

模糊狀態機於即時戰略遊戲人工智慧的理論應用與實作

Real Implementation by Fuzzy State Machine for Real-Time Strategy Game

陳光雄 施昆志*

南台科技大學多媒體與電腦娛樂科學所

M93V0203@webmail.stut.edu.tw

摘要

現今遊戲產業發展迅速，各種技術不斷推陳出新，使得遊戲所能呈現的內容越來越豐富及多元化，而遊戲的產值也逐年增加，因此，如何提升遊戲本身的可玩性也就相對的非常重要。傳統即時戰略遊戲多採用作弊 (Cheating) 或有限狀態機 (Finite state machine, FSM) 來設計遊戲人工智慧，使用作弊的方式設計經常會使得玩家覺得努力白費，而失去對遊戲的興趣；而採用有限狀態機的設計方式容易使得玩家找出其中的遊戲規則，而失去對遊戲的耐玩度。針對此一缺點，本研究利用模糊狀態機具有模糊決策的能力，使得電腦在決策時能因應更多的變化，增加遊戲的可玩性。我們同時將此方法應用在實際的遊戲上，結果遊戲的過程較有彈性且富變化，也驗證了所提方法的可行性。

關鍵詞：遊戲人工智慧、模糊狀態機、模糊邏輯、即時戰略遊戲

ABSTRACT

Game techniques and corresponding products are widely developed in recent years. How to attract and interest people is their common development goal. Traditional ways of game design in artificial intelligence (AI) are generally cheating and/or finite state machine (FSM). In the cheating way of game design, the game player cannot perceive the subsequent computer behaviors or reactions, which often make the play's efforts futile and useless. Consequently, the game player loses confidence and interesting gradually. In the FSM way of design, the player can easily predict the rules of computer behaviors and reactions, which also makes the game process insipid and unattractive. This motivates us to use fuzzy state machine as a real-time AI strategy to design and develop a game. It is predictable that games will become more attractive and interesting by using the proposed fuzzy AI Strategy. Real game implementation confirms the effectiveness of the proposed approach.

Keywords : Game artificial intelligence 、Fuzzy state machine 、Fuzzy logic 、Real-time strategy

1.前言

人工智慧 (Artificial intelligence, AI) 在電腦遊戲中的應用開始於 1980 年代大型遊樂場遊戲的流行，但是此時的人工智慧被人們重視的程度仍不及對遊戲多媒體效果的追求。在 1989 年，IBM 開發的超級電腦深藍 (Deep Blue) 可以模擬人類思考模式，並於 1996 年開始與世界棋王卡斯帕夫洛 (Garry Kasparov) 進行對戰，但是結果卻是電腦落敗。隔

年 (即 1997 年)，超級電腦深藍二度挑戰棋王，結果以二勝一負三和的成績打敗了世界棋王 [7]，人工智慧此時才真正的被一般人們所注意 [8][9][11]。此後遊戲人工智慧也被更深入地研究且應用於不同的遊戲類型中，如即時戰略遊戲 (Real-time strategy, RTS)。

在現行即時戰略遊戲中，最常被使用的 AI 技術就是作弊 (Cheating)，其執行方式是即時戰略遊

戲一開始便讓電腦擁有比玩家還多的可用資源，電腦不需要執行收集資源的動作便可以隨意擁有資源。其次是利用有限狀態機 (Finite state machine, FSM) 來控制遊戲腳色的行為，利用事先定義好的條件來轉移狀態，使得遊戲腳色能夠依據不同的情況反應動作。這類的遊戲人工智慧設計容易被玩家找出規則而降低遊戲的耐玩性。在本篇論文中，將討論如何利用模糊狀態機 (Fuzzy state machine, FuSM) 應用於即時戰略遊戲之中，以提高即時戰略遊戲的耐玩度。在論文架構的安排上，首先定義並描述何為電腦即時戰略遊戲，並討論即時戰略遊戲上的人工智慧設計方式與其它類型遊戲的不同之處，接著介紹目前最常被採用的人工智慧設計方式，即有限狀態機 [3]，並討論此一設計方式可能發生的問題，最後提出如何將模糊狀態機理論實際應用於即時戰略遊戲之中，遊戲的實驗結果證實了所提方法的有效性與可行性。

