

第二章 模糊控制理論

2.1 模糊控制之理論背景

模糊理論 (Fuzzy Theory) 的起源始於 1965 年美國加州柏克萊大學的 L. A. Zadeh 教授提出以模糊集合 (Fuzzy Sets) 來將模糊資訊量化的理論[1]。模糊理論較之於傳統數學理論，在結構上很類似，但本質上有所不同。例如：在邏輯上，模糊邏輯是採用多元邏輯，敘述除了可以是真或假外，其中還允許許多漸進的值；而模糊集合論中，集合的歸屬度，不再只有是與否，而有程度上多或少的差別。當這些模糊理論發展的更成熟時，其觀念往往擴大到可以包含傳統數學理論中相對應的理論，讓傳統理論變成只是模糊理論的特例[37]。目前模糊理論已廣泛地應用於各個領域，如：自動控制、廢水處理、教育成果評量、決策支援、圖形識別、醫療診斷、經濟預測 等。

英國倫敦大學 Queen Mary 分校的 E. H. Mamdani 教授首先根據 L. A. Zadeh 所提出的語意式分析法 (Linguistic Approach) 與模糊推論 (Fuzzy Inference)，成功的應用於蒸汽機自動運轉控制上[2-3]，從此模糊控制便逐漸在控制領域嶄露頭角，而成為控制領域的一支。1980 年代模糊控制被日本廣泛應用於家電產品及地下鐵自動駕駛系統上，更是因而聲名大噪。顯然可見，模糊控制是模糊理論應用最成功、成果也最豐碩的一支。根據模糊控制理論所設計出來的模糊邏輯控制器 (Fuzzy Logic Controller, FLC)，是一個容易控制與掌握的非線性控制器，並具有絕佳的適應性 (Adaptation)、強健性 (Robustness) 以及容錯能力 (Fault Tolerance) 等優點，因此特別適用於非線性、時變或模式不確定的系統上。目前與模糊控制有關的研究，在理論方面有，Qiao 和 Mizumoto 利用非等距式三角型歸屬函數以及局部線性化的技巧，將 FLC 近似為傳統的 PD 控制器[9]；Lewis 和 Liu 利用非等距式三角型歸屬函數解析輸入與輸出之關係，只要輸入或輸出歸屬函數呈現非等距式

分佈，FLC 才能提供足夠的非線性程度[10]；模糊理論與類神經網路相結合之模糊類神經網路（Fuzzy Neural Network）是目前熱門的研究課題，Nie 和 Linkens 提出具有自組織能力之模糊化小腦模型控制器[11]。在應用方面有，水質控制[5]、核子反應爐控制[6]、伺服馬達控制[7]、電力變壓器故障診斷[8]、倒立單擺控制[39] 等。模糊控制是由模糊集合所衍生出來的控制理論，因此底下先由模糊集合開始來作概論式的介紹模糊控制理論。

2.2 模糊集合

一般的普通集合稱為明確集合（Crisp Set），明確集合是屬於二值邏輯，而模糊集合則是將二值邏輯延伸至多元邏輯，也就是說模糊集合是明確集合概念的推廣，明確集合是模糊集合的特例。吾人可以利用特徵函數（Characteristic Function）和歸屬函數（Membership Function）來說明兩者之間的差異，因此底下先說明特徵函數與歸屬函數的定義。

一．特徵函數

設 A 是論域（Universe of Discourse） X 中的一個集合，則對任意的 $x \in X$ ，令 $\Phi_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$ ，則稱 $\Phi_A(x)$ 為集合 A 的特徵函數。其中，特徵函數 $\Phi_A(x)$ 的值只有 1 或 0，亦即 $\Phi_A(x) = 1$ 表示元素 x 屬於集合 A ，否則 $\Phi_A(x) = 0$ 表示元素 x 不屬於集合 A 。

二．歸屬函數

定義於論域 X 上的模糊集合 \tilde{A} ，都可以用一個歸屬函數 $\mu_{\tilde{A}}(x)$ 來表示。歸屬函數 $\mu_{\tilde{A}}(x)$ 的值在 $[0,1]$ 這個區間內，稱為歸屬度（Membership Grade），歸屬度代表元素 x 屬於模糊集合 \tilde{A} 的程度。當 $\mu_{\tilde{A}}(x) = 1$ 時，則 x 完全屬於 \tilde{A} ；當 $\mu_{\tilde{A}}(x) = 0$ 時，則 x 完全不屬於 \tilde{A} ；而當 $\mu_{\tilde{A}}(x) = 0.5$ 時，則是最模糊。

