

模糊理論應用於年輕男女配對系統之研究

劉仲鑫 黃俊程

中國文化大學資訊工程系

11114 臺北市陽明山華岡路 55 號

liu3.gold@msa.hinet.net

摘要

許多年輕男女喜歡玩男女配對之類的遊戲，本文主要透過定位裝置、地理位置服務、NS2 模擬路徑和模糊理論，來達到能夠在現實中配對的方法。本研究透過全球衛星定位系統來提供風雨無阻的的定位，用位置服務來提供地理位置和可以行進的方向，再者透過 NS2 來模擬使用者的行走速度和方向，用模糊理論來為人們配對，來達到延伸性的定位和配對服務。

關鍵詞：NS2，模糊理論，全球衛星定位系統

The Study of Using Fuzzy Set in Young Men and Women Matching System

CHUNG-HSIN LIU and CHUN-CHENG HUANG

Department of Computer Science and Information Engineering, Chinese Culture University

55, Hwa-Kang Rd., Yang-Ming-Shan, Taipei, Taiwan 11114, R. O. C.

liu3.gold@msa.hinet.net

ABSTRACT

Numerous young men and women enjoy playing the matching game. This study employs a global positioning service (GPS), location services, an NS2 simulator, and fuzzy theory to create a real-world matching method. We apply GPS to provide location information, and obtain the user's geographical location and moving direction using the developed system. Then, we use the NS2 simulator to estimate the user's walking speed and direction. Finally, we exploit fuzzy theory to determine the user's matches, thereby increasing the positioning and matching services available.

Key Words: NS2, fuzzy theory, global satellite positioning system

一、前言

近年來，許多交通工具、行人和各地地標的定位和導航的需求大增，讓原本用於軍事、地理測量和航空導航用的全球衛星定位系統（global position system, GPS）普及於一般民間使用。位置服務（location based service, LBS），是一個透過有線或無線技術的結合，來提供具有時效性與地域性的資訊內容。綜合以上兩者就能夠提供由 GPS 來定位和由 LBS 顯示地理資訊的延伸定位服務。

網路上有一種男女配對的遊戲，主要是透過輸入兩人的姓名、星座和血型...等資料，再提供各種關係暗示的回答來滿足好奇心旺盛的男女。但此遊戲只能在遠端或同一地點上。因此，我們提供在現實中能夠搜尋年輕男女的一套機制，主要是讓登記在案的年輕男女透過 GPS 和 LBS 的顯示在地圖上，接著我們幫他們配對，再發簡訊讓他們依循著我們的指示找對方來增加趣味性，如果不滿意配對時，可以發出 SOS 訊號，我們會為他或她重新配對直到滿意為止。

二、文獻探討

（一）全球衛星定位系統

全球衛星定位系統（global position system, GPS）[5]，其應用在導航定位，另外任何人都可取得精確的三度空間位置、速度和時間。原由美國國防因戰略需要所開發，隨著戰爭結束已將一部分開放給民間使用於定位、追蹤和導航等相關作業上。

GPS，一般來說是由空間衛星、地面監控和用戶接收端所組成，第一個是空間衛星，由 24 顆在地球上空高度約兩萬公里的衛星所組成。每顆衛星用近似圓形的軌道以 11 小時 58 分鐘繞地球一次，共有六個軌道，一個軌道有六顆衛星，軌道間隔 60 度，和赤道間隔 55 度之傾角。第二是地面監控部分，又分成地面監控站（master control station）、地面天線（ground antenna）和監測站（monitoring station），由監測站向地面天線索取得的資料傳送給主控站統整分析。第三是用戶接收端，負責接收 GPS 衛星的訊號，分析獲得的導航核定位置資訊，處理並完成導航核定工作。

GPS 定位原理基本上只要三顆衛星就能夠定位了，但因為精度不好，用四顆衛星較好。三顆衛星下，使用空間距離交會（spatial resection）法來確定位置，即由衛星發訊號到 GPS 上所花的時間來確定距離，再依據此距離為三顆衛星畫出三個空間球體，再取它們的交點，並排除一個不在地表上

的，就是接收位置了。

GPS 精度依其標準可分為標準定位精度（standard position system, SPS）和精密定位精度（precise position system, PPS）兩種。選擇可用性（selective availability, SA）開啟時，SPS 誤差達 100 公尺，當 SA 關閉時約 30 公尺。基於精度的提升，可用差分定位（differential GPS, DGPS）法來校正，誤差就下降到二到五公尺。若是 PPS 則要配合 P(Y) 碼來計算，但 P(Y) 碼不易取得，所以大部分都以 SPS 配合 DGPS 使用。

GPS 與位置服務使用 GPS 的位置服務（LBS），指的是使用者依 GPS 提供得地理位置資訊來做擴充的服務。LBS 主要包括四大單元，分別為行動裝置、定位系統、無線網路、網際網路（geographic information system, GIS），行動裝置指的是具有運能力的支行動式終端設備（如手機...等），定位系統多以 GPS 為主，無線網路由現在的電信業者提供，網際網路 GIS 多以 Google Maps API 為主。

TWD67 為早期台灣以虎仔山為基準用三角測量計算所得來的座標系統，而且只適用於台灣附近區域。TWD97 是用 GPS 來取得地表的經緯度，適用於全球任何一個地方，為目前台灣所用之座標系統。WGS84 為目前世界上通用的座標系統。

（二）網路模擬器第二版

本研究使用網路模擬器第二版（network simulator version 2, NS2）網路模擬工具，NS2 是一個網路開放的免費工具，目前包含的模組都是由全球各地不同的人所共同開發出來，本研究使用之無線個人網路模組是由紐約大學電機系所開發完成，內容除了無線電波之發送用 NS2 現有工具模擬以外，其他通訊協定部份皆完全符合 IEEE 802.15.1，如圖 1 為系統圖（NS2）[2-3, 8]。

NS2 網路模擬器中的無線網路模型最初是由 CMU's Monarch group 所研究發展而來，僅提供模擬單純的獨立型無線網路（ad-hoc networks），接著再以此為基礎進而延伸出中控型無線網路（wired and wireless networks）及 Mobile IP 等網路架構，後面將介紹由 CMU's Monarch group 所發展的獨立型無線網路架構及其延伸後的中控型無線網路 [7]。

NS2 是一套物件導向的網路模擬器，由 Berkely 所開發完成。是 1989 年改版自 REAL network simulator 模擬器，並由 VINT project（UCB）組織所開發，目前由 SAMAN 和 CONSER 負責維護能模擬真實網路系統的架構和特性。

