

## 程式使用說明

### 程式問題

$$x(t+1) = x(t) + \cos[\phi(t) + \theta(t)] + \sin[\theta(t)]\sin[\phi(t)] \quad (10.18)$$

$$y(t+1) = y(t) + \sin[\phi(t) + \theta(t)] - \sin[\theta(t)]\cos[\phi(t)] \quad (10.19)$$

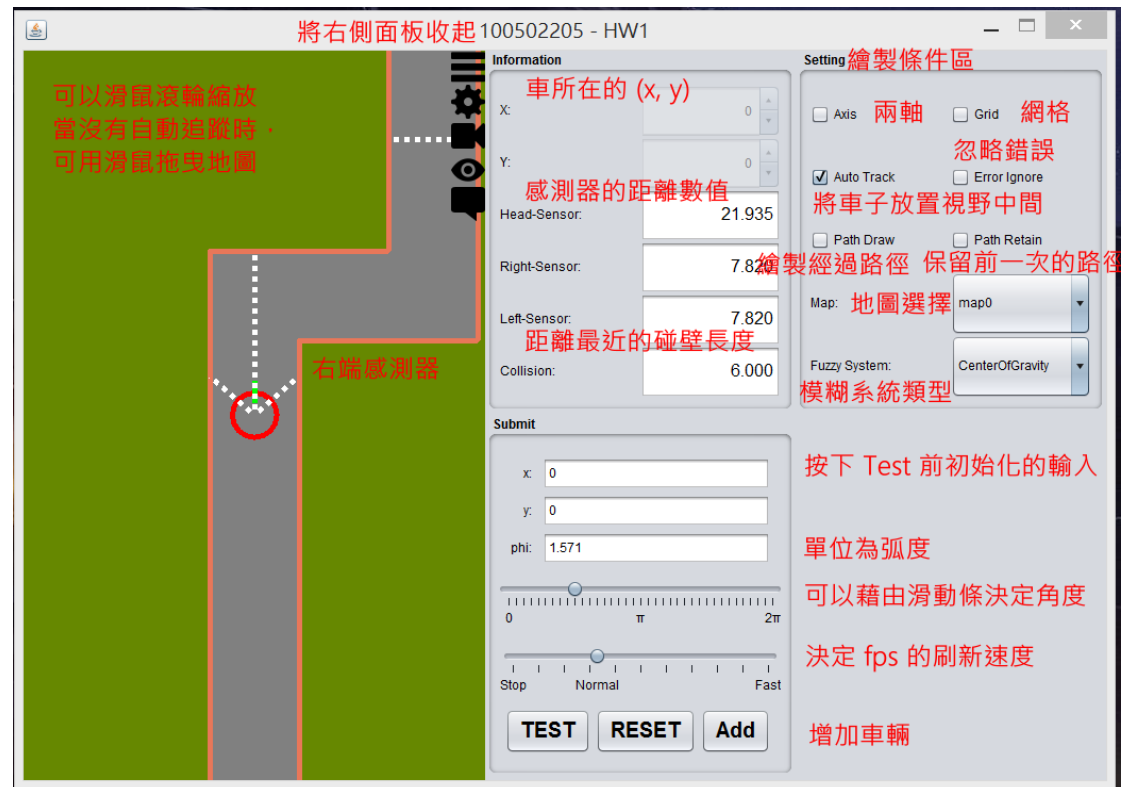
$$\phi(t+1) = \phi(t) - \arcsin\left[\frac{2\sin[\theta(t)]}{b}\right] \quad (10.20)$$

其中  $\phi(t)$  是模型車與水平軸的角度， $b$  是模型車的長度， $x$  與  $y$  是模型車的座標位置， $\theta(t)$  是模型車方向盤所打的角度，其中  $\phi(t)$  是模型車與水平軸的角度， $b$  是模型車的長度， $x$  與  $y$  是模型車的座標位置， $\theta(t)$  是模型車方向盤所打的角度，我們對模擬的輸入輸出變數限制如下：

$$\begin{cases} \phi(t) \in [-90^\circ, 270^\circ] \\ \theta(t) \in [-40^\circ, 40^\circ] \end{cases}$$