如圖 2-1 所示為明確集合與模糊集合之差異性比較，一個代表「年青人」的集合，論域為年齡大小，由圖 2-1(a)來看十七歲完全不屬於「年青人」，但是十八歲又完全屬於「年青人」，此種明確界定的描述方式，顯然有些不合理；若由圖 2-1(b)來看「年青人」是一個模糊的年齡描述，23 歲完全屬於「年青人」，而 18 歲和 28 歲屬於年青人的程度降為 0.5，因此用模糊集合來描述「年青人」就顯得恰當許多。

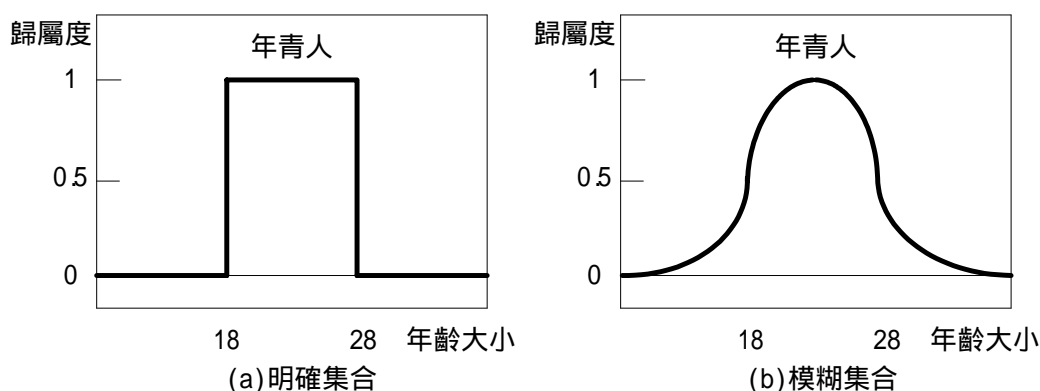


圖 2-1 明確集合與模糊集合的差異比較

三．模糊集合的表示式

模糊集合的表示式分為離散型及連續型兩種：

1. 離散型：當論域 $X = \{x_1, x_2, \dots, x_n\}$ 是有限元素的集合時，

$$\tilde{A} = \sum_{i=1}^n \frac{\mu_{\tilde{A}}(x_i)}{x_i} \quad (2-1)$$

2. 連續型：當論域 X 是由連續的數值所形成的集合時，

$$\tilde{A} = \int_X \frac{\mu_{\tilde{A}}(x)}{x} \quad (2-2)$$

2.2.1 模糊集合之基本性質

一． α -截集 (α -Cut)

模糊集合 \tilde{A} 的 α -截集係一明確集合，以符號 \tilde{A}_{α} 來表示，定義為在論域 X

中，歸屬度大於或等於 α 的所有元素的集合，表示為：

$$A_{\sim\alpha} = \{x | \mu_{\sim A}(x) \geq \alpha, x \in X\} \quad (2-3)$$

二．支集 (Support)

模糊集合 $\sim A$ 的支集，定義為在論域 X 中，歸屬度大於0的所有元素的集合，表示為：

$$Supp(\sim A) = \{x | \mu_{\sim A} > 0, x \in X\} \quad (2-4)$$

三．核心 (Core 或 Kernel)

模糊集合 $\sim A$ 的核心，定義為在論域 X 中，歸屬度等於1的所有元素的集合，表示為：

$$Core(\sim A) = \{x | \mu_{\sim A}(x) = 1, x \in X\} \quad (2-5)$$

四．高度 (Height)

模糊集合 $\sim A$ 的高度，定義為在論域 X 中的最大歸屬度，表示為：

$$Height(\sim A) = \max_{x \in X} \mu_{\sim A}(x) \quad (2-6)$$

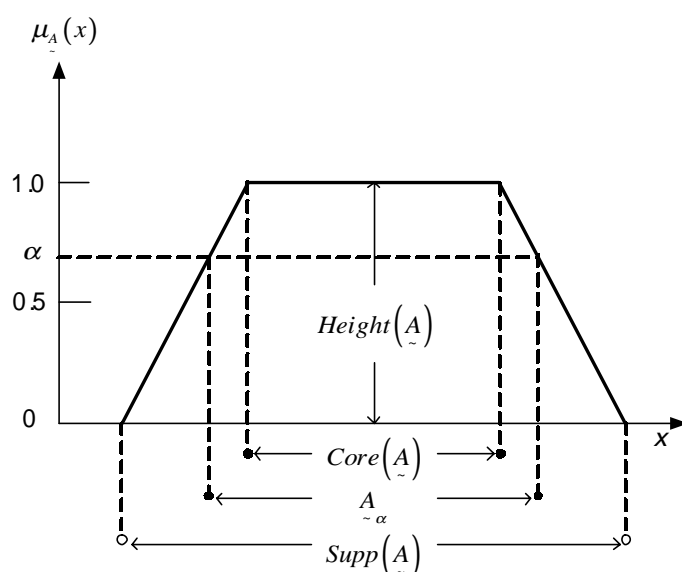


圖 2-2 模糊集合基本性質之示意圖

五．語意變數 (Linguistic Variables)