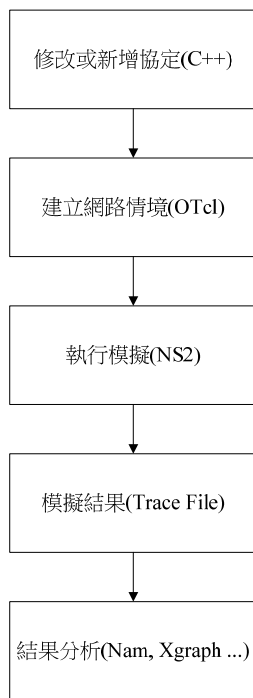


圖 1. 系統圖 (NS2)

由於 NS 是一套免費的 Open Source 軟體，在學術研究方面具有公信力，並且新增修改協定方便。所以我們採用 NS2 作為本研究的模擬器。在網路架構的方面，可以模擬路由器、link、網路上的節點等等；網路特性方面有封包的延遲或封包的丟棄等模擬，圖 2 為 NS2 模擬器架構圖 [1]。

當 NS2 中的 node 接收到一個 routing packet，node 會呼叫 routing agent 的“recv”函式去接收 packet。第二是“command”，這個函式是 TCL command line 呼叫使用的，我們可以在 TCL script 中啟動一個 hello timer 或是給一個像 interface queue 的 class。C++ 寫程式碼，OTCL 寫負責溝通的介面程式碼供 TCL 語言使用。

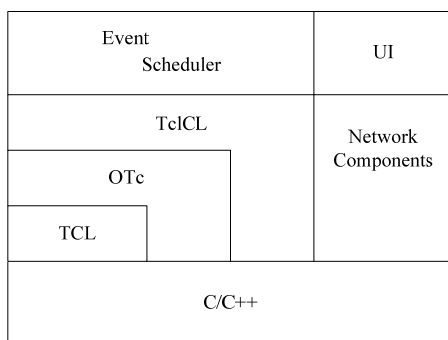


圖 2. NS2 模擬器架構圖

(三) 模糊理論

模糊理論是加州柏克萊大學的教授 Zadeh 於 1965 年所提出的，將人類思考和推理的不確定性以數學模式表示。1980 年代，企業界的研究人員也開始重視模糊理論的相關研究，並逐漸進入實用的階段，自此後模糊理論便迅速的發展起來。在 1984 年的時，成立了國際性的模糊理論的系統學會 (International Fuzzy Systems Association, 簡稱 IFSA)，模糊理論系統學會 [IFSA] 從 1985 年起每隔一年便會在美國、西班牙和日本等地舉辦國際性的模糊理論研討會。模糊理論在日本和中國大陸的發展卻十分興盛，尤其在日本，企業界普遍將模糊理論融入產品之中 [3]。

Fuzzy 在語義上有不明確的意思。傳統的二值邏輯 (binary logic) 只有包含對 (true) 跟錯 (false) 兩種，但是在人類的思考和語意的表達上並非只有對、錯兩種。每個人對這些語意上模糊的定義都不盡相同。所以我們無法明確地判斷這些語意所要表達的意思，這不是二值邏輯所能表達的。在 Fuzzy 中將二值邏輯擴展成連續多值邏輯 (continuous multi-value logic)，依照這些大概的資訊對人類的概念作一個大概的定量化處理。

在 Fuzzy 中我們利用歸屬函數 (membership function) 值來表示一個概念，使用 0 和 1 之間的數值來表示一個元素屬於哪一個概念的等級，這個數稱為該元素對於此集合的歸屬度 (membership grade)，圖 3 為模糊控制系統圖 [4]。

模糊控制系統主要分成：模糊化介面、Fuzzy 知識庫、決策邏輯和解模糊化介面四個部份。其中，模糊化介面 (fuzzification interface) 負責接收資料擷取界面量測的值。接著進行量化工作，以便觀測量的範圍轉換到語言變量對等的論域。最後，將輸入資料轉換成可被模糊集合表現的語言數值。Fuzzy 知識庫，即為資料庫 (data base)，主要是提供語言變數所需要的定義 (例如：變數論域、語言項子集合及歸屬函數的規劃等，以方便供給語言控制規則和 FLC 中的模糊資料管理)。而語言控制規則庫 (rule base) 利用各種方

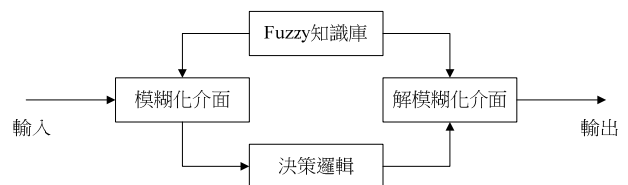


圖 3. 模糊控制系統圖

法定義出來的一組語言控制規則，用來描述控制目標和領域專家的控制策略。解模糊化介面 (defuzzification interface)，主要是進行等級對應的工作，以便將輸出變數值的範圍轉換到對等的論域。解模糊化，由推論出來的控制動作，產生出實際明確的控制動作。決策邏輯 (decision making logic)，就是 Fuzzy 推論機構，它具有模擬人類做決策判斷的能力。這是根據人類特有的近似推理方式，延伸的 Fuzzy 推論技術實作出來的計算機構 (computation unit)。

模糊推理技術中最常用的方法是 Mamdani 方法。他應用了一套有經驗的人類操作員提供的模糊規則。Mamdani-style 模糊推理過程按四個步驟執行：輸入變數的模糊化、規則評估、聚合規則的輸出和最終的逆模糊化，其完整流程如圖 4 完整流程圖 [4]，圖 5 為清晰集與模糊集，縱向為隸屬度，橫向為公分 [4]。

說明流程前必須先了解模糊集，圖 5 中的 (a) 清晰集和 (b) 模糊集的分別之處，在於清晰集有明確的在訂出標

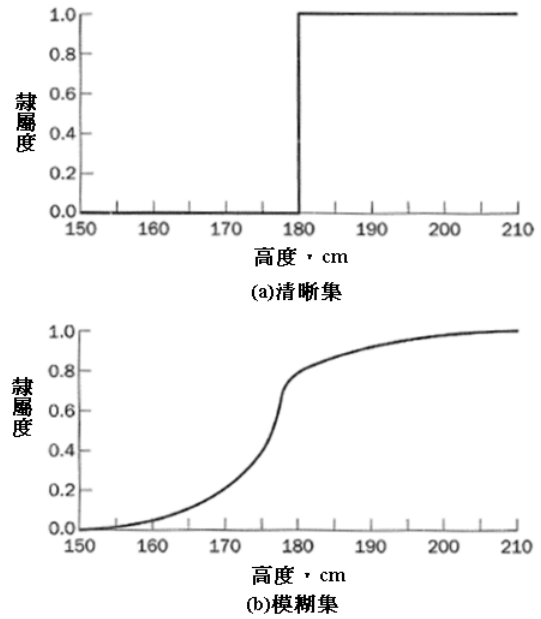


圖 5. 清晰集與模糊集

準，以圖 5 的清晰集為例，180 公分以上皆屬於集合，180 公分以下皆不屬於集合。模糊集則沒有明確的定出標準，以圖 5 的 160 公分為例，圖 5 的 160 公分的隸屬度在 0.1 多，表示 160 公分在這個集合占有 0.1 的可能性。接著開始說明流程，圖 6 為模糊化 [4]。

