PHAN XUÂN MINH
NGUYỄN DOÃN PHƯỚC

Lý thuyết ĐIỀU KHIỂN MỜ

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

Phan Xuân Minh & Nguyễn Doãn Phước

LÝ THUYẾT
ĐIỀU KHIỂN MỜ

(IN LẦN THỨ 5, CÓ SỬA ĐỔI VÀ BỔ SUNG)

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

Hà Nội - 2006

Chịu trách nhiệm xuất bản: **PGS. TS. Tô Đăng Hải**
Biên tập: **Nguyễn Thị Ngọc Khuê**
Vẽ bìa: **Trần Thắng**

In 1000 cuốn khổ 16 x 24 cm, tại Công ty cổ phần In Hàng không
Quyết định xuất bản số: 409-2006/CXB/9-33/KHKT
In xong và nộp lưu chiểu Quý 4 năm 2006

Lời nói đầu

Ngay từ khi được xuất bản lần đầu vào năm 1997, cuốn sách ***Lý thuyết điều khiển mờ*** *đã được đông đảo bạn đọc xa gần hưởng ứng đón nhận và gửi về cho nhóm tác giả chúng tôi nhiều những ý kiến đóng góp. Điều này đã khích lệ chúng tôi rất nhiều và cũng dựa vào các ý kiến đóng góp đó, chúng tôi đã sửa đổi, bổ sung để đến lần xuất bản thứ 4 này với nội dung gần gũi, thân thiện hơn với bạn đọc, cũng như phù hợp hơn với xu thế phát triển chung hiện nay của chuyên ngành* ***Điều khiển mờ****, chẳng hạn như từ lần in lại lần thứ 3 chúng tôi đã bổ sung thêm phần* ***Điều khiển mờ và mạng nơ-ron*** *là xu hướng được sử dụng khá rộng rãi trong các bài toán điều khiển phi tuyến hiện nay. Một lần nữa chúng tôi chân thành cảm ơn các bạn đọc đã quan tâm và gửi cho chúng tôi những thiện ý quý báu đó.*

Có thể nói, ngay từ khi mới ra đời vào những năm đầu của thập kỷ 90, chuyên ngành điều khiển mờ đã được phát triển rất mạnh mẽ và đem lại nhiều thành tựu bất ngờ trong lĩnh vực điều khiển. Ưu điểm cơ bản của điều khiển mờ so với các phương pháp điều khiển kinh điển là có thể tổng hợp được bộ điều khiển mà không cần biết trước đặc tính của đối tượng một cách chính xác. Ngành kỹ thuật mới mẻ này, như Zahde đã định hướng cho nó vào năm 1965, có nhiệm vụ chuyển giao nguyên tắc xử lý thông tin, điều khiển của hệ sinh học sang hệ kỹ thuật. Khác hẳn với kỹ thuật điều khiển kinh điển là hoàn toàn dựa vào sự chính xác tuyệt đối của thông tin mà trong nhiều ứng dụng không cần thiết hoặc không thể có được, điều khiển mờ chỉ cần xử lý những thông tin "không chính xác" hay không đầy đủ, những thông tin mà sự chính xác của nó chỉ nhận thấy được giữa các quan hệ của chúng với nhau và cũng chỉ có thể mô tả được bằng ngôn ngữ, đã có thể cho ra những quyết định chính xác. Chính khả năng này đã làm cho điều khiển mờ sao chụp được phương thức xử lý thông tin và điều khiển của con người, đã giải quyết thành công các bài toán điều khiển phức tạp, các bài toán mà trước đây không giải quyết được và đã đưa nó lên vị trí xứng đáng là ***kỹ thuật điều khiển của hôm nay và tương lai****. Điều khiển mờ hay còn gọi là điều khiển "thông minh" là những bước ứng dụng ban đầu của trí tuệ nhân tạo vào kỹ thuật điều khiển.*