語意變數是指在指定的論域之下用來描述一些自然語言的模糊集合，語意變數的論域可以和明確集合的論域相同，但是論域之下則為許多模糊子集合的彙總，語意變數是以自然語言中的字句為值，稱為語意值 (Linguistic Value)，而不是以數為值的變數[35]。如圖 2-3 所示，一部汽車的「車速」就是一個語意變數，令車速的範圍是時速 0~240(km/hr)，吾人可以主觀地在論域[0,240]下，定義三個語意值：慢、中、快。

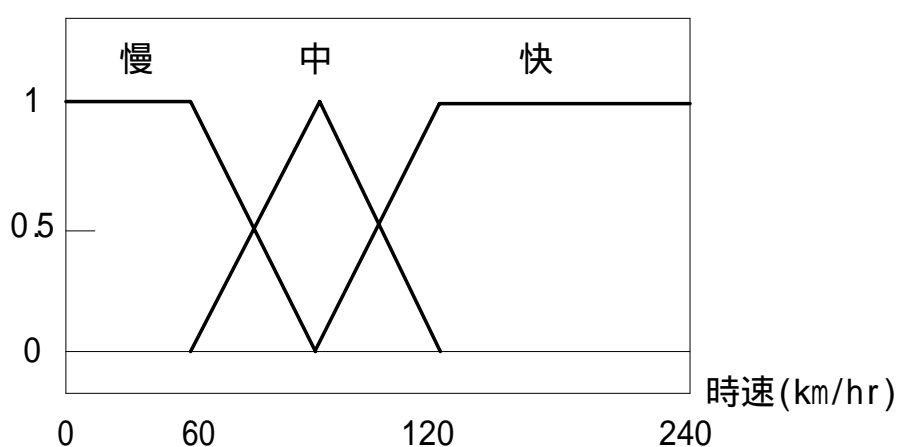


圖 2-3 車速的歸屬函數圖

2.2.2 模糊集合之基本運算

傳統明確集合中交集、聯集與補集的運算法則是唯一的；然而在模糊集合中，此三個基本運算法則並非唯一的。因為模糊集合的歸屬函數可能是介於[0,1]區間中的任何數值，因此模糊集合中交集、聯集與補集的運算必須加以修正，以適應模糊集合的特性[41]。

一．模糊交集 (Fuzzy Intersection)

Zadeh 最早提出之模糊集合的交集運算是取最小值，若令兩個模糊集合為 \tilde{A} 和 \tilde{B} ，則其交集 $\tilde{C} = \tilde{A} \cap \tilde{B}$ 的歸屬函數可由下式表示：

$$\tilde{C} = \mu_{\tilde{A} \cap \tilde{B}}(x) = \wedge \left[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) \right] = \min \left[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) \right] \quad (2-7)$$

其中 \wedge 表示取最小值運算。

以式(2-7)進行交集運算可以使得明確集合維持原來之運算結果，但是除了取最小值運算外，亦可採用其他方式，這些運算方式統稱為三角範數 (Triangular norm, 簡稱 T-norm)，如表 2-1 為常用之三角範數。

表 2-1 常用之三角範數

中文名稱	英文名稱	運算方式
最小值	Minimum	$T_{\min}(a, b) = a \wedge b = \min(a, b)$
幾何積	Algebraic Product	$T_{ap}(a, b) = a \cdot b = ab$
邊界積	Bounded Product	$T_{bp}(a, b) = a \odot b = \max(0, a + b - 1)$
激烈積	Drastic Product	$T_{dp}(a, b) = a \oslash b = \begin{cases} a, & b = 1 \\ b, & a = 1 \\ 0, & a, b < 1 \end{cases}$

二．模糊聯集 (Fuzzy Union)

Zadeh 最早提出之模糊集合的聯集運算是取最大值，若令兩個模糊集合為 \tilde{A} 和 \tilde{B} ，則其聯集 $\tilde{C} = \tilde{A} \cup \tilde{B}$ 的歸屬函數可由下式表示：

$$\tilde{C} = \mu_{\tilde{A} \cup \tilde{B}}(x) = \vee \left[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) \right] = \max \left[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) \right] \quad (2-8)$$

其中 \vee 表示取最大值運算。

以式(2-8)進行聯集運算可以使得明確集合維持原來之運算結果，但是除了取最大值運算外，亦可採用其他方式，這些運算方式統稱為反三角範數 (Triangular conorm, 簡稱 T-conorm 或 S-norm)，如表 2-2 為常用之反三角範數。