首先是模糊化，目的在於把清晰的外界輸入 x_1 和 y_1 ，表示成模糊集所能表式的清晰數值，例如： x_1 在 A_1 的隸屬度是 0.5，在 A_2 為 0.2，在 A_3 為 0， x_1 有 0.5 的可能性屬於模糊集 A_1 、有 0.2 的可能性屬於模糊集 A_2 和完全不屬於模糊集 A_3 。 y_1 也是依此判斷。

取得模糊化後的清晰數值，就可以執行規則評估，在規則評估前要先介紹 AND 和 OR 運算，如公式 (1) 所示 OR 運算。

$$\mu_{A \cup B}(x) = \text{probor}[\mu_A(x), \mu_B(x)] = \mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x) \quad (1)$$

OR 運算是取各集合中的所有的部分並扣除多加的部分，在模糊運算中就是直接取各集合中最大的隸屬度。接著介紹 AND 運算，如公式 (2)。

$$\mu_{A \cap B}(x) = \text{prod}[\mu_A(x), \mu_B(x)] = \mu_A(x) \times \mu_B(x) \quad (2)$$

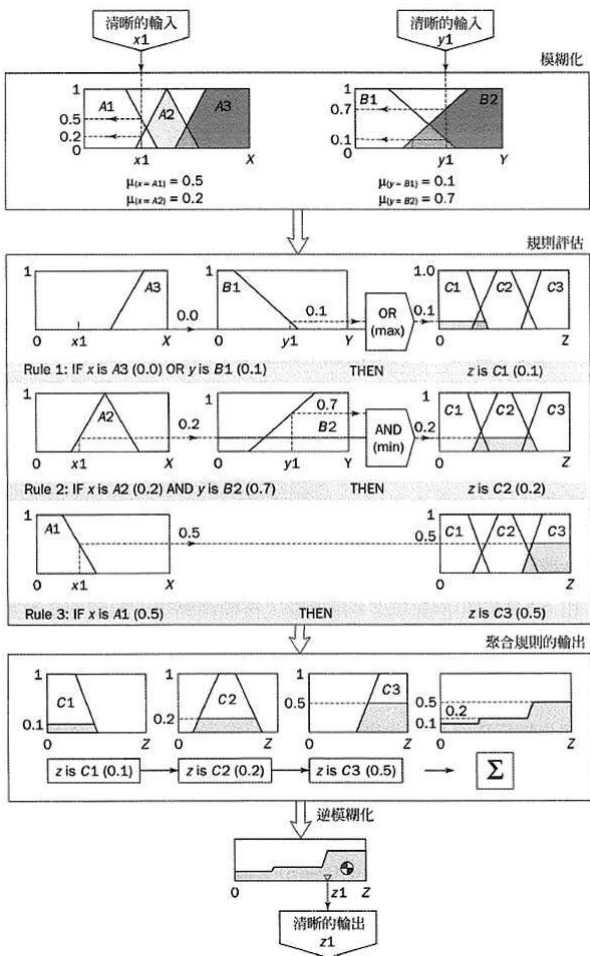


圖 4. 完整流程圖

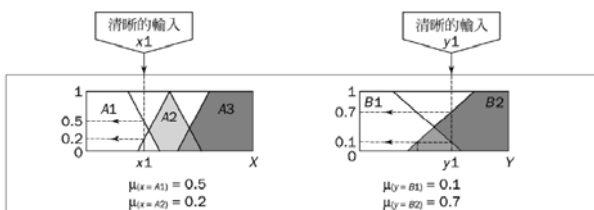


圖 6. 模糊化

AND 運算是取各集合中的共有的部分，在模糊運算中就是直接取各集中最小的隸屬度。之後就可以進行規則評估如圖 7 [4]。

取得模糊化後的清晰數值，接著再依據自己訂的規則取得輸出模糊集的隸屬度。目的在於提供電腦具有類似人類的判斷能力，如：

規則 1：如果 x 屬於 A3 或 y 屬於 B1，那麼 z（輸出值）就屬於 C1，而 z 在 C1 中的隸屬度由 x 在 A3 中的隸屬度 (0.0) 和 y 在 B1 中的隸屬度 (0.1) 來決定，因為作 OR 運算，所以取兩者中最大的隸屬度 (0.1) 作為 C1（輸出值）的隸屬度。

規則 2：如果 x 屬於 A2 且 y 屬於 B2，那麼 z（輸出值）就屬於 C2，而 z 在 C2 中的隸屬度由 x 在 A2 中的隸屬度 (0.2) 和 y 在 B2 中的隸屬度 (0.7) 來決定，因為作 AND 運算，所以取兩者中最小的隸屬度 (0.2) 作為 C2（輸出值）的隸屬度。

規則 3：如果 x 屬於 A1，那麼 z（輸出值）就屬於 C3，而 z 在 C3 中的隸屬度由 x 在 A1 中的隸屬度 (0.5) 來決定，因為沒作運算，所以直接取 z 在 C3 隸屬度 (0.5) 作為 C3（輸出值）的隸屬度。

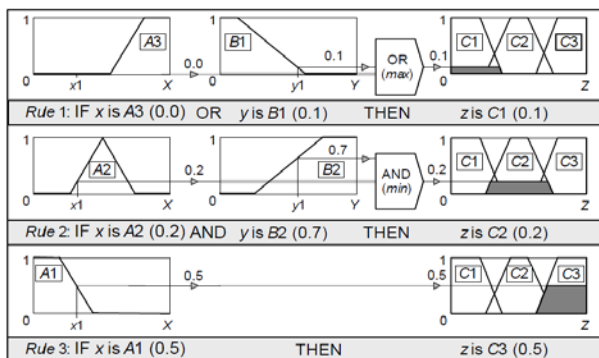


圖 7. 規則評估

這些數值將會在下一個階段使用，也就是聚合規則的輸出，圖 8 為聚合規則輸出 [4]。

聚合規則的輸出是把所有規則輸出進行單一化的過程。我們取之前所有規則的隸屬函數並將它們合併到一個模糊集中，以圖 8 為例，把之前的三個規則輸出的模糊集 C1、C2 和 C3 合併成一個新的模糊集，並保留各自的隸屬度和所對應在輸出等級的數值。目的在於方便接下來的逆模糊化運算如圖 9 [4]。