Cuốn sách ***Lý thuyết điều khiển mờ*** *đề cập đến các phương pháp toán học để tổng hợp và phân tích một hệ thống điều khiển mờ, cung cấp cho bạn đọc những kiến thức cơ bản nhất để có khả năng tự tổng hợp bộ điều khiển mờ đi từ đơn giản đến phức tạp.*

Nhân dịp được xuất bản lần 4, chúng tôi xin được một lần nữa gửi lời cám ơn chân thành đến GS.TSKH. P. Gadow, GS.TSKH. H. Schreiber, những người đã luôn khuyến *khích, ủng hộ và cung cấp cho chúng tôi các tài liệu cần thiết cho lần xuất bản đầu tiên, và tới những người thân trong gia đình, những người đã luôn cho chúng tôi tình cảm, sự khích lệ cũng như khoảng thời gian yên tĩnh để có thể hoàn thành được công việc.*

Chúng tôi rất mong tiếp tục nhận được những ý kiến phê bình và đóng góp của các bạn đọc. Thư góp ý xin gửi về:

Trường Đại học Bách khoa Hà Nội
Khoa Điện   Bộ môn Điều khiển tự động,
Số 1, đường Đại Cồ Việt, C9 / 305-306
Điện thoại: (04) 8680451 / (04) 8692985

Hà Nội, ngày 17 tháng 5 năm 2004
Các tác giả.

Mục lục

1 NHẬP MÔN

1.1 Bộ điều khiển "mờ" lý tưởng

Con người có một khả năng tuyệt vời là chỉ cần qua một quá trình học hỏi tương đối ngắn cũng có thể hiểu rõ và nắm vững một quá trình phức tạp. Khả năng này được chứng tỏ thường xuyên trong cuộc sống đời thường, cho dù bản thân con người không ý thức được điều đó. Hãy xét phản ứng của người cha trong một gia đình làm ví dụ, khi ông ta lái xe cùng gia đình đi nghỉ, trong đó người cha được xem như là thiết bị điều khiển và chiếc xe là đối tượng điều khiển. Biết rằng người cha, hay thiết bị điều khiển, có nhiệm vụ trọng tâm là điều khiển chiếc xe đưa gia đình tới đích, song để hiểu rõ được hơn phương thức thực hiện nhiệm vụ đó của người cha, cũng nên cần xem xét ông ta phải xử lý những thông tin gì và xử lý chúng như thế nào.

Đại lượng điều khiển thứ nhất là con đường trước mặt. Người cha có nhiệm vụ điều khiển chiếc xe đi đúng phần đường quy định, tức là phải luôn giữ cho xe nằm trong phần đường bên phải kể từ vạch phân cách, trừ những trường hợp khi phải vượt xe khác. Để làm được công việc đó, thậm chí người cha cũng không cần phải biết một cách chính xác rằng xe của ông hiện thời cách vạch phân cách bao nhiêu centimeter, chỉ cần nhìn vào con đường trước mặt, ông ta cũng có thể suy ra được rằng xe hiện đang cách vạch phân cách nhiều hay ít và từ đó đưa ra quyết định phải đánh tay lái sang phải mạnh hay nhẹ.

Đại lượng điều khiển thứ hai là tốc độ của xe. Với nguyên tắc, để các thành viên gia đình trên xe cảm thấy chuyến đi được thoải mái và cũng để tiết kiệm xăng, người cha có nhiệm vụ giữ nguyên tốc độ xe, tránh không phanh hoặc tăng tốc khi không cần thiết. Giá trị về tốc độ của xe mà người cha phải giữ cũng phụ thuộc nhiều vào môi trường xung quanh như thời tiết, cảnh quan, mật độ xe trên đường ... và cũng còn phụ thuộc thêm là ông ta có quen con đường đó hay không? Tuy nhiên quy luật điều khiển này cũng không phải cố định. Giả sử trước mặt có một xe khác đi chậm hơn, vậy thì thay cho nhiệm vụ giữ nguyên tốc độ, người cha phải tạm thời thực hiện một nhiệm vụ khác là giảm tốc độ xe và tự điều khiển xe theo một tốc độ mới, phù hợp với sự phản ứng của xe trước cho tới khi ông ta vượt được xe đó.

