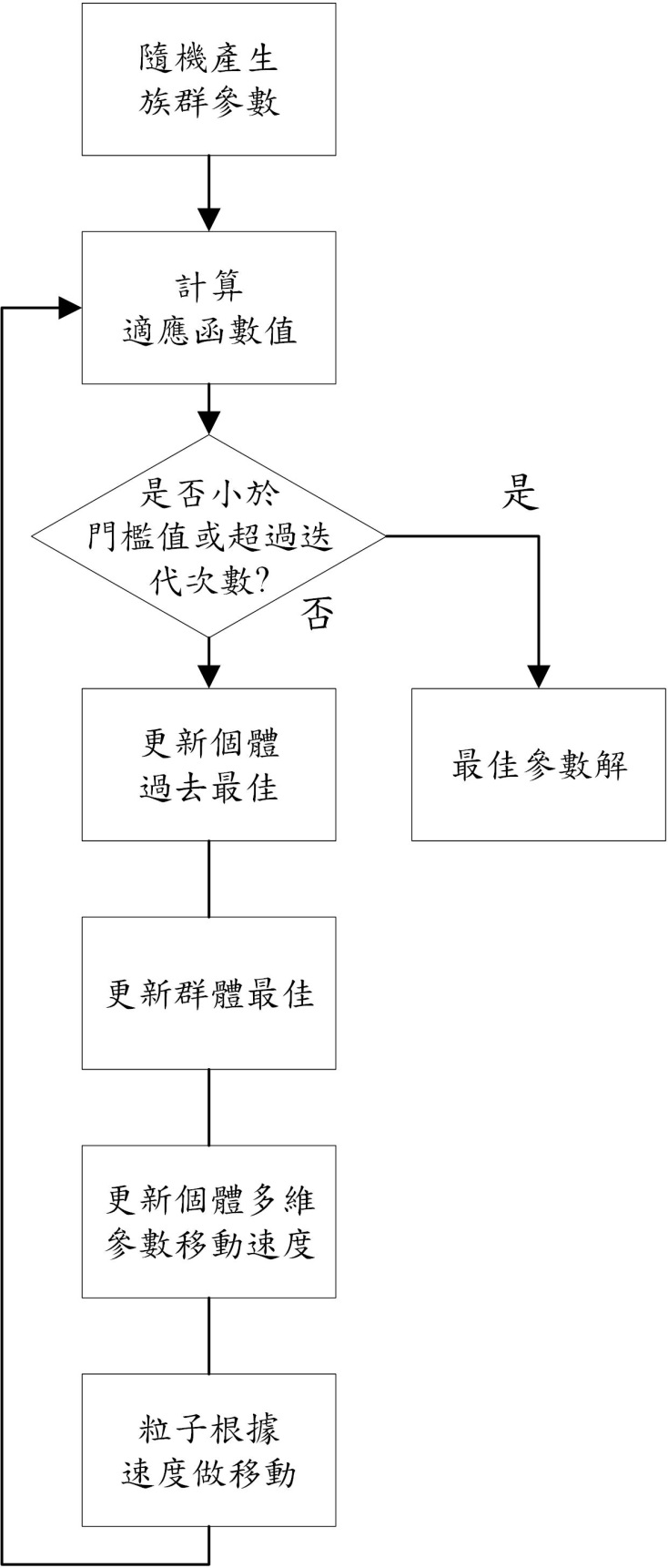
圖1 神經網路架構

各參數範圍如表2所示：

表2 參數範圍

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 參數項目 | x | m |  |  | w、 |
| 參數數值範圍 | 0~30 | 0~30 | 0~1 | 0~10 | -40~40 |

PSO演算法流程如圖2所示。



訓練出想要的最佳解或迭代結束

個體的每個參數速度，根據過去慣性、個體過去最佳解和群體最佳解做更新，並加入一個大小-0.05~0.05的隨機雜訊，使得陷入區域最佳解後仍可以做部分探索，增加收斂速度。