模糊推論過程中的最後一個步驟是逆模糊化。模糊化可以幫助我們評估規則，但是模糊系統的最終輸出必需是一個清晰的數值。逆模糊化過程的輸入是聚合規則輸出後的模糊集，且輸出是單一的數值 [6]。

有幾種逆模糊化的方法，但最常用的是質心技術。這種技術尋找一個點，這個點所在的垂直線能夠將聚合集分割成兩個相等的部份。這個重力的質心 (COG) 的數學表示如公式 (3)。

$$COG = \frac{\int_a^b \mu_A(x) \cdot x dx}{\int_a^b \mu_A(x) dx} \quad (3)$$

質心逆模糊化方法在 ab 區間找到表示模糊集 A 的重力質心的點，可以得到 COG 的合理估值，其算法如圖 10 之 COG 算法圖 [4]，縱向為隸屬度，橫向為輸出數值。

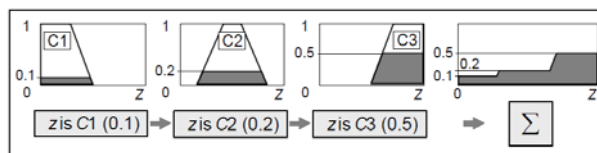


圖 8. 聚合規則輸出

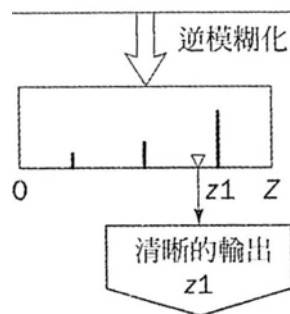


圖 9. 逆模糊化運算

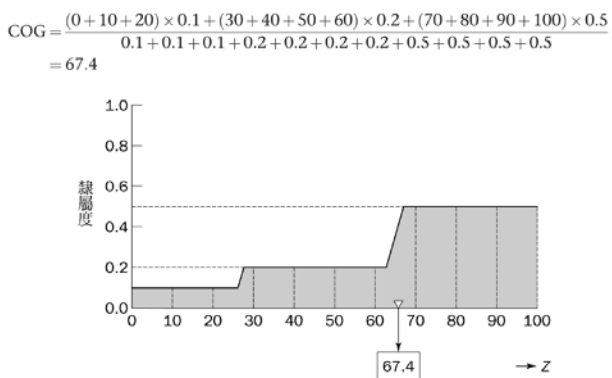


圖 10. 算法圖

依據先前定義的輸出數值乘上對應的隸屬度，接著再相加，其後再把以相對應的隸屬度相加，最後把前者除以後者，即可得到一個清晰的輸出數值。以圖 10 為例：因為先前已將 z（輸出）的數值從 0 分佈到 100，所以只要依據橫向的數值乘上相對應的隸屬度（如：因為輸出數值 0、10 和 20 都在隸屬度 0.1 的範圍內，30、40、50 和 60 在隸屬度 0.2 的範圍內，所以 0、10 和 20 都要乘上 0.1，所以 30、40、50 和 60 都要乘上 0.2，其他輸出數值亦是如此），再除以對縱向對應到的隸屬度（如：因為輸出數值 0、10 和 20 都在隸屬度 0.1 的範圍內，30、40、50 和 60 在隸屬度 0.2 的範圍內，所以除數會有 3 個 0.1 和 4 個 0.2...等）。

(四) 活動圖與循序圖

循序圖表示一組物件在系統執行期間的交互作用，以時間為著重點的一連串處理程序，強調事件發展的先後順序，以其決定訊息傳遞的先後順序。

活動圖很像流程圖，可以把它看成是流程圖的一種推廣。除了可以從高層次的觀點來塑模企業處理，活動圖也可以用來捕捉使用案例的執行細節。從更微觀的角度來說，活動圖也可以用來塑模一個複雜運算的內部邏輯。為了因應日趨複雜的企業處理工作流程，藉此以提供更豐富，更有彈性的語意來加強活動圖的能力。圖 11 為流程循序圖。

首先，使用者必需是先登記在案的人才能使用，主要是為了確認身分的真實性。登記在案的使用者可以透過具有 GPS 功能的智慧型手機或者之類具有定位和上網功能之裝置來使用我們提供的服務，使用者必需先驗證身分才能使用我們提供的服務。如果身分正確的話，即可進入到登入畫面。若驗證失敗的話，就會進入到登入畫面，在此畫面下使用者無法使用我們提供的任何服務。圖 12 為登入活動圖。

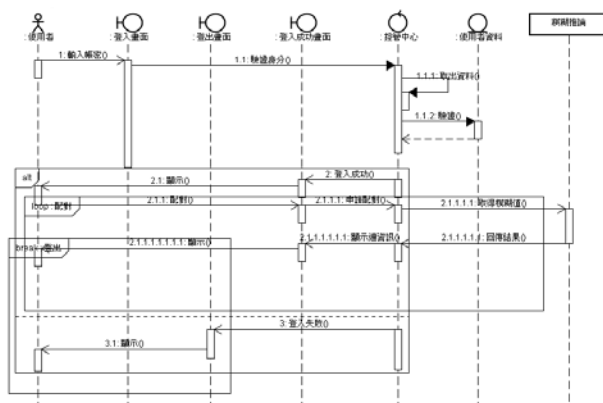


圖 11. 流程循序圖

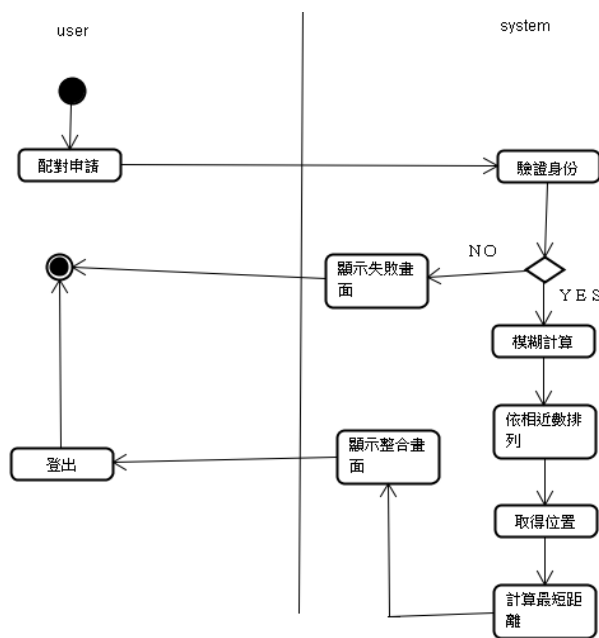


圖 12. 登入活動圖

在登入成功畫面下，若使用者提出配對的要求時，就會執行模糊推論，主要是控制中心會接收到使用者輸入的身高和 BMI 數值，接著把這些數值輸入到模糊計算中，進而得到一個數值，再從模糊列表中依數值相近者來排列順序，然後依位置列表找出和使用者間的距離是合理的配對對象，再把兩者間的距離依能行走的道路做最短規劃，最後會把這些資料送到整合中心，整合中心再把這些資料中可以顯示的部份傳到登入成功畫面中顯示給使用者看。