表 2-2 常用之反三角範數

中文名稱	英文名稱	運算方式
最大值	Maximum	$S_{\max}(a,b) = a \vee b = \max(a,b)$
幾何和	Algebraic Sum	$S_{as}(a,b) = a \hat{+} b = a + b - ab$
邊界和	Bounded Sum	$S_{bs}(a,b) = a \oplus b = \min(1, a+b)$
激烈和	Drastic Sum	$S_{ds}(a,b) = a \cup b = \begin{cases} a, & b = 0 \\ b, & a = 0 \\ 1, & a, b > 0 \end{cases}$

三 . 模糊補集 (Fuzzy Complement)

\bar{A} 是模糊集合 A 的補集，則 \bar{A} 的歸屬函數可由下式表示：

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (2-9)$$

2.3 模糊推論

模糊推論 (Fuzzy Inference) 是依據近似推理 (Approximate Reasoning) 的概念發展而來，相較於傳統的嚴正推理 (Exact Reasoning) 更合理且更具彈性[35]。近似推理有二個重要的模糊蘊含推論規則 (Fuzzy Implication Inference Rules) [4]，一個稱為廣義的肯定前件論式 (Generalized Modus Ponens, GMP)，又稱為前向推理 (Forward Reasoning)；另一個稱為廣義的否定後件論式 (Generalized Modus Tollens, GMT)，又稱為後向推理 (Backward Reasoning)。事實上，廣義的肯定前件論式是肯定前件論式 (Modus Ponens) 的一般化，而廣義的否定後件論式是肯定後件論式 (Modus Tollens) 的一般化，分別以表 2-3 及表 2-4 來加以描述。

表 2-3 廣義的肯定前件論式

描 述	實 例
前題 1(規則) : IF x is \underline{A} , THEN y is \underline{B}	如果天下雨, 則地是濕的
前題 2(事實) : x is \underline{A}'	天下雨了
推理結果 : y is \underline{B}'	地是濕的

表 2-4 廣義的否定後件論式

描 述	實 例
前題 1(規則) : IF x is \underline{A} , THEN y is \underline{B}	如果天下雨, 則地是濕的
前題 2(事實) : y is \underline{B}'	地是乾的
推理結果 : x is \underline{A}'	天沒有下雨

此處的 x 和 y 是語意變數, \underline{A} 、 \underline{A}' 、 \underline{B} 、 \underline{B}' 都是模糊集合, 由於 GMP 的推理方式是接受輸入資訊後推理出結果, 因此比較適用於模糊控制系統。

2.3.1 模糊推論方式

在一個模糊控制系統中, 控制規則可以用模糊關係來表示, 而模糊推論則以合成運算的方式來完成, 合成運算的方式有很多種, 不同方式的合成運算所得到的結果也會有所差異。目前模糊推論的方法雖然有許許多多, 但大體來說可以分為直接推論法和間接推論法兩種[34], 由於直接法在模糊控制裡使用最多, 底下僅作直接法的說明。

$$\begin{aligned}
 &\text{規 則 : IF } x \text{ is } \underline{A} , \text{ THEN } y \text{ is } \underline{B} \\
 &\text{事 實 : } x \text{ is } \underline{A}' \\
 &\text{結 論 : } y \text{ is } \underline{B}' \qquad (2-10)
 \end{aligned}$$

其中「 x is \tilde{A} 」稱為前件部（前題部），而「 y is \tilde{B} 」稱為後件部（結論部）。所謂的「直接推論法」就是將控制規則利用模糊關係 \tilde{R} 來表示後，再將模糊關係 \tilde{R} 與事實 \tilde{A}' 透過合成運算的方式推論得到結論 \tilde{B}' ，其演算過程可以由圖 2-4 來表示。