寫一個程式使用模糊系統、理論模擬車子行徑，並且想辦法最佳化。

## 程式開啟時的畫面

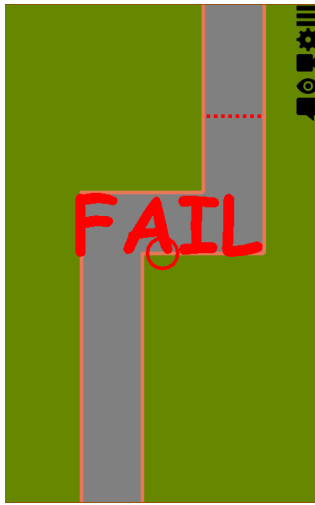


## 執行環境

- JRE 1.7
- 如有發生短暫延遲問題，這將會導致模擬數據誤差，請將 Submit 面板中的 Stop - Normal - Fast 滾動條調至 Normal 數值以下。
- 測試於 Win 8, Mac OS X 執行正常。
- 若有其他問題請 e-mail 聯絡 [morris821028@gmail.com](mailto:morris821028@gmail.com)

## 如何開始第一次模擬

1. 打開 `xxx.jar` 檔案，確認有上圖的顯示畫面後，表示為正常啟動。
2. 在左下角的 **Submit** 面板，將數值打入 `x`, `y`, `phi` 角，接下來按下下方的 **TEST** 按鈕即可開始測試。
3. 過程途中會在畫面中提示是否有碰壁 (以紅色 "FAIL" 表示)，反之當抵達終點時，則會以綠色呈現成功 ("SUCCESS")



4. 在此次作業中，只有 `map0.txt` 設有終點線，其餘地圖皆沒有終點線的設置。右下方則會顯示相關的模糊集合的塞選結果。

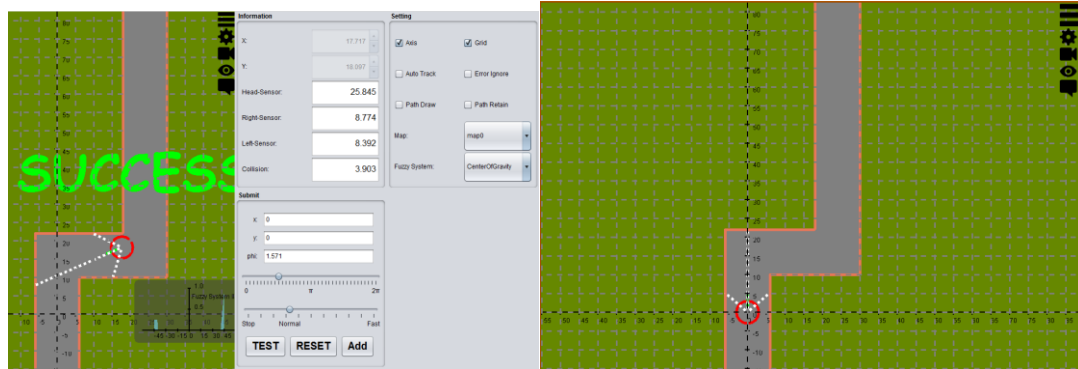
## 更多支持

1. 多台車輛的設計，目前由於多執行緒的處理會動的障礙物上存有疑慮。
2. 可以開啟源代碼進行匯入地圖操作。

地圖撰寫格式：

基底多邊形的點個數  $n$ ，接下來會有  $n$  個浮點數座標值，接著會有一個整數(0/1)表示是否有終點線，若有終點線，接下來會有兩個浮點數座標值。接著將會有數個簡單多邊形的障礙物輸入。