(五) Floyd-Warshall 演算法

Floyd-Warshall 演算法 (Floyd-Warshall algorithm) 是解決任意兩點間的最短路徑的一種演算法。Floyd-Warshall 演算法的原理是動態規劃。動態規劃是一種用於求解包含重疊

子問題的最優化問題的方法。其基本思想是，將原問題分解為相似的子問題，在求解的過程中通過子問題的解求出原問題的解，其時間複雜度為 $O(N^3)$ ，空間複雜度為 $O(N^2)$ 。演算法 (4) Floyd-Warshall 如下：

```

for k ← 1 to n do
for i ← 1 to n do
for j ← 1 to n do
if (  $D_{i,k} + D_{k,j} < D_{i,j}$  ) then
 $D_{i,j} \leftarrow D_{i,k} + D_{k,j}$ ;
    
```

(4)

設 $D_{i,j,k}$ 為從 i 到 j 的只以 $\{1, k\}$ 集中的節點為中間節點的最短路徑的長度，若最短路徑經過點 k ，則 $D_{i,j,k} = D_{i,k,k} - 1 + D_{k,j,k} - 1$ ；若最短路徑不經過點 k ，則 $D_{i,j,k} = D_{i,j,k} - 1$ 。因此， $D_{i,j,k} = \min(D_{i,k,k} - 1 + D_{k,j,k} - 1, D_{i,j,k} - 1)$ ，簡單來說就是看經過 k 點是否會使長度變短，若有則更新，若無則把舊值輸入到新增的新的變數裡。假設路徑如圖 13 路徑圖。

再依據其路徑長，我們可以得到一個 D^0 矩陣，代表在只能經過自己的情況下到各點的權重，其中 ∞ 表示無法到達。 π^0 矩陣代表只能經過自己的情況下到達各點所要經過的點，其中 null 表無法到達，0 表示自己，如圖 14 過程圖所示。

D^1 表示各點在可以經過點 1 的情況下到達各點所需的權重，因為 $D^0[4, 1]$ (其數值為 2) 加上 $D^0[1, 2]$ (其數值為 3) 比 $D^0[4, 2]$ (其數值為 ∞) 來的小，依據此 $D^1[4, 2]$ 的數值從 ∞ 改成 5，當 D^1 的矩陣有修改時 π^1 也要一起修改，其中 π^1 表示在可以經過點 1 的情況下到達各點所要經過的點，因為是在 D^1 矩陣更新數值，所以 $\pi^1[4, 2]$ 的數值會從 null 改成 1，表示點 4 要到點 2 必須先經過點 1，接著再看

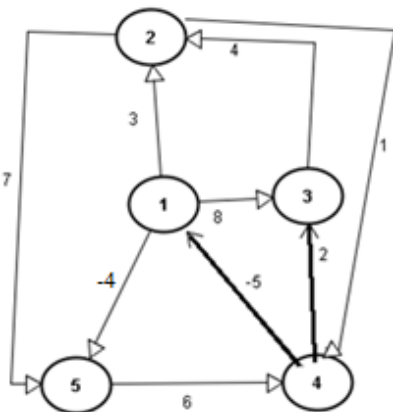


圖 13. 路徑圖

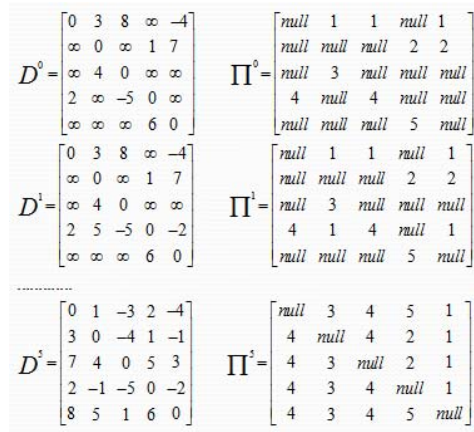


圖 14. 過程圖

$\pi^1[4, 1]$ (其數值為 4) 表示要經過點 4，完整路徑為點 4 到點 1 再到點 2。如果是在 D^0 更新的話，其數值就是 n 。因為有 5 個點所以要算到 π^5 ，表示可以經過點 5 的情況下到達各點所需要經過的點。以點 5 到點 1 為例其順序為：

$\pi^5[5, 1]$ (其數值為 4)，再看 $\pi^5[5, 4]$ (其數值為 5)，因此路徑為 $5 \rightarrow 4 \rightarrow 1$ 。

三、實作

(一) 實作酷哥找正妹

當正妹或帥哥想配對時，可以透過 TRACKER 或擁有 GPS 功能的移動裝置來連接我們的伺服器端，那麼我們就可以取得使用者的位置並隨時更新資訊，圖 15 為取得位置。

圖 15 中，要先設定服務器的位置和 PORT，使用者依據我們給的 IP 來連接上線，其中會顯示使用者目前裝置的電量、座標、移速和使用者身分等資訊。

接著再依據使用者帳號來判斷是男是女，當確定是女生時，我們會將正妹資料輸入伺服器並在右邊的 Google Maps 上更改位置，圖 16 輸入位置。

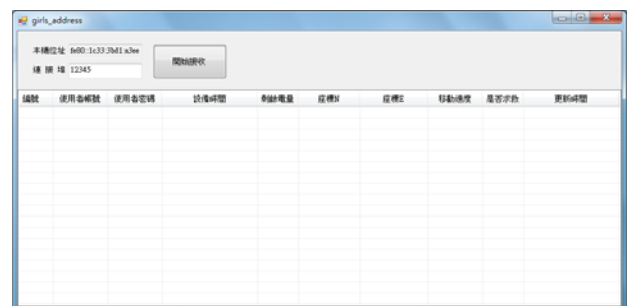


圖 15. 取得位置

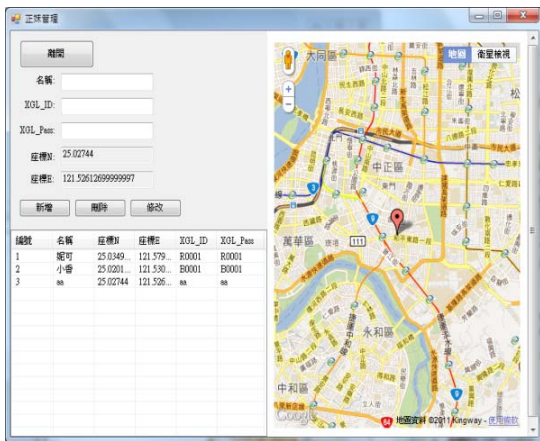


圖 16. 輸入位置

圖 16 中，我們會依據之前取得的訊息來輸入名稱、ID、密碼和座標，右邊會顯示座標在地圖上的位置，左下方會把我們輸入的資料記錄起來。

想要找正妹的酷哥必須是我們登記在案的，這樣我們才能確保其真實性與安全性。我們事先會把酷哥的資料和已經確認身分的正妹作配對，如果在這裡不配對則不會通知和正妹有關的一切資訊，圖 17 顯示配對的情形。