Ngoài những đại lượng điều khiển trên mà người cha phải đưa ra, ông ta còn có nhiệm vụ theo dõi tình trạng xe như phải tìm hiểu xem nước làm mát máy có bị nóng quá không?, áp suất dầu thấp hay cao ··· để từ đó có thể phân tích, nhận định kịp thời các lỗi của xe.

Đối tượng điều khiển là chiếc xe cũng có những tham số thay đổi cần phải được theo dõi và thu thập thường xuyên cho công việc ra các quyết định về đại lượng điều khiển. Các tham số đó là áp suất hơi trong lốp, nhiệt độ máy .... Sự thay đổi các tham số đó, người cha nhận biết được có thể trực tiếp qua các đèn báo hiệu trong xe, song cũng có thể gián tiếp qua phản ứng của xe với các đại lượng điều khiển.

Người cha, trong quá trình lái xe, đã thực hiện tuyệt vời chức năng của một bộ điều khiển, từ thu thập thông tin, thực hiện thuật toán điều khiển (trong đầu) cho đến đưa ra tín hiệu điều khiển kịp thời mà *không cần phải biết một cách chính xác về vị trí, tốc độ, tình trạng... của xe*. Hoàn toàn ngược lại với khái niệm điều khiển chính xác, người cha cũng chỉ cần đưa ra những đại lượng điều khiển theo *nguyên tắc xử lý "mờ"* như:

- nếu xe hướng nhẹ ra vạch phân cách thì đánh tay lái nhẹ sang phải,
- nếu xe hướng đột ngột ra ngoài vạch phân cách thì đánh mạnh tay lái sang phải,
- nếu đường có độ dốc lớn thì về số,
- nếu đường thẳng, khô, tầm nhìn không bị hạn chế và tốc độ chỉ hơi cao hơn bình thường một chút thì không cần giảm tốc độ.

...

Các nguyên lý điều khiển "mờ" như vậy, tuy chúng có thể khác nhau về số các mệnh đề điều kiện, song đều cùng có một cấu trúc:

"NẾU *điều kiện* 1 VÀ ··· VÀ *điều kiện n* THÌ

quyết định 1 VÀ ··· VÀ *quyết định m*".

Vậy bản chất nguyên lý điều khiển mờ như người cha đã làm và thể hiện bằng thuật toán xử lý xe của ông như thế nào?, có những hình thức nào để xây dựng lại được mô hình điều khiển theo nguyên lý điều khiển "mờ" của người cha khi lái xe?, làm cách nào để có thể tổng quát hóa chúng thành một nguyên lý điều khiển mờ chung và từ đó áp dụng cho các quá trình tương tự?. Câu trả lời sẽ là nội dung của toàn bộ quyển sách này.

Trên cơ sở kiến thức đã có về điều khiển tự động, quyển sách này sẽ lần lượt giới thiệu với độc giả những khái niệm, bản chất và các phương pháp tổng hợp chính các bộ điều khiển mờ cũng như ứng dụng của chúng.

1.2 Khái niệm về tập mờ

1.2.1 Nhắc lại về tập hợp kinh điển

Khái niệm về tập hợp được hình thành trên nền tảng logic và được *G.Cantor* định nghĩa như là một sự xếp đặt chung lại các vật, các đối tượng có cùng chung một tính chất, được gọi là *phần tử* của tập hợp đó. Ý nghĩa logic của khái niệm tập hợp được xác định ở chỗ *một vật hoặc một đối tượng bất kỳ chỉ có thể có hai khả năng hoặc là phần tử của tập đang xét hoặc không*.

Cho một tập hợp A. Một phần tử x thuộc A được ký hiệu bằng $x\in A$. Ngược lại ký hiệu $x\notin A$ dùng để chỉ x không thuộc tập hợp A. Một tập hợp không có một phần tử nào được gọi là *tập rỗng*. Ví dụ tập hợp các số thực x thỏa mãn phương trình $x^2+1=0$ là một tập rỗng. Tập rỗng được ký hiệu bằng $\varnothing$.