2. 文獻探討

2.1 人工智慧

長久以來，人們一直希望製造出一個實體能像人一樣具有思考及高度的智慧，能夠幫助我們處理複雜事務，但是又不會疲倦、抱怨，人工智慧就是其中最主流的代名詞 [4]。它能模擬人類思考的過程與步驟，並利用電腦程式當邏輯工具來解決問題，也就是說，人工智慧提供一套簡單而具結構化的方法來設計出可以協助人們做決策的程式。目前人工智慧的發展已經應用到許多重要的領域如電腦視覺 (Computer vision)、語音辨識 (Speech understanding)、專家系統 (Expert system)、自然語言處理 (Natural language understanding)、智慧型代理人 (Intelligent agent)、類神經網路 (Artificial neural network)、機器人應用 (Robotic application)，未來也勢必成爲一個人工智慧的年代。

2.2 發展即時戰略遊戲人工智慧需具備的特性

即時戰略遊戲需同時具備一般戰爭遊戲的特性與動作遊戲的元素，而在戰爭類型遊戲中，人工智慧的需求相對重要，因爲它關係到遊戲本身的可變性與耐玩性。由於在即時戰略類型遊戲中，電腦所必須要控制的非玩家角色 (Non-player

character, NPC) 數量不只一個，而且 NPC 的行為反應必須是即時性的，所以在設計即時戰略類型的遊戲人工智慧時，必須注意以下幾點：

- (1) 演算法時間複雜度不能太高，因爲必須即時反應。
- (2) 演算法須具有彈性，因爲控制的 NPC 不只一個，且類型不一定相同，例如攻擊單位與生產製造單位。
- (3) 即使相同類型的單位其動作也不應該完全一致。

2.3 有限狀態機

有限狀態機是目前最受歡迎的遊戲人工智慧技術 [3][5]，有限狀態機是一種抽象機制，是由一組狀態、一個起始狀態、輸入、狀態轉換函數所組成，實際的狀態會決定狀態機的行為 [6]。有限狀態機很早就被應用在遊戲的設計之中，例如小精靈遊戲 (圖 1) 中鬼怪的行為模式，鬼怪的行為有隨意走動、追逐玩家或閃躲玩家，它們會依據不同的狀態，來表現出不同的行為，在還沒有遇到玩家時，鬼怪會隨意走動，當鬼怪遇到玩家，馬上就會改變狀態，從隨意走動變成追逐玩家，當玩家吃到大力丸之後，鬼怪的狀態又會從追逐玩家轉變成閃躲玩家。

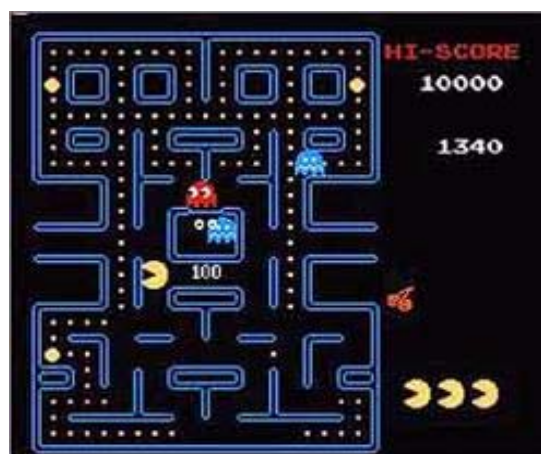


圖 1. 小精靈遊戲

圖 2 爲利用圖形來描述鬼怪在遊戲中可能出現的有限狀態機，每一個圓圈表示鬼怪目前可能所處的狀態，有三種可能的狀態：隨意走動、追逐玩家

及閃躲玩家，而箭頭表示狀態轉變的方向跟條件。值得注意的是狀態轉移的條件是明確的，簡單來說條件推論的結果不是真 (True) 就是假 (False)。

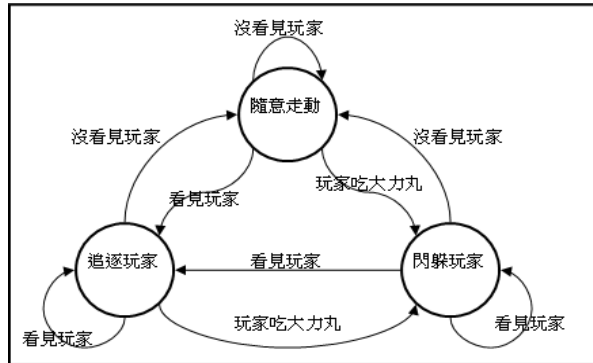


圖 2. 鬼怪有限狀態機

有限狀態機適合使用在類似即時戰略這類型電腦需要控制數個 NPC 的遊戲之中，因為系統架構簡單易懂，在實作上容易寫成程式，並且可以使得 NPC 表現出多樣化的行為。下面歸納出其主要的優點：