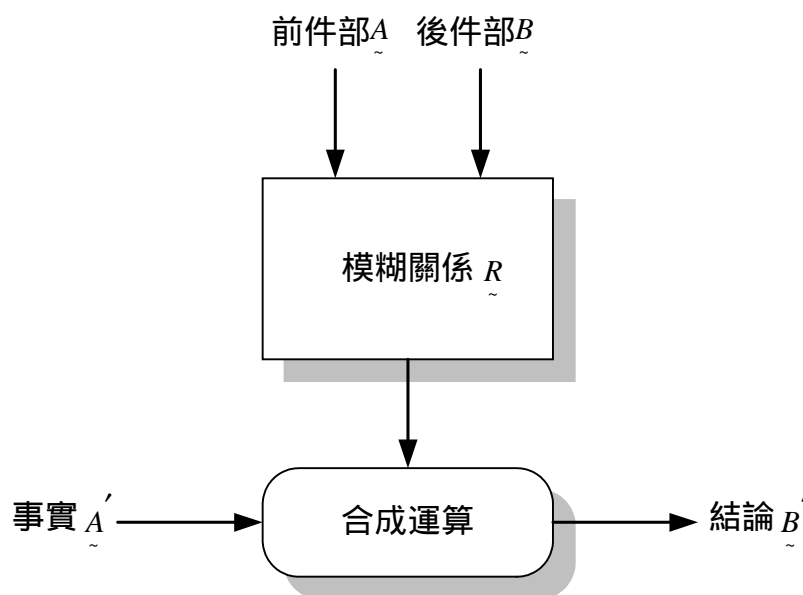


圖 2-4 直接推論法的演算過程

下面針對兩種常用之模糊推論法：Mamdani 模糊推論法和 Takagi-Sugeno 模糊推論法來加以說明其推論過程。

一 . Mamdani 模糊推論法

Mamdani 模糊推論法是模糊推論法中最常使用的一種，為了簡化說明起見，假設一個二輸入單輸出的模糊控制系統，且二條模糊控制規則分別表示成式(2-11)和(2-12)，其推論過程示意圖，如圖 2-5 所示。

$$R_1 : \text{IF } x_1 \text{ is } \tilde{A}_{\sim 11} \text{ and } x_2 \text{ is } \tilde{A}_{\sim 12} , \text{ THEN } y_1 \text{ is } \tilde{B}_{\sim 1} \quad (2-11)$$

$$R_2 : \text{IF } x_1 \text{ is } \tilde{A}_{\sim 21} \text{ and } x_2 \text{ is } \tilde{A}_{\sim 22} , \text{ THEN } y_2 \text{ is } \tilde{B}_{\sim 2} \quad (2-12)$$