Có nhiều cách để biểu diễn một tập hợp. Cách biểu diễn dễ chấp nhận hơn cả là cách liệt kê những phần tử của tập hợp, ví dụ

$A_1 = \{1, 2, 3, 5, 7, 11\}$ hoặc

$A_2 = \{$cây, 4, nhà, φ, xe máy$\}$.

Tuy nhiên cách này sẽ tỏ ra bất tiện khi phải biểu diễn những tập hợp có nhiều phần tử (hoặc vô số các phần tử). Thường dùng nhất là cách biểu diễn thông qua tính chất tổng quát của các phần tử. Một phần tử x thuộc A khi và chỉ khi nó thỏa mãn tính chất tổng quát này, ví dụ

$A_1 = \{ x \mid x$ là số nguyên tố $\}$ hoặc

$A_2 = \{ x \mid x$ là số thực và $x < 4\}$.

Sau đây là một số ký hiệu thường dùng của các tập hợp quen biết:

a) $\mathbb{N}$ - tập hợp các số tự nhiên,

b) $\mathbb{R}$ - tập hợp các số thực,

c) $\mathbb{Q}$ - tập hợp các số thực hữu tỷ,

d) $\mathbb{C}$ - tập hợp các số phức.

Cho hai tập hợp A và B. Nếu mọi phần tử của A cũng là phần tử của B thì tập A được gọi là *tập con* của B và ký hiệu bằng $A \subseteq B$. Ngoài ra nếu như còn được biết thêm rằng trong B có ít nhất một phần tử không thuộc A thì A được gọi là *tập con thực* của B và ký hiệu bằng $A \subset B$.

Hai tập hợp A và B cùng đồng thời thỏa mãn $A \subseteq B$ và $B \subseteq A$ thì được nói là chúng bằng nhau và ký hiệu $A=B$. Với hai tập hợp bằng nhau, mọi phần tử của tập này cũng là phần tử của tập kia và ngược lại.

Cho một tập hợp A. Ánh xạ $\mu_A : A \rightarrow \mathbb{R}$ định nghĩa như sau

$$\mu_A(x) = \begin{cases} 1 & \text{nếu } x \in A \\ 0 & \text{nếu } x \notin A \end{cases} \tag{1.1a}$$

được gọi là *hàm thuộc* của tập A. Như vậy $\mu_A(x)$ chỉ nhận hai giá trị hoặc bằng 1 hoặc bằng 0. Giá trị 1 của hàm thuộc $\mu_A(x)$ còn được gọi là giá trị đúng, ngược lại 0 là giá trị sai của $\mu_A(x)$. Một tập X luôn có

$$\mu_X(x) = 1, \text{ với mọi } x$$

được gọi là *không gian nền* (*tập nền*).

Một tập A có dạng

$$A = \{x \in X \mid x \text{ thỏa mãn một số tính chất nào đó}\}$$

thì được nói là có tập nền X, hay được định nghĩa trên tập nền X. Ví dụ tập

$$A = \{x \in \mathbb{R} \mid 2 < x < 4\},$$

có tập nền là tập các số thực $\mathbb{R}$.

Với khái niệm tập nền như trên thì hàm thuộc μ_A của tập A có tập nền X sẽ được hiểu là ánh xạ μ_A: $X \rightarrow \{0,1\}$ từ X vào tập $\{0, 1\}$ gồm hai phần tử 0 và 1.