- (1) 有限狀態機可以將複雜的行為模式，簡化為易懂的系統架構。
- (2) 易於擴充，可以在最小幅度的改變下增加新的行為模式 [10]。
- (3) 與其他的人工智慧技術如類神經網路或遺傳演算法比較起來，有限狀態機更易於系統的除錯 [4]。
- (4) 演算法容易轉換成爲實際的程式碼。

但是，也由於狀態轉換的條件明確，容易使得玩家在玩過數次之後發現其中的規則，因而降低遊戲的多變性與耐玩度，而且並非所有的狀態轉換條件都可以輕易的被轉換成布林條件式。在即時戰略遊戲中，如果我們想要讓電腦的思考方式更接近人腦的思維，例如，當玩家所控制的軍隊與電腦所控制的軍隊接觸時，電腦可以根據威脅的程度來決定採用的戰術，這樣的需求就不易利用有限狀態機實作出來。

2.4 模糊邏輯

在現實生活中，人類也往往需要在沒有足夠的資訊情況下，做出明確的判斷，著名的「禿頭理論」

便是其中一例。該理論描述究竟有多少根頭髮才算是禿頭呢？假如我們定義頭上沒有任何一根頭髮便稱之為禿頭，那如果頭上只有一根頭髮，算不算是禿頭呢？在一般常識的定義下，如果定義只比禿頭多一根頭髮也稱之為禿頭，依此不斷的推論下去，便會產生全世界都是禿頭的矛盾結果。模糊理論便是用來解決這類的問題。

模糊理論是由美國加州大學 L. A. Zadeh 教授於 1965 年提出來的。他提出了模糊集合 (Fuzzy set) 的觀念，強調以模糊邏輯來描述現實生活中事務性質的等級 [2]。電腦強於人類的地方在於電腦擁有優越的計算能力，對於無法以計算法則解決的問題，如概念、思考、推理、識別等問題，電腦解決的能力就不如人類了。因此，模糊理論就是針對人腦對於模糊的訊息或不完整的資料，因其不需經過精密繁雜的計算過程，仍能做出有效判斷的目的而發展出來。同樣是採用傳統布林條件作為設計就可以達成的工作，Timothy Masters 在他所著作的書中也有提到，採用模糊規則可以比傳統使用布林邏輯規則要少 50% ~ 80% 的規則，便可以完成相同的任務 [12]，如此可以減少電腦邏輯運算的數量。

3. 模糊狀態機

本研究提出在即時戰略遊戲中，當電腦所控制的軍隊與玩家所控制的軍隊接觸時，要設計出一套戰術讓電腦可以根據評估敵我優劣勢的差異而採取不同的作戰方式，爲了達到這個目的，本研究採用改良式的有限狀態機，即模糊狀態機 [2]，來設計即時戰略遊戲人工智慧，如圖 3：

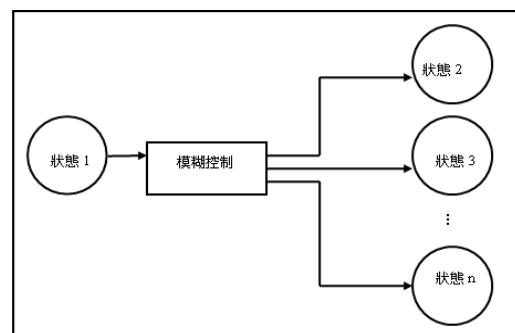


圖 3. 模糊狀態機操作流程

模糊控制主要是由三個步驟所組成，圖 4 說明這些組成的步驟。第一個步驟是模糊化

(Fuzzification)，最主要的目的是把實際明確的資料 (Crisp value) 轉換成模糊集中的歸屬程度 (Degree of membership)。第二個步驟是模糊推論 (Fuzzy inference)，藉由簡單的模糊規則建立決策。在這個步驟中，模糊推論的輸出也是模糊集中的歸屬程度。第三個步驟為解模糊化 (Defuzzification)，就是把推論完成的模糊輸出轉換成可用的資訊。