## 程式運行截圖

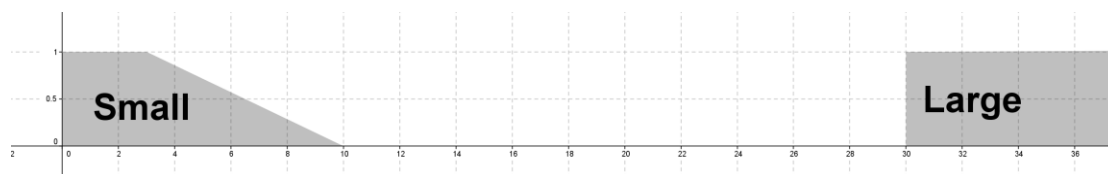


## 函數實驗

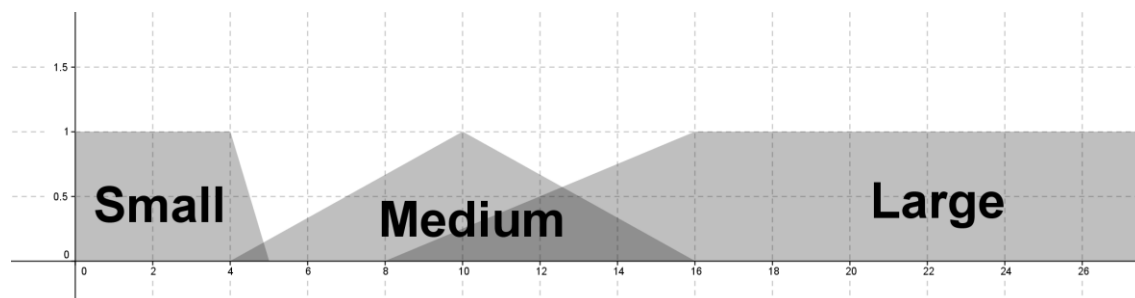
1. 經過實驗，三個感測器分別坐落於前方(90 度)、左右各偏轉 50 度左右的夾角比左右 90 度更好。用以應付多變的轉角彎度，防止模稜兩可的擺動。
2. 不管哪裡種去模糊化系統，由於題目限制擺角於  $[-40, 40]$  度之間，因此很容易在加權平均、質心、離散化而導致邊界擺角的機率甚低，當越多函數則稀釋程度越高。因此函數定義可以超過限制擺角，只需要在去模糊化時，約束至條件內即可。
3. 設計函數時，要討論向其中一個方向偏轉，也就是不能設計過於對稱的圖形，否則將會在死胡同裡打轉。

## 函數設計

1. 前方 d1 感測器的數值  
d1 歸屬函數



d2, d3 歸屬函數



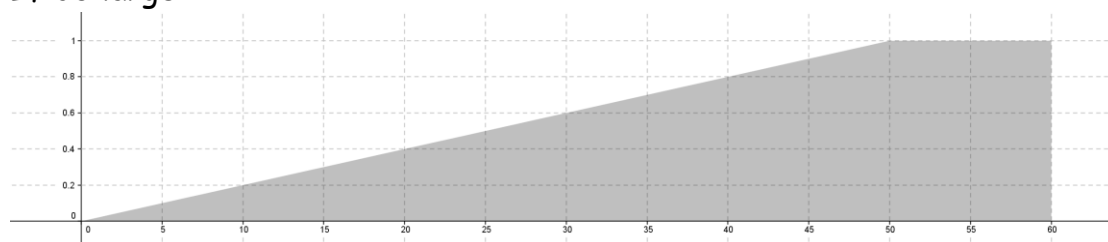
## 2. 函數式加權平均

- If d3 large,  $\theta = 55$
- If d2 large,  $\theta = -55$
- If d2 medium,  $\theta = -40$
- If d3 medium,  $\theta = 40$
- If d1 large and d2 small,  $\theta = -30$
- If d1 large and d3 small,  $\theta = 30$
- If d1 small,  $\theta = -60$
- 規則如上述七條，以最簡單的常數關係。測試結果還算不錯。