Có thể dễ dàng thấy được rằng $A \subseteq B$ khi và chỉ khi $\mu_A(x) \le \mu_B(x)$, tức là

$$A \subseteq B \quad \Leftrightarrow \quad \mu_A(x) \le \mu_B(x). \tag{1.2}$$

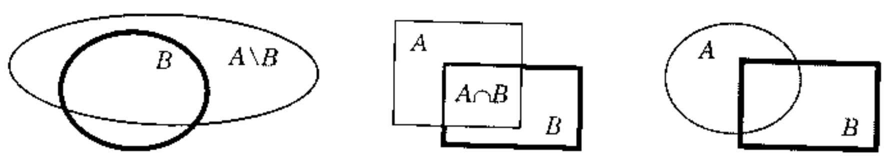
Thật vậy từ $A \subseteq B$ và $x \in A$ ta luôn có $x \in B$ và do đó $\mu_A(x)=\mu_B(x)=1$. Ngược lại khi $x \notin A$ ($\mu_A(x)=0$), chưa thể khẳng định được x có thuộc B hay không. Bởi vậy $\mu_B(x)$ có thể bằng 0 và cũng có thể bằng 1, nói cách khác $\mu_A(x) \le \mu_B(x)$ hay hàm thuộc $\mu(x)$ là *hàm không giảm*.

Hàm thuộc $\mu_A(x)$ với bốn phép toán trên tập hợp gồm phép hợp, giao, hiệu (*hình 1.1*) và phép bù có các tính chất sau:

Hiệu của hai tập hợp

Hiệu của hai tập hợp A, B có cùng một không gian nền X là một tập hợp, ký hiệu bằng $A \setminus B$, cũng được định nghĩa trên tập nền X, gồm các phần tử của A mà không thuộc B (*hình 1.1a*). Hàm thuộc $\mu_{A\setminus B}(x)$ của hiệu $A \setminus B$ chỉ nhận giá trị đúng ($\mu_{A\setminus B}(x)=1$) khi $x \in A$ và $x \notin B$, tức là khi $\mu_A(x)=1$ và $\mu_B(x)=0$. Ở các trường hợp khác nó sẽ nhận giá trị sai, hay $\mu_{A\setminus B}(x)=0$. Bởi vậy ta luôn có

$$\mu_{A\setminus B}(x) = \mu_A(x) - \mu_A(x)\mu_B(x). \tag{1.3}$$



a) $A \setminus B$ b) $A \cap B$ c) $A \cup B$

Hình 1.1: Các phép toán trên tập hợp.
a) Hiệu của hai tập hợp.
b) Giao của hai tập hợp.
c) Hợp của hai tập hợp.

Giao của hai tập hợp

Giao (hay còn gọi là phép hội các hàm thuộc) của hai tập hợp A, B có cùng không gian nền X là một tập hợp, ký hiệu bằng $A \cap B$, cũng được định nghĩa trên tập nền X, gồm các phần tử vừa thuộc A và vừa thuộc B (*hình 1.1b*). Hàm thuộc $\mu_{A\cap B}(x)$ của tập $A \cap B$ sẽ chỉ nhận giá trị 1 khi $x \in A$ và $x \in B$, tức là chỉ khi có đồng thời $\mu_A(x)=1$ và $\mu_B(x)=1$. Do đó ta được

$$\mu_{A\cap B}(x) = \mu_A(x)\mu_B(x). \tag{1.4}$$

Để ý rằng hàm thuộc chỉ có một trong hai giá trị 0 hoặc 1 nên còn có

$$\mu_{A\cap B}(x) = \min\{\mu_A(x),\ \mu_B(x)\}. \tag{1.5}$$

Nói cách khác, hai công thức (1.4), (1.5) là tương đương.