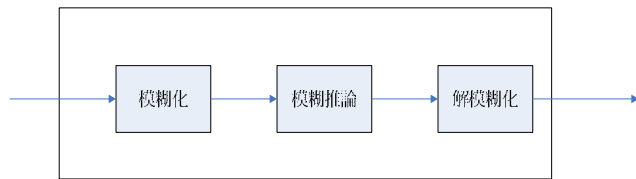


圖 4. 模糊控制步驟

3.1 模糊化

在這個步驟中，利用歸屬函數 (Membership function) 來求得實際資料對應到模糊集中的歸屬程度，下面是兩種常用的歸屬函數形式 [1]。

(1) 三角歸屬函數 (Triangular membership function)，如圖 5 所示：

$$f(x) = \begin{cases} 0; & x \leq x_0 \\ \frac{x}{x_1 - x_0} - \frac{x_0}{x_1 - x_0}; & x_0 < x < x_1 \\ 1; & x = x_1 \\ \frac{-x}{x_2 - x_1} + \frac{x_2}{x_2 - x_1}; & x_1 < x < x_2 \\ 0; & x \geq x_2 \end{cases} \quad (1)$$

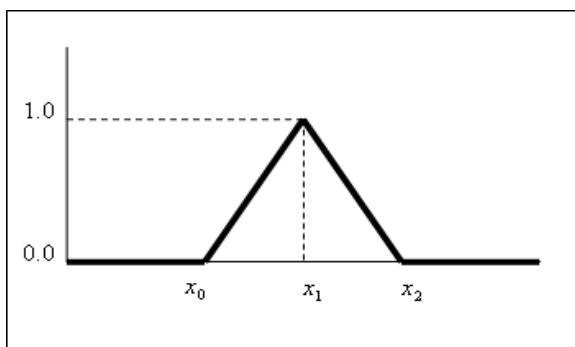


圖 5. 三角歸屬函數

(2) 梯形歸屬函數 (Trapezoid membership function)，如圖 6 所示：

$$f(x) = \begin{cases} 0; & x \leq x_0 \\ \frac{x}{x_1 - x_0} - \frac{x_0}{x_1 - x_0}; & x_0 < x < x_1 \\ 1; & x_1 \leq x \leq x_2 \\ \frac{-x}{x_3 - x_2} + \frac{x_3}{x_3 - x_2}; & x_2 < x < x_3 \\ 0; & x \geq x_3 \end{cases} \quad (2)$$

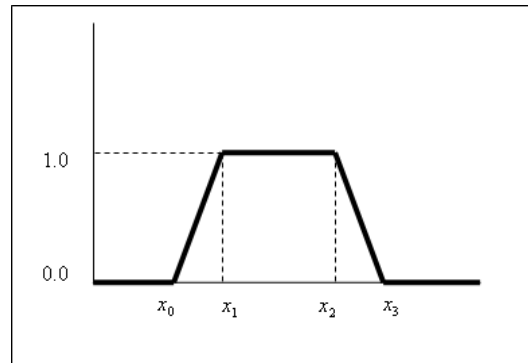


圖 6. 梯形歸屬函數

利用上述兩種歸屬函數的基本型態，便可以根據實際情況的需要，來組合設計出合適的歸屬函數。舉例來說，在即時戰略遊戲中，利用兩個變數，即“玩家與電腦直線距離”與“玩家與電腦軍隊數量差距”來評估電腦與玩家的優劣勢，並藉此決定電腦下一步該採取什麼行動。圖 7 是將電腦與玩家距離資料模糊化的歸屬函數，將實際數量差距的值轉換為模糊集合的值，即可得到該值在逼近、附近、遠、極遠四種模糊集合的歸屬程度。

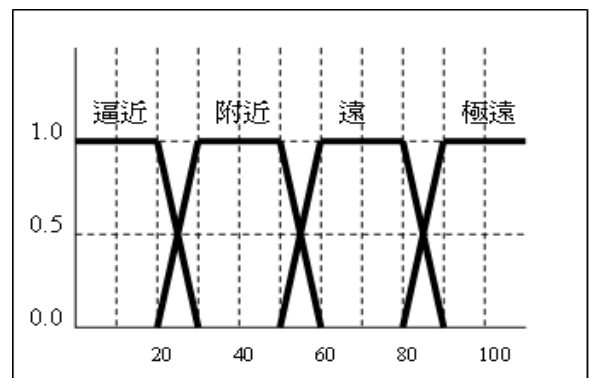


圖 7. 電腦與玩家距離資料歸屬函數

例如當電腦與玩家距離為 22 單位時，即可求得對應的歸屬程度結果值如表 1 所示：

表 1 距離歸屬程度值

| 模糊集合 | 歸屬程度值 |
|------|-------|
| 逼近 | 0.8 |
| 附近 | 0.2 |
| 遠 | 0.0 |
| 極遠 | 0.0 |