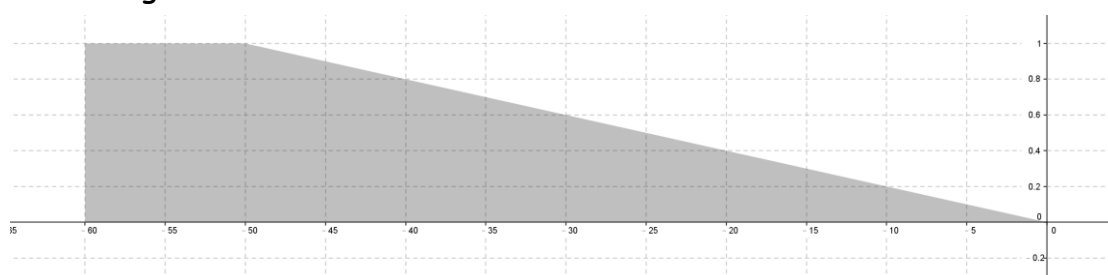
「if condition, then statement」, statement 中的變數不存在於 condition 去做調整，後來發現由於太複雜去討論而一直失敗。

## 3. 重心法、最大平均法、修正型最大平均法、中心平均法

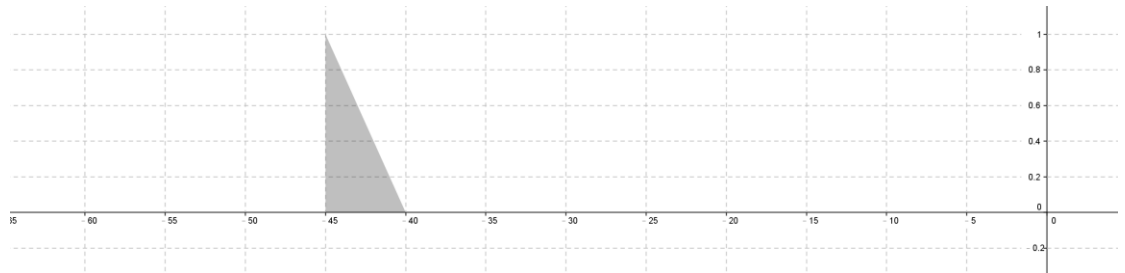
- If d3 large



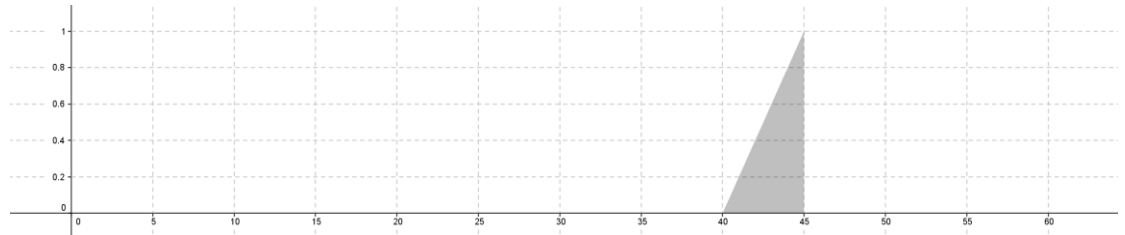
- If d2 large



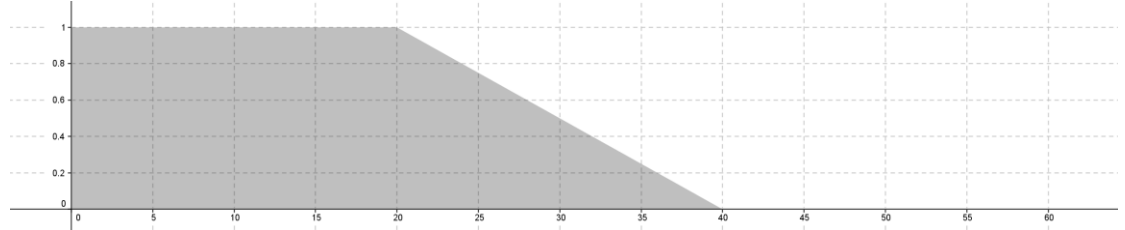
- If d2 medium



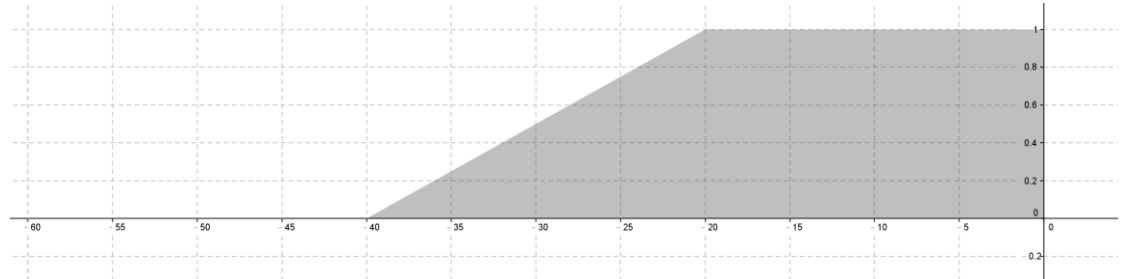
- If d3 medium



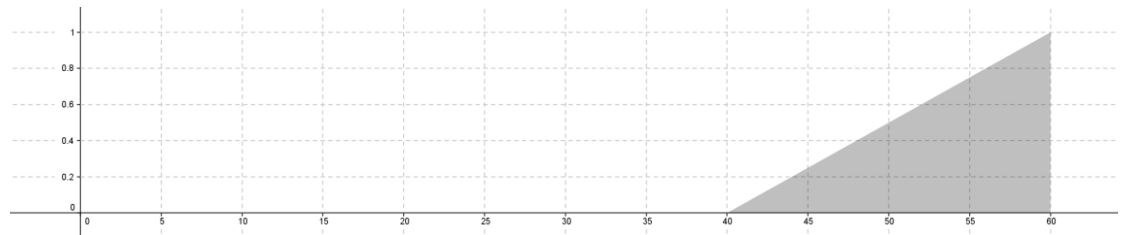
- If d1 large and d2 small