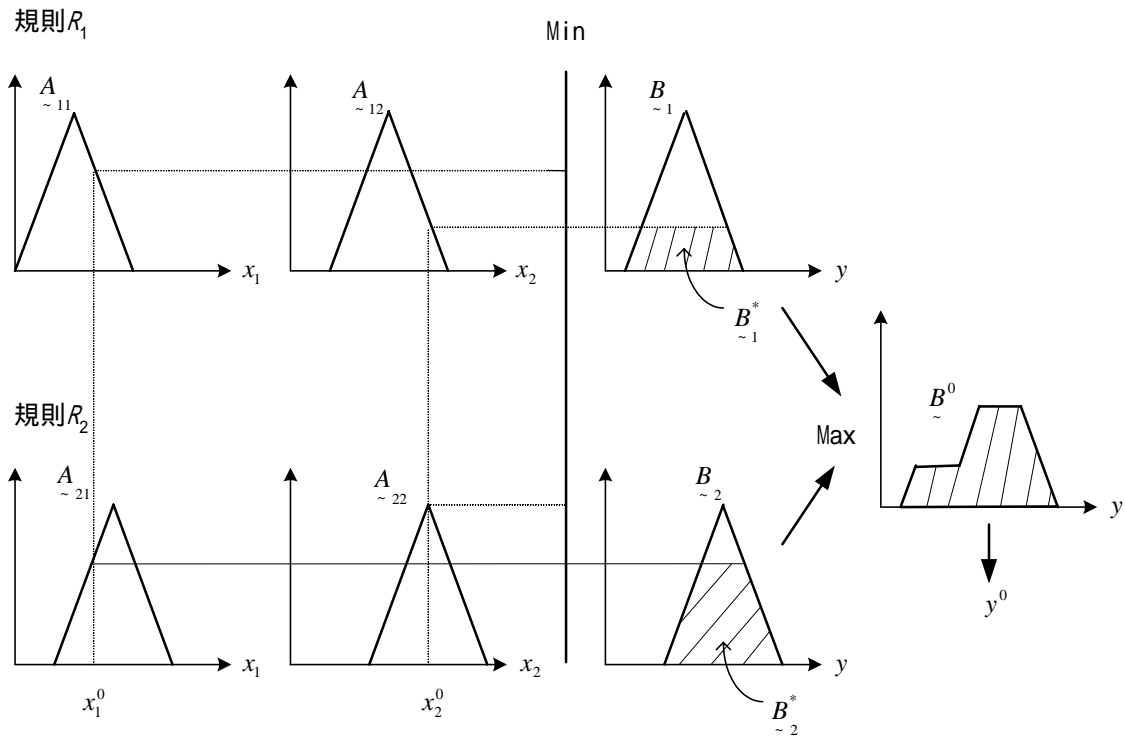


圖 2-5 Mamdani 模糊推論法示意圖

整個推論步驟之說明如下：

1. 由已知輸入值為 x_1^0 、 x_2^0 ，計算各規則前件部之適合度 w_i ，即：

$$w_i = \mu_{A_{i1}}(x_1^0) \wedge \mu_{A_{i2}}(x_2^0) \quad (2-13)$$

2. 決定各規則的推論結果 $B_{\sim i}^*$ ，即：

$$\mu_{B_{\sim i}^*}(y) = w_i \wedge \mu_{B_i}(y) \quad (2-14)$$

3. 統合所有規則求得 B_{\sim}^0 ，並以重心法來解模糊化求得明確輸出值 y^0 ，亦

即：

$$\mu_{B_{\sim}^0}(y) = \mu_{B_{\sim 1}^*}(y) \vee \mu_{B_{\sim 2}^*}(y) \quad (2-15)$$

$$y^0 = \frac{\int \mu_{B_{\sim}^0}(y) y dy}{\int \mu_{B_{\sim}^0}(y) dy} \quad (2-16)$$

二 . Takagi-Sugeno 模糊推論法

Takagi-Sugeno 模糊推論法的每個控制規則的前件部與 Mamdani 模糊推論法相同，而特點是其後件部為輸入變數之線性組合，同樣假設一個二輸入單輸出的模糊控制系統，且二條模糊控制規則分別表示成式(2-17)和(2-18)，其推論過程示意圖，如圖 2-6 所示。

$$R_1 : \text{IF } x_1 \text{ is } A_{\sim 11} \text{ and } x_2 \text{ is } A_{\sim 12} , \text{ THEN } y_1 = a_1x_1 + b_1x_2 + c_1 \quad (2-17)$$

$$R_2 : \text{IF } x_1 \text{ is } A_{\sim 21} \text{ and } x_2 \text{ is } A_{\sim 22} , \text{ THEN } y_2 = a_2x_1 + b_2x_2 + c_2 \quad (2-18)$$