比較現有群體解和過去群體最佳解，做更新

比較個體解和過去個體最佳解，做更新



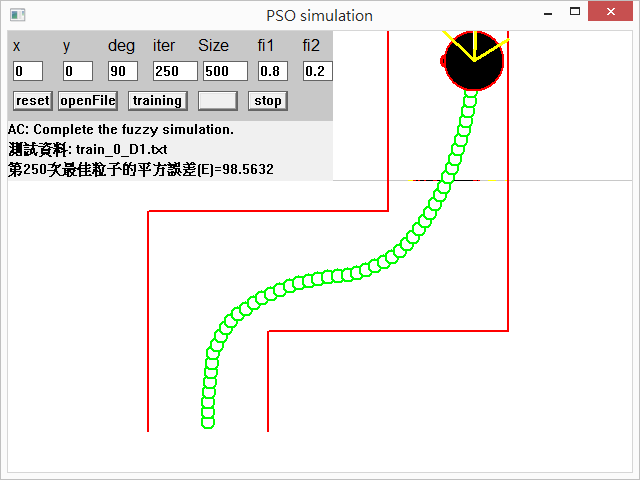


隨機產生粒子群參數，包含高斯函數平均值m、高斯函數標準差、每顆神經元權重w、第一顆神經元權重，以及每個參數的速度，其範圍介於-1~1。

圖2 PSO演化流程

**3.如何使用本視窗程式?**

程式的執行結果如圖3所示，左上角提供參數設定，依序為起始點座標設定(x, y)、起始角度、迭代次數、基因數量、交配機率、突變機率。



動作按鈕

參數設定

目前狀態和過程中的輸入參數

訓練、測試檔案

適應函數值

圖3 程式結果

執行順序：(1)→(2)→(3)

(1)openFile：選擇訓練資料。

(2)training：依照目前設定參數檔進行訓練。

(3)start：依照訓練結果啟動自走車模擬。

(4)stop：暫停自走車模擬。

(5)reset：讓自走車回到起點，但保留訓練結果。

**4.實驗參數對結果的影響**

**(1)迭代次數**(藍色為成功走完，紅色為非成功走完)

基因數量 = 500， = 0.8， = 0.2，使用資料集train\_0\_D1.txt

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 第幾次  迭代次數 | 1 | 2 | 3 | 4 | 5 | 6 |
| 450  (約44s) | 48.99 | 45.12 | 53.68 | 52.30 | 46.42 | 54.21 |
| 350  (約35s) | 42.34 | 98.75 | 56.10 | 38.38 | 61.03 | 44.82 |
| 250  (約24s) | 98.56 | 52.54 | 92.01 | 40.67 | 39.26 | 46.55 |
| 150  (約14秒) | 49.26 | 61.81 | 69.67 | 47.31 | 62.89 | 47.53 |
| 50  (約5秒) | 70.27 | 104.01 | 92.13 | 93.26 | 133.48 | 69.4 |

※大約在150次迭代次數有較佳、較穩定的結果。

**(2)族群大小**

迭代次數 = 150， = 0.8， = 0.2，使用資料集train\_0\_D1.txt

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 第幾次  族群大小 | 1 | 2 | 3 | 4 | 5 | 6 |
| 700  (約花費22秒) | 53.61 | 37.63 | 34.62 | 54.49 | 64.23 | 31.17 |
| 500  (約花費15秒) | 43.54 | 45.72 | 66.02 | 33.34 | 52.33 | 54.63 |
| 300  (約花費9秒) | 111.12 | 83.33 | 68.87 | 48.97 | 118.81 | 74.48 |
| 200  (約花費6秒) | 145.83 | 74.35 | 172.62 | 68.09 | 167.86 | 79.89 |
| 100  (約花費3秒) | 65.83 | 75.62 | 60.98 | 195.49 | 82.00 | 129.77 |