- If d1 large and d3 small



- If d1 small



- 為了方便計算連續函數，採用離散取點，間隔 0.1 抓取  $[-60, 60]$  之間的點座標。對於以下四種方法討論於簡單彎道的過程記錄：

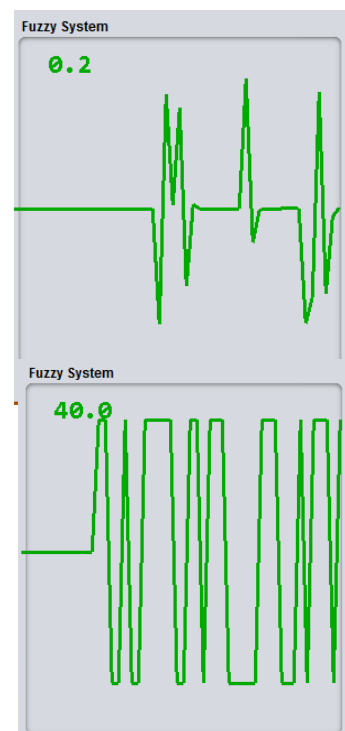


- 重心法

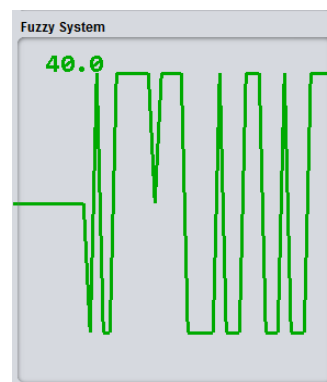
當中表現最為穩健，而且路徑平滑。

- 最大平均法

由於定義的函數較為簡單，而且大多都是以梯形的方式存在，雖然能在機率高的情況下走完全程，路徑中會呈現左右擺盪情況。並且大多都已極值的方式出現。



- 修正型最大平均法  
情況會稍微好一些，但是對於並沒有向重心法一樣的圓滑曲線，仍是因為原先的函數設計所惹的問題。



對於某一種方法而去定義相關的函數進行測較好，而在最大平均法的部分，必須使用較具有變化性的曲線，否則很容易在極值狀況下擺動。