設計玩家與電腦軍隊數量差距的歸屬函數可設計為如圖 8 所示：

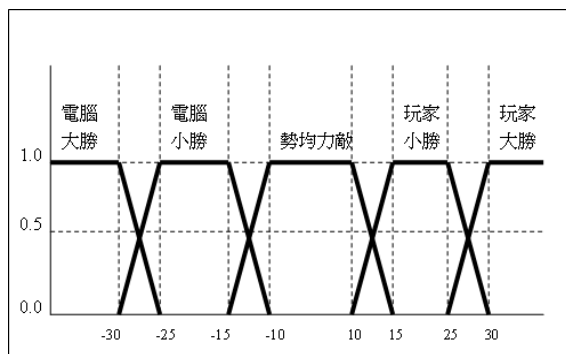


圖 8. 電腦軍隊數量歸屬函數

假定玩家軍隊數量為 30 單位，電腦軍隊數量為 57 單位時，則玩家與電腦軍隊數量差距為 -27，而所計算出的相對歸屬程度結果值如表 2 所示：

表 2 軍隊數量差距歸屬程度值

| 模糊集合 | 歸屬程度值 |
|------|-------|
| 電腦大勝 | 0.4 |
| 電腦小勝 | 0.6 |
| 勢均力敵 | 0.0 |
| 玩家小勝 | 0.0 |
| 玩家大勝 | 0.0 |

3.2 模糊推論

將實際資料模糊化後，接著定義一組邏輯推演的規則，可以藉由模糊輸入推導出模糊結論，該規則定義如式 (3) ~ (7)：

$$\forall n, A, B \in \text{Fuzzy set} \quad (3)$$

$$T(n) = \text{Degree of membership} \quad (4)$$

$$T(A \text{ and } B) = \text{Min}(T(A), T(B)) \quad (5)$$

$$T(A \text{ or } B) = \text{Max}(T(A), T(B)) \quad (6)$$

$$T(\text{not } A) = 1 - T(A) \quad (7)$$

上述規則在作交集運算 (AND) 時，採用的是保守評估，而在作聯集運算 (OR) 時，採用的是積極評估。因此，在玩家軍隊數量為 30 單位，電腦軍隊數量為 57 單位，且兩軍距離為 22 單位時，可以推論出以下結果：

該進攻歸屬程度

$$= (\text{逼近歸屬程度 and 電腦小勝歸屬程度})$$

$$= \text{Min}(0.8, 0.6)$$

$$= 0.6 \quad (8)$$

利用這樣的方法，可以制定出許多模糊規則，而這樣的思考方式與策略，假如玩家軍隊逼近且電腦軍隊數量小勝於玩家軍隊數量時，則電腦應該選擇進攻，比較為接近人類制定決策時的方式。由於模糊規則的數量多寡與模糊集合數有關，所以在制定模糊規則時，應避免重複定義。舉例來說，假如在制定模糊規則時，認為只要電腦軍隊數量大勝且距離逼近、電腦軍隊數量大勝且距離在附近、電腦軍隊數量大勝且距離都應該選擇攻擊，則在制定規則時，可以定義成電腦軍隊數量大勝且距離沒有極遠則電腦選擇攻擊，如此可以將三條模糊規則簡化成為一條，可以減少電腦邏輯運算的次數，避免過多的模糊規則增加電腦運算的負擔。

3.3 解模糊化

解模糊化主要是將模糊推論後的模糊結果，轉換成有用的資訊，以利後續的處理。在模糊理論中，常用的方法有下列幾種 [1]：

- (1) 重心法 (Center of gravity)
- (2) 面積法 (Area)
- (3) 高度法 (Height)
- (4) 最大值平均法 (Midpoint of maximum)

本研究利用模糊推論後的模糊集合歸屬程度，轉換為百分比，藉由這個輸出資訊作為一個狀態轉換另外一個狀態的或然率 (probability)，採

用這樣的方法，電腦不需要負擔太多的計算成本，符合即時戰略遊戲對於即時性的需求。

因此，在玩家軍隊數量為 30 單位，電腦軍隊數量為 57 單位，且兩軍距離為 22 單位時，可以推理得到電腦有 60% 的程度應該進攻玩家。藉由敵我實力的差距，來決定該使用何種回應策略的百分比程度，這樣的決策方式比較接近實際戰爭對於行動策略的決策方式。