圖 17 中，我們會把要求配對的酷哥會現作模糊化配對，再依據數值來幫他配對，當確定配對對象時，我們會把酷哥的名稱、ID、密碼和手機號碼輸入其中，再為他挑選模糊輸出數值相近的人配對。完成配對時，會把其訊息顯示在下方的儲存表中。

接著我們要先啟動配對程式來等待酷哥上線，當酷哥連上伺服器時會我們依據圖 18 給予提示，如圖 18 之距離提示。

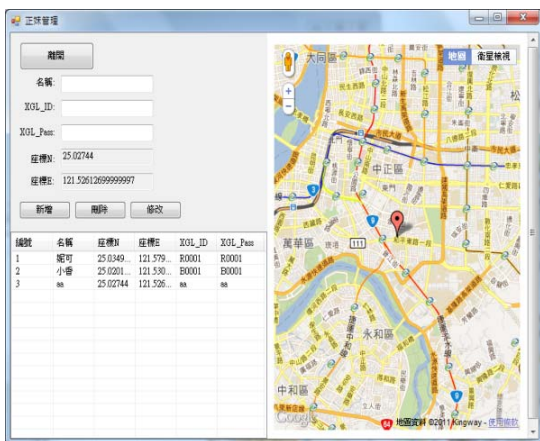


圖 17. 配對



圖 18. 距離

圖 18 中，右方顯示酷哥和被配對之正妹的所在位置，彼此間會以紅線連接，並標示連接者、連接時間、所在方向和兩者間的距離，接著再發簡訊通知酷哥和正妹彼此的資訊。左方上面是我們伺服器端的 IP 和 PORT 酷哥必須依此 IP 來連接，下方會顯示連接者的資料和歷史訊息。

(二) NS2 模擬傳輸路徑

我們使用 NS2 模擬器來來模擬在動態路由下的點一 (node 1) 所能到達的路徑，如圖 19 路徑圖所示。

圖 19 中點 1 和點 3 表示使用者，其餘的點表示我們的服務點，直線表示可以行走的路徑，至於點 1 和點 3 到各點的長度，如表 1 路徑長。

表 1 中，距離為 ∞ 者表無法到達，而負數表捷徑，並假設使用者使用無線傳輸方式來進行傳輸，如圖 20 傳輸方式圖所示。

圖 20 中，隨著下方的時間軸的移動，我們預設的點也會跟著移動好用來模擬傳輸方向、預定到達地點和時間，同時程式也會計算遺失率和延遲，若遺失或延遲過大，那我們會更換路徑以達到降低延遲和遺失率。

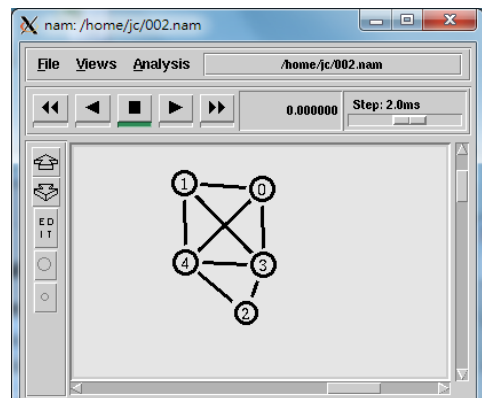


圖 19. 路徑圖

表 1. 路徑長

距離	點 1	點 2	點 3	點 4	點 5
點 1	0	3	8	∞	-4
點 2	∞	0	∞	1	7
點 3	∞	4	0	∞	∞
點 4	2	∞	-5	0	∞
點 5	∞	∞	∞	6	0

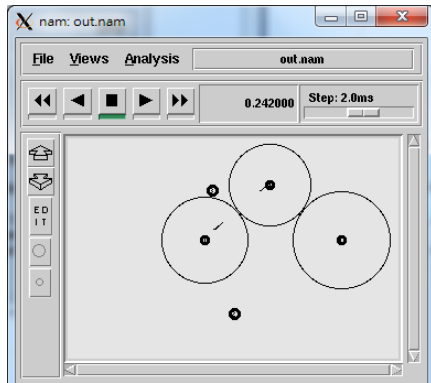


圖 20. 傳輸方式

至於從哪點移動到哪點之判斷方式，我們使用 Floyd-Warshall algorithm 來做為判斷的依據，其演算法最終判斷結果，如表 2 最終判斷。

由表 2 可知如果點 1 要到點 3 要先經過點 4，因為輸出矩陣在 [1, 3] 的地方是 4，接著再看 [1, 4] 會得到 5，所以要經過點 5，接著看 [1, 5] 要經過點 1，故其完整路徑為點 1 到點 5 在到點 4 最後到達目的地點 3。這是經過的 Floyd-Warshall algorithm 所判斷到達各點之最短路徑之方式。因為我們使用 NS2 來模擬使用者的行動方向，故如果到達我們所預定的時間但使用者卻未到達指定點，我們就假設使用者迷路了，所以會再依據使用者現在的位置作為出發點再算一次演算法，並發出一封簡訊給予提示。

表 2. 最終判斷

距離	點 1	點 2	點 3	點 4	點 5
點 1	∞	3	4	5	1
點 2	4	∞	4	2	1
點 3	4	3	∞	2	1
點 4	4	3	4	∞	1
點 5	4	3	4	5	∞

(二) 模糊化配對

此實驗透過管理人員在外表方面所累積的經驗來分析配對情況，並建立其規則。將身高和 BMI 分級並當作模糊規則的輸入；完成推論後，經過解模糊化，得到一個明確的輸出信號，我們依此數值為判斷配對依據，來評估配對方式。其步驟如下：

1. 資料模糊化

對資料做模糊化。首先定義身高，利用詢問所得到的結果，大致上將身高分成三個等級，矮、中等和高，如圖 21 身高分佈圖所示。

其縱軸表示身高模糊集的隸屬度、橫軸表示身高數值，身高分級={矮、中等、高}=由我們所詢問的路人所得。

矮：150~170 公分

中等：165~185

高：180 公分以上

若身高為 176 公分，則分配到中等這個等級，且其隸屬度是 1。

接著定義 BMI，在此之所以不用體重，主要是因為身高跟體重有相對上的關係，其資料來源於衛生署發布的 BMI 資料，在加上一些模糊地帶，可分成三個等級，輕、標準和重，如圖 22 BMI 分佈圖所示。

