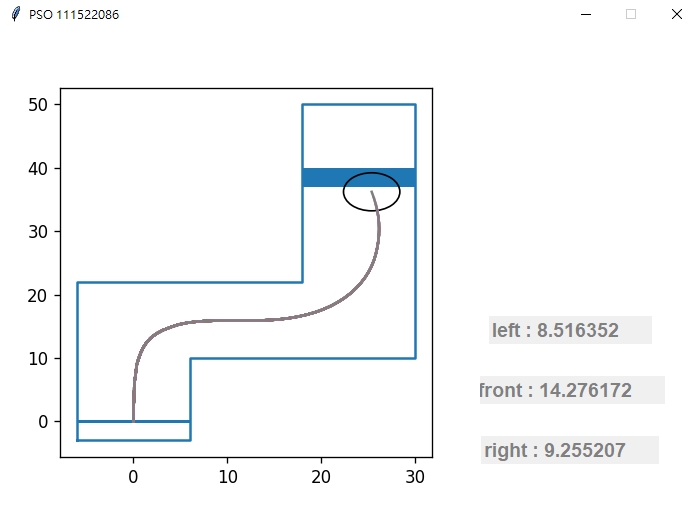
計算型智慧HW3 – PSO

111522086 林思婷

1. 程式介面說明



藍色細線為軌道的牆壁，黑色圓圈為自走車，起點置於軌道座標(0,0)的位置，終點線位於藍色粗線位置處。程式執行後會自動以動畫顯示自走車每一步的位置並在介面右側顯示左/右前方45度角、正前方等三個測量距離sensor所測量到的距離。

此次作業共有六份python檔，分別為：

1. psoParameter (PSO實作細節)
2. generatePSO
3. pso (particle初始化、管理和設置，計算適應值)
4. RBF
5. geometry
6. playground

前4份檔案利用基因演算法訓練RBFN的網路參數

執行2.檔案，讀進train4D.txt檔等ground truth進行訓練，訓練完後得一組網路參數，實作的細節將在第三節中詳細說明。

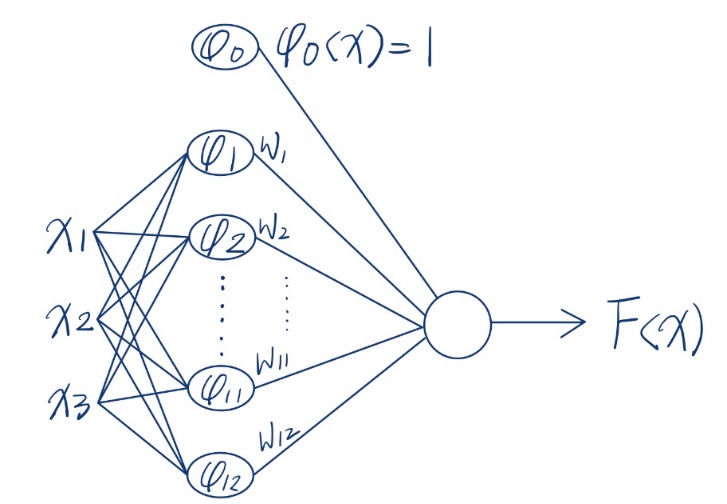
後4份檔案將sensor偵測所得距離輸入RBFN並輸出方向盤角度

1. 實驗結果

如一、圖所示，在移動的過程中均能根據sensor回傳距離改變方向盤的角度，以避免觸碰軌道牆壁，順利抵達終點線。

1. 基因演算法實作細節

此次作業採用實數型基因演算法，下圖為RBFN





此RBFN選用高斯型基底函數

我們需將RBFN的所有可訓練的參數作為一個粒子(particle)的內容

故此粒子的位置和速度皆為1 + J + J \* xDim + J維度，如下所示：, (此次作業J=12、xDim=3)

各維度的初始值範圍：

,

(分母不可為0)

適應函數為：



N：作業1產生的N筆成功到達目的訓練資料

：表示訓練資料的方向盤期望輸出值

目標即利用PSO找出一個最佳的particle(其適應值越小越好)

以下詳述PSO實作細節

* PoolSize (族群數): 128
* MaxIteration: 30
* phi1: 0.4、phi2: 0.6

起初隨機產生初始族群(初始位置依據不同維度給予相對應範圍的值)

可參考上頁各維度的初始值範圍

Step1.計算各個粒子的適應函數值(此次作業適應值應越小越好)

Step2.所有粒子依據適應值作排序

Step3.記錄(自己過去找到最好的)和

Step4.根據下方公式計算下一個時間的速度的位置，

重複上述步驟直到迭代完畢(此次作業設定迭代30次)

1. 分析

這次有改變RBFN隱藏層神經元的數量，從原本3個增加至12個，因為沒有修過類神經網路，所以沒有想到增加神經元的數量可以提升網路架構輸入與輸出之間映射關係的能力(較高維度的隱藏層空間可以有較為準確的預測值)。一開始以3個神經元大概訓練10次僅成功抵達終點一次，增加至12個神經元後訓練5次均成功抵達。

也嘗試過加入之前所讀的論文使用的方法(加入SA和DLS)

[A Novel Hybrid Particle Swarm Optimization Algorithm for Path Planning of UAVs](https://ieeexplore.ieee.org/document/)

SA: 一開始讓 能夠接受次優解，避免陷入局部最佳解

DLS: 抽換各維度的值比較適應值，若較好則替換(如下圖)

但此方法將增加可調整參數數量，故後來未採用。