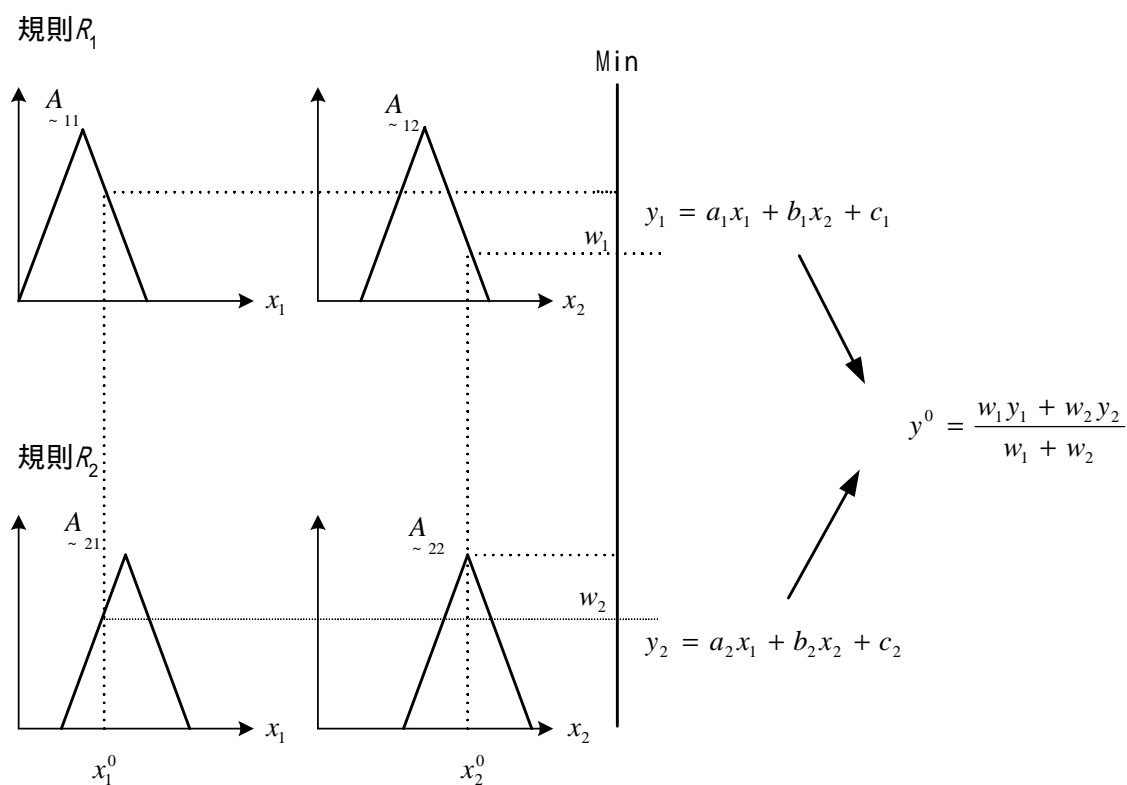


圖 2-6 Takagi-Sugeno 模糊推論法示意圖

整個推論步驟之說明如下：

1. 由已知輸入值為 x_1^0 、 x_2^0 ，計算各規則前件部之適合度 w_i ，即：

$$w_i = \mu_{A_{i1}}(x_1^0) \wedge \mu_{A_{i2}}(x_2^0) \quad (2-19)$$

2. 計算每一規則的推論值，即：

$$y_i = a_i x_1 + b_i x_2 + c_i \quad (2-20)$$

3. 統合所有規則以加權平均法來解模糊化求得明確輸出值 y^0 ，亦即：

$$y^0 = \frac{\sum_{i=1}^n w_i y_i}{\sum_{i=1}^n w_i} \quad (2-21)$$

2.4 模糊控制

模糊控制簡單地說就是將專家的知識和操作人員的經驗，表示成「IF—THEN」的語意式控制規則後，再利用模糊推論法則得到輸出量來達成控制目的，因此模糊邏輯控制器不像傳統控制器需要具有分析控制系統的數學工具，茲歸納其特點如下[4,33]：

1. 不用數值而用語意式的模糊變數來描述系統特徵。
2. 利用條件式控制法則來描述系統變數之間的關係。
3. 不必事先對受控體建立完整的數學模型。
4. 具有較佳的適應性、強健性及容錯能力。
5. 特別適用於非線性、時變和模型不確定的系統。

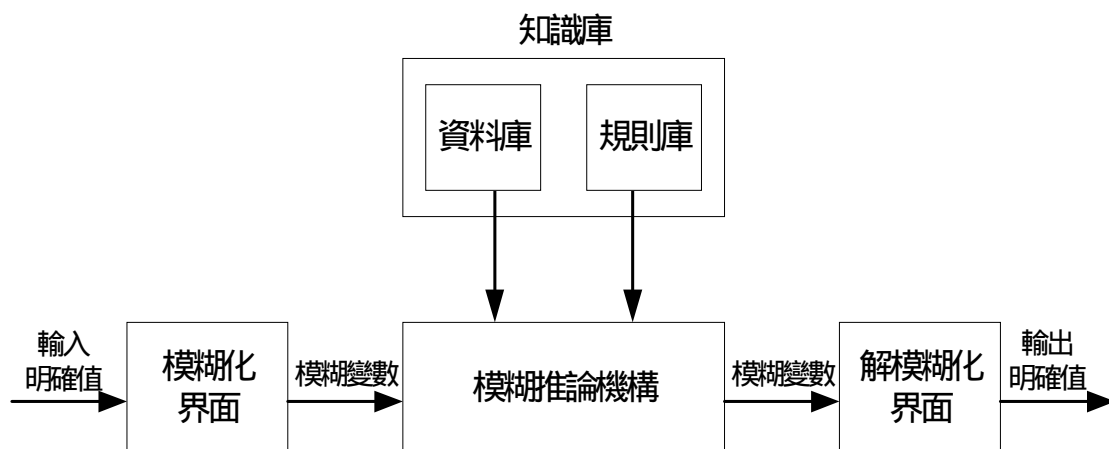


圖 2-7 模糊邏輯控制器方塊圖

如圖 2-7 所示為模糊邏輯控制器方塊圖，其中包括四個主要單元：模糊化界面、知識庫、模糊推論機構與解模糊化界面。一般而言，模糊邏輯控制器（FLC）的設計可分成四大步驟，分別說明如下：

步驟一：定義輸入及輸出變數

首先要決定受控系統有那些操作狀態是必須被量測的，以及那些輸出動作是必須被控制的。通常將 FLC 輸入變數的個數稱為維數，從理論上來講，FLC 的維數愈高，控制的效果愈好，但是維數高的 FLC 實現起來相當複雜和困難；而維數低的 FLC，控制效果又不理想。因此，目前大都使用二維 FLC，其控制精度即可滿足一般要求，一般輸入變數選擇誤差和誤差的變化，輸出變數選擇控制量的變化。決定好輸出入變數後，最後必須考量其實際操作範圍，經由比例因子（Scaling Factor）轉換到適當的論域。

步驟二：決定模糊化（Fuzzification）的策略

安排每個模糊變數的語意值，亦即將每個輸出入變數分別切割成數個模糊子集合，如：正大（PB）、正中（PM）、正小（PS）、零（ZO）、負小（NS）、負中（NM）、負大（NB），並決定其相對應的歸屬函數。訂定歸屬函數的適當與否，對於控制器的效能有著重大影響，在一般理論的研究與探討中，常用的歸屬函數形式有三角形、梯形、高斯函數形和鐘形四種，分別說明如下：