其縱軸表示 BMI 模糊集的隸屬度、橫軸表示 BMI 數值，BMI 分級=由我們所詢問的路人所得={輕、標準、重}。

瘦：20 以下

標準：16~26

重：24 以上

若 BMI 數值為 35，那麼會被分配到重這個等級，且模糊集的隸屬度為 0.91。

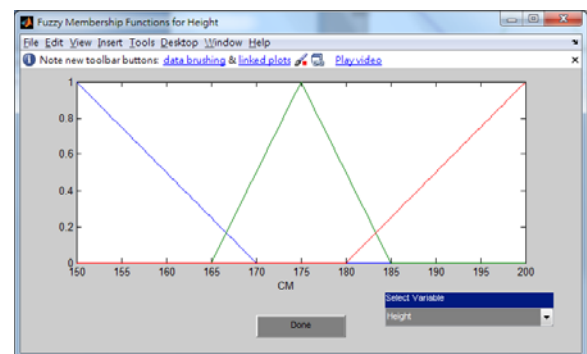


圖 21. 身高分佈圖

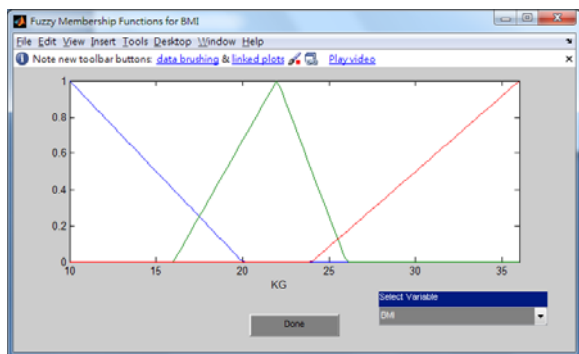


圖 22. BMI 分佈圖

2. 定義規則

首先會依據使用者的身高分佈來確認第一分佈的條件，接著再依據使用者的 BMI 來決定第二個條件，若使用者的身高分佈在矮則其等級分佈在非常小到中等_小之間，再依據 BMI 分布來確認最後坐落在哪個等級上，其他分佈亦是如此判斷，其定義規則如圖 23 規則圖所示。

建立模糊規則表我們將上面的模糊規則寫成模糊規則表，如表 3 之規則表。

若身高在 170 公分，依據上述的身高分佈圖得出身高指標位於中等，再者此人的 BMI 數值為 15，依據 BMI 分佈圖得出 BMI 指標位於瘦，再依據規則表，此人被會分配到小這個輸出等級。

3. 定義模糊函數與解模糊化方法

在獲取專家知識時常使用 Mamdani 方法。因為這種方法可以用更直接、更接近人類的方式來描述專家的意見，並使用上述的質心法來解模糊化。

```

Rule # 1:
IF Height = short AND BMI = thin
THEN y = very small
Rule # 2:
IF Height = short AND BMI = fat
THEN y = small
Rule # 3:
IF Height = short AND BMI = medium
THEN y = medium_small
Rule # 4:
IF Height = medium AND BMI = thin
THEN y = small
Rule # 5:
IF Height = medium AND BMI = fat
THEN y = medium_small
Rule # 6:
IF Height = medium AND Height = medium
THEN y = large
Rule # 7:
IF Height = high AND BMI = thin
THEN y = small
Rule # 8:
IF Height = high AND BMI = fat
THEN y = medium_large
Rule # 9:
IF Height = high AND BMI = medium
THEN y = very large
    
```

圖 23. 規則圖

表 3. 規則表

配對輸出等級		身高指標		
		矮	中等	高
BMI 指標	瘦	極小	小	小
	標準	中等小	大	非常大
	胖	小	中等小	中等大

4. 逆模糊化

為每個使用者算出每個人的數值，再讓數值相近的人先行配對，如果不滿意，可以依照使用者所提出的條件再做一次模糊化配對（例如把配對對象的部份參數調高，像把身高調高一點...等），直到使用者滿意為止。只要更動部份參數即可提高配對數值，第一個參數是身高、第二個是 BMI，如表 4 結果表。

表 4 中身高 190 和 195 在身高分佈圖中為高這個等級，BMI 數值 35 在 BMI 分佈圖再胖這個等級，而數值 22 在標準這個等級。故身高 190 和 BMI 數值 35 分配到中等_大這個等級，再用之前說過的步驟來取得清晰的輸出數值，最後得到 0.6344，儘管身高 195 和 BMI 數值 35 被分配到相同的等級裡（中等_大），但身高的參數數值較高，所以最後得到的輸出數值會較大些（0.6360），身高 190 和 BMI 數值 22 被分配到非常大這個等級，所以得出的數值會很大（0.9430）。

定義模糊輸出的等級，等級越高代表越好，其中分成六個等級，其分佈如圖 24 等級分佈圖所示。

縱軸表示輸出等級的模糊集的歸屬度，橫軸表示模糊集的輸出配對等級數值分別如下所示：

- 極小：0.2 以下
- 小：0.15~0.4
- 中等_小：0.36~0.6
- 中等_大：0.46~0.8
- 大：0.71~0.9
- 非常大：0.86 以上

若依據上述的規則表得出配對輸出等級位於非常大這個等級，其配對等級數值會分佈於 0.86 到 1 之間，再依據

表 4. 結果表

身高	BMI	Fuzzy 數值
190	35	0.6344
195	35	0.6360
190	22	0.9430

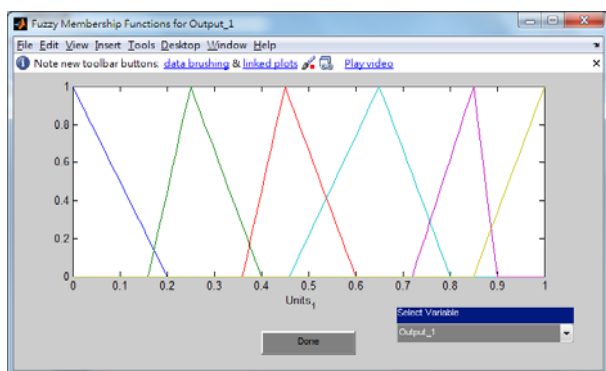


圖 24. BMI 分佈圖

其身高模糊集的隸屬度數值和 BMI 模糊級的隸屬度數值作 AND 或 OR 運算得出新的隸屬度數值，接著使用這新的隸屬度數值做上述解模糊化運算，即可得出最終輸出等級數值，如：

身高：190 公分

身高隸屬度：0.6...