3.4 實驗

本論文利用 Java 程式開發一個模糊狀態機的實驗程式，在這個程式中先輸入軍隊的數量，電腦再依據輸入的數量來決定是攻擊或者是不攻擊，在程式中以圖 9 代表玩家角色，以圖 10 表示電腦不攻擊的圖案，而圖 11 表示電腦攻擊的圖案。

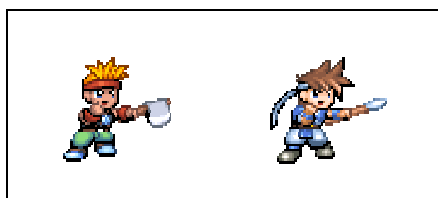


圖 9. 玩家



圖 10. 電腦不攻擊狀態



圖 11. 電腦攻擊狀態

利用有限狀態機設計，在符合電腦軍隊數量比玩家軍隊數量多的情況之下，電腦所有角色都會轉變成攻擊狀態，如圖 12 所示。



圖 12. 使用有限狀態機結果

而利用本研究所提出的方法來設計遊戲，同樣的數量情況之下，電腦並不會全部轉變為攻擊狀態，如圖 13 所示，而是利用模糊規則來決定是否轉為攻擊，因此較具彈性與變化，當然也能增加遊戲的多變性與耐玩性。



圖 13. 使用模糊狀態機結果

而且在每一次相同情況下，電腦的反應不見得會完全一樣，這樣可以讓玩家不易預測出電腦所將採用的策略為何，而讓整個遊戲過程猶如實際戰爭一樣，不易捉摸。

4. 結論

本研究提出藉由模糊狀態機設計出即時戰略遊戲所需的人工智慧，相較於使用有限狀態機，此一方式更加接近人類的思維模式，使得電腦的行為準則更加多樣而富變化，遊戲的過程更加接近實際作戰，藉此增加遊戲的趣味性與耐玩度，我們同時也將此理論應用於實際的戰爭遊戲中，遊戲的過程驗證了所提方法的有效性與可行性。利用模糊狀態機來設計即時戰略遊戲的優點歸納如下：

- (1) FuSM 降低 NPC 行為的可預測性，可以提升遊戲的可重玩性 (Replayability)。
 - (2) 在傳統的方式下，NPC 選擇攻擊或者是不攻擊行為較為固定，採用 FuSM 會讓玩家面對一個擁有不同反應程度的 NPC。
 - (3) 在相同判斷變數條件下，可以減少所需定義的規則。
 - (4) 藉由調整歸屬函數便可以輕易改變 NPC 面對相同情況的反應，不需要大幅度修改規則。
 - (5) 較為容易將企劃的腳本直接轉換為 AI 規則，降低企劃轉換為實際程式的難度。
6. David M. Bourg, Glenn Seemann, (2002), *AI for game developers*, O'Reilly
 7. IBM Deep Blue，來源：
<http://www.research.ibm.com/deepblue/>
 8. Khoo, A, Zubek, R., (2002), Applying inexpensive ai techniques to computer games, *IEEE Intelligent Systems*
 9. Kenneth D. Forbus, James V. Mahoney, Kevin Dill, (2002), How qualitative spatial reasoning can improve strategy game AIs, *IEEE Intelligent Systems*

參考文獻

1. 王進德，蕭大全，(1994)，*類神經網路與模糊控制理論入門*，台北：全華圖書公司
2. 陳耀茂，(2003)，*Fuzzy 模糊理論*，台北：五南出版社
3. 吳忠霖，(2004)，*即時戰略連線遊戲基礎架構之研究*，國立台灣科技大學機械工程系碩士論文
4. 呂威甫、鄭軒鴻，(2004)，*人工智慧 - 現代方法總論*，台北：松崗科技股份有限公司
5. Daniel Sanchez-Crespo Dalmau, Daniel Sanchez-Crespo, (2002), *Core techniques and algorithms in game programming*, New Riders
10. Ottawa, Canada, São Carlos, SP, Brasil, (2004), 3D web-based virtual simulation environments extensibility through interactive non-linear stories, *proceedings of the eighth IEEE international symposium on distributed simulation and real-time applications*
11. Siang Y. Chong, Member, IEEE, Mei K. Tan, Jonathon D. White, (2005), Observing the evolution of neural networks learning to play the game of othello, *IEEE transactions on evolutionary computation*
12. Timothy Masters, (1993), *Practical neural network recipes in c++*, Morgan Kauffman