※迭代次數在150次時，族群次數約500次數值開始趨於穩定。

**(3)個體過去最佳參考係數**

迭代次數 = 150，族群大小 = 500， = 0.2，使用資料集train\_0\_D1.txt

圖4是使用不同個體過去最佳參考係數的收斂結果(橫軸為第幾代，縱軸為該代適應函數最小值)。

圖 4不同個體過去最佳參考係數比較

**(4)群體過去最佳參考係數**

迭代次數 = 150，族群大小 = 500， = 0.8，使用資料集train\_0\_D1.txt

圖5是使用不同群體過去最佳參考係數的收斂結果(橫軸為第幾代，縱軸為該代適應函數最小值)。

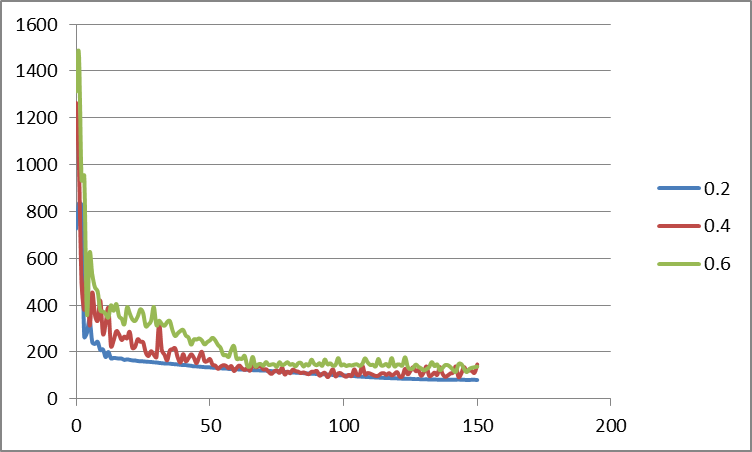


圖5 不同群體過去最佳參考係數比較

**(5)各種訓練資料訓練結果**

以迭代次數 = 150，族群大小 = 500， = 0.8， = 0.2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 第幾次  檔案 | 1 | 2 | 3 | 4 | 5 | 6 |
| train\_0\_D1 | 43.54 | 45.72 | 66.02 | 33.34 | 52.33 | 54.63 |
| train\_1\_D1 | 56.02 | 70.49 | 42.36 | 52.90 | 106.76 | 99.55 |
| train\_2\_D1 | 82.35 | 96.61 | 42.02 | 73.94 | 62.70 | 116.01 |
| train\_3\_D1 | 50.43 | 65.10 | 73.06 | 60.37 | 60.37 | 76.34 |
| train\_4\_D1 | 58.64 | 74.50 | 71.36 | 58.29 | 64.21 | 69.52 |
| train\_5\_D1 | 89.10 | 40.75 | 68.50 | 68.54 | 82.98 | 82.37 |
| train\_6\_D1 | 82.04 | 59.69 | 95.18 | 23.17 | 76.50 | 53.25 |
| train\_7\_D1 | 83.71 | 43.24 | 70.11 | 54.22 | 90.19 | 119.15 |
| train\_8\_D1 | 103.50 | 75.97 | 55.57 | 46.04 | 81.80 | 69.00 |
| train\_9\_D1 | 39.80 | 49.45 | 82.71 | 55.76 | 79.25 | 53.13 |
| train\_10\_D1 | 65.65 | 97.27 | 56.03 | 32.88 | 65.38 | 48.10 |
| train\_11\_D1 | 49.71 | 49.71 | 51.47 | 62.23 | 69.62 | 35.65 |
| train\_12\_D1 | 86.13 | 103.34 | 65.70 | 85.78 | 104.45 | 54.72 |

**(5)不同起始位置對訓練出的解影響**

train\_0\_D1以迭代次數 = 150，族群大小 = 500， = 0.8， = 0.2

|  |  |
| --- | --- |
| (x, y) | 結果 |
| (0, 0) |  |
| (2, 4) |  |
| (0, 8) |  |

**5.實驗遇到的問題**

(1)適應函數值收斂慢

在粒子移動時加入隨機雜訊，嘗試最佳解附近的解，比較不會耗費較多的迭代次數陷入某個區域最佳解而無法跳脫。

(2)是否每一個維度的參數都需要一個速度變數?

不同維度的解可能散布在不同數值，需要不同速度去做逼近。

**6.心路歷程**

PSO和GA的使用上，PSO的概念和實作上都較為GA容易，平均來說，PSO收斂找出最佳解的速度較GA為優異，但須額外花費記憶體去儲存粒子在多維度的速度和個體最佳解、群體最佳解的參數，讓我理解到演算法不一定要複雜或變化性大，能夠有較好的效能和速度的演算法，才是解決問題的不二首選。