身高分佈等級：高

BMI：22

BMI 隸屬度：0.8...

BMI 分佈等級：標準

輸出分佈等級：非常大

AND 運算後的隸屬度：0.6...

OR 運算後的隸屬度：0.8....

質心法解模糊化得出數值：0.9430。

(三) Fuzzy 配對與傳統比較

為了驗證上章所提出的 Fuzzy 配對的有效性，我們與傳統配對作一個比較，好驗證其功用，表 5 為隨機數值表。

首先我們隨機產生十組數字，其中 1 到 5 組為男生，另外 6 到 10 組為女生，有身高和 BMI 兩個數值，身高分佈於 100 公分到 210 公分之間，而 BMI 分佈於 10 到 36 之間，其中 Fuzzy 數值是把身高和 BMI 的數值輸入到上述的模糊化配對所得出的數值，以下先作傳統配對，表 6 為傳統配對表。

傳統配對，也就是隨意配對，在此直接用男生配女生，即 1 配 5、2 配 6、3 配 7 依此類推，配對完後再依據身高差和 BMI 差來決定成功率，而成功率從 0 分到 100 分，身高每差 1 公分扣 1 分扣滿 80 分為止，BMI 數值每差 3 就扣 10 分。如：1 號跟 6 號身高差 89 公分，所以先扣 80 分，BMI

表 5. 隨機數值表

編號	身高	BMI	模糊數值
1	109	22	0.56
2	154	27	0.27
3	133	19	0.49
4	111	32	0.37
5	139	32	0.37
6	198	17	0.41
7	167	11	0.31
8	109	14	0.51
9	180	15	0.27
10	136	17	0.5

表 6. 傳統配對表

編號	配對對象	身高差	BMI 差	成功率
1	6	89	5	4
2	7	13	16	57
3	8	24	5	60
4	9	69	17	1
5	10	3	15	67

數值差 5，故扣 16 分，最後得出 4 分。這就是 1 號跟 6 號的成功率。

接著依據 10 人的 Fuzzy 數值的接近度來配對，其配對結果如表 7 之模糊配對表。

我們依據各別 Fuzzy 數值相差最少者且為異性的條件下來給予配對，因為 1 號是男生且 Fuzzy 數值為 0.56，8 號是女生且 Fuzzy 數值 0.51 與 1 號相差最少，所以把他們配成一對，其他人亦是如此。

最後，比較使用傳統配對和 Fuzzy 配對兩者的成功率，把各別 5 筆成功率相加再取平均，傳統配對得出 37.8 的平均成功率、Fuzzy 配對得出 46.8 的平均成功率，結果顯示 Fuzzy 配對在相對上是有效提高配對的成功率如圖 25。

表 7. 模糊配對表

編號	配對對象	身高差	BMI 差	成功率
1	8	0	8	74
2	9	26	12	44
3	10	3	2	91
4	7	56	21	14
5	6	59	15	11

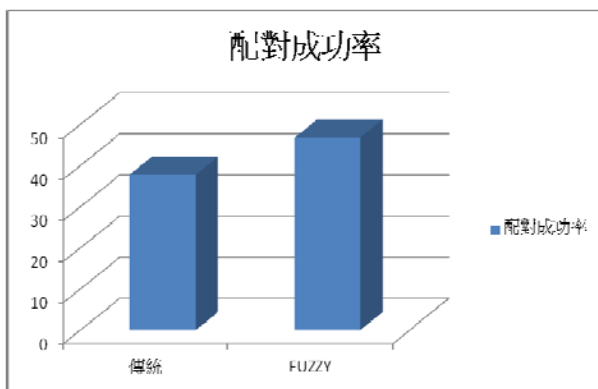


圖 25. 比較圖

其縱軸數值為平均成功率，橫軸為配對方式名稱。

傳統配對成功率： $(4+57+67+60+1)/5=37.8$

Fuzzy 配對成功率： $(74+44+91+14+11)/5=46.8$ 。

四、結論

本論文主要是提供一個能在現實中為人們配對的方式，主要是透過全天候無阻的 GPS 來提供精準的定位，使用模糊理論來為人們做定義不清楚上的配對，並用 NS2 來模擬道路加入此人的行走速度來確認此人是否有在時限內到達指定點。再者我們架構了一個循序圖來表示流程該如進行，什麼時候該做什麼事。最後，我們希望能用在路上尋找到配對對象的方式來提升男女配對的樂趣程度，進而提升遊戲的品質。

其中，我們使用 Floyd-Warshall 演算來取得該點到達各點的路徑，希望能縮短花在路上的時間，加快配對的兩人見面。並使用模糊理論來分析配對者的外在參數，主要是希望能提升配對的成功率，若配對失敗則我們會在配對者的要求給予修改參數數值，直到配對者滿意為止。我們使用 NS2 來模擬配對者的行走方向與速度，若我們伺服器顯示配對者朝不對的方向前進，那我們馬上發簡訊通知配對者已修正失誤。以上這些機制主要是想讓配對者能夠輕鬆玩、輕

鬆找和輕鬆配對。

未來的研究，感謝學者專家的具體建議，茲整理如下：

- 一、若 GPS 在室內或其他障礙無法接受訊號，系統可結合智慧型手機進行處理。
- 二、本論文輸入部分只有身高和 BMI，未來可以再增加一些特徵（例如臉部影像等）以增加配對成功率。
- 三、本論文因需要輸入一些個人私密資料，未來系統建置時需再考慮資安防護和個資保密問題。

參考文獻

1. 柯志亨 (民 95)，計算機網路實驗－以 NS2 模擬工具實作，學貫出版社，台北市。
2. 柯志亨 (民 96)，計算機網路實驗－使用 NS2 模擬多媒體通訊與無線網路，學貫出版社，台北市。
3. 張振宏 (民 96)，利用統計式模糊流量控制防止分散式阻斷服務攻擊，國立成功大學電腦與通信工程研究所碩士論文。
4. 游謹憶 (民 98)，人工智慧基礎理論實務與實作，智控科技股份有限公司，台北。
5. 劉仁筑、鄔智仁 (民 97)，GPS 程式設計開發實戰，學貫出版社，台北市。
6. 歐俊宏 (民 98)，儲料位置選擇之模糊層級分析法應用－中鋼公司為例，國立成功大學資訊管理研究所碩士論文。
7. 謝復雅 (民 95)，基於 IEEE 802.16e 的跨層換手協定研究，國立中央大學通訊工程研究所碩士論文。
8. 鍾添曜 (民 100)，元智資工網路實驗室－NS2，民 100 年 11 月 4 日，取自 http://netlab.cse.yzu.edu.tw/ns2/netlab/ns2_web site/。

收件：100.11.07 修正：100.12.05 接受：100.12.22