1. 三角形歸屬函數：

三角形歸屬函數具有構造簡單、運算容易，同時在適當的安排下，可以簡化理論分析之複雜度，以符號 $\Lambda(x; a, b, c)$ 來表示，其數學公式為：

$$\Lambda(x; a, b, c) = \begin{cases} 0 & , x \leq a \\ \frac{x-a}{b-a} & , a \leq x \leq b \\ \frac{c-x}{c-b} & , b \leq x \leq c \\ 0 & , c \leq x \end{cases} \quad (2-22)$$

其中， a 、 b 、 c 為實數值之參數且 $a < b < c$ 。

2. 梯形歸屬函數

梯形歸屬函數類似於三角形歸屬函數，以符號 $\Pi(x; a, b, c)$ 來表示，其數學公式為：

$$\Pi(x; a, b, c, d) = \begin{cases} 0 & , x \leq a \\ \frac{x-a}{b-a} & , a \leq x \leq b \\ 1 & , b \leq x \leq c \\ \frac{d-x}{d-c} & , c \leq x \leq d \\ 0 & , d \leq x \end{cases} \quad (2-23)$$

其中， a 、 b 、 c 、 d 為實數值之參數且 $a < b < c < d$ 。

3. 高斯函數歸屬函數

高斯函數歸屬函數與梯形歸屬函數的最大不同點是，梯形歸屬函數的支點在論域中為有界集合，而高斯函數歸屬函數的支點則涵蓋整個論域，以符號 $G(x; m, \sigma)$ 來表示，其數學公式為：

$$G(x; m, \sigma) = e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2-24)$$

其中， m 代表高斯函數的中心值， σ 代表高斯函數的標準差。

4. 鐘形歸屬函數

鐘形歸屬函數以符號 $B(x; a, b, c)$ 來表示，其數學公式為：

$$B(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2-25)$$

其中， a 、 b 、 c 為實數值之參數。

步驟三：模糊規則之建立與推論

模糊規則庫是由一條一條的「IF—THEN」形式的表示式所組成，一般規則庫的建立方式，是直接將專家的知識與現場操作人員的經驗，轉化成語意式的控制規則，此法的缺點是無法做線上修正。因此，可以結合類神經網路來做線上即時的規則庫修改，以適應環境變化；模糊推論是模糊控制的核心，目前的模糊推論法則有許多不同的方法，如 2.3.1 節所介紹的 Mamdani 模糊推論法和 Takagi-Sugeno 模糊推論法等，視不同的情況而有不同的選擇。

步驟四：決定解模糊化 (Defuzzification) 的策略

模糊推論所得到的結果，最後必須經由解模糊化方法，將模糊量轉換成明確的輸出控制量，才能用來實際下命令給致動器。以下係針對常用的兩種解模糊化法則來說明：重心法 (Center of Gravity) 和高度法 (Height) [36]。

1. 重心法

重心法是求推論結果陰影面積的重心，並以其對應的元素為輸出操作量，這是最常用的，也是似乎最合理的，可惜的是計算時間稍長，其數學表示式如下：

(1) 當論域是連續時，

$$y^0 = \frac{\int_Y \mu_{\tilde{B}}(y) \cdot y dy}{\int_Y \mu_{\tilde{B}}(y) dy} \quad (2-26)$$

(2) 當論域是離散時，

$$y^0 = \frac{\sum_{i=1}^l \mu_{\tilde{B}}(y_i) \cdot y_i}{\sum_{i=1}^l \mu_{\tilde{B}}(y_i)} \quad (2-27)$$

其中， l 代表輸出模糊變數 y 的量化數， y^0 代表解模糊化後的明確輸出量， y_i 代表輸出變數 y 第 i 個量化值， $\mu_{\tilde{B}}(y_i)$ 代表 y_i 屬於模糊集合 \tilde{B} 的歸屬度。

2. 高度法

高度法是以被觸動規則前件部的適合度為後件部的高度，計算方法相當簡單，不像重心法需要動用到積分來解決，因此，使用此法的運算效率很高。

$$y^0 = \frac{\sum_{i=1}^n h(B_{\sim i}) \cdot y_i}{\sum_{i=1}^n y_i} \quad (2-28)$$

其中， n 代表觸發到的規則數， $h(B_{\sim i})$ 代表後件部模糊集合 $B_{\sim i}$ 的高度， y_i 代表後件部模糊集合 $B_{\sim i}$ 在未經砍頭前之最高點的 y 值。