Ngoài ra, từ (1.4) và (1.5) ta cũng nhận thấy hàm thuộc $\mu_{A\cap B}(x)$ thỏa mãn các tính chất sau:

1) $\mu_{A\cap B}(x)$ chỉ phụ thuộc vào $\mu_A(x)$ và $\mu_B(x)$. (1.6a)

2) Nếu B là không gian nền, tức là mọi phần tử x đều thuộc B thì $A\cap B = A$, do đó

$$\mu_B(x) = 1 \quad \text{với mọi } x \quad \Rightarrow \quad \mu_{A\cap B}(x) = \mu_A(x). \tag{1.6b}$$

3) $\mu_{A\cap B}(x) = \mu_{B\cap A}(x)$, tức là phép giao có tính giao hoán. (1.6c)

4) Phép giao có tính kết hợp, tức là $(A\cap B)\cap C = A\cap(B\cap C)$. Suy ra

$$\mu_{(A\cap B)\cap C}(x) = \mu_{A\cap(B\cap C)}(x). \tag{1.6d}$$

5) Nếu $A_1 \subseteq A_2$ thì $A_1\cap B \subseteq A_2\cap B$. Thật vậy, từ $x\in A_1\cap B$ ta có $x\in A_1$ và $x\in B$ nên cũng có $x\in A_2$ và $x\in B$, hay $x\in A_2\cap B$. Từ kết luận trên và theo (1.2) ta có được

$$\mu_{A_1}(x) \le \mu_{A_2}(x) \quad \Rightarrow \quad \mu_{A_1\cap B}(x) \le \mu_{A_2\cap B}(x). \tag{1.6e}$$

Hợp của hai tập hợp

Hợp (hay còn gọi là phép tuyển) của hai tập hợp A, B có cùng không gian nền X là một tập hợp, ký hiệu bằng $A\cup B$, cũng được định nghĩa trên nền X, gồm các phần tử của A và của B (*hình 1.1c*). Hàm thuộc $\mu_{A\cup B}(x)$ của tập $A\cup B$ sẽ nhận giá trị 1nếu hoặc $x\in A$ hoặc $x\in B$, tức là hoặc $\mu_A(x)=1$ hoặc $\mu_B(x)=1$. Do đó

$$\mu_{A\cup B}(x) = \max\{\mu_A(x),\ \mu_B(x)\}. \tag{1.7}$$

Điều này cũng tương đương với

$$\mu_{A\cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x)\mu_B(x), \tag{1.8}$$

do hàm thuộc chỉ có một trong hai giá trị 0 hoặc 1.

Ngoài ra, hàm thuộc $\mu_{A\cup B}(x)$ xác định theo hai công thức (1.7) và (1.8) còn thỏa mãn các tính chất sau:

1) $\mu_{A\cup B}(x)$ chỉ phụ thuộc vào $\mu_A(x)$ và $\mu_B(x)$. (1.9a)

2) Nếu B là một tập rỗng, tức $B=\emptyset$ thì $A\cup B = A$, do đó

$$\mu_B(x) = 0 \quad \text{với mọi } x \quad \Rightarrow \quad \mu_{A\cup B}(x) = \mu_A(x). \tag{1.9b}$$

3) $\mu_{A\cup B}(x) = \mu_{B\cup A}(x)$, tức là phép hợp có tính giao hoán. (1.9c)

4) Phép hợp có tính kết hợp, tức là $(A\cup B)\cup C = A\cup(B\cup C)$. Suy ra

$$\mu_{(A\cup B)\cup C}(x) = \mu_{A\cup(B\cup C)}(x). \tag{1.9d}$$

5) Nếu $A_1\subseteq A_2$ thì $A_1\cup B\subseteq A_2\cup B$. Thật vậy, từ $x\in A_1\cup B$ ta có hoặc $x\in A_1$ hoặc $x\in B$ nên cũng có $x\in A_2$ hoặc $x\in B$, hay $x\in A_2\cup B$. Cùng với kết luận này ta có

$$\mu_{A_1}(x) \le \mu_{A_2}(x) \quad \Rightarrow \quad \mu_{A_1\cup B}(x) \le \mu_{A_2\cup B}(x). \tag{1.9e}$$

Bù của một tập hợp

Bù của một tập hợp A có không gian nền X, ký hiệu bằng A^C, là một tập hợp gồm các phần tử của X không thuộc A. Phép bù là một phép toán trên tập hợp có giá trị đúng nếu $x\notin A$ và sai nếu $x\in A$, tức là

$$\mu_{A^C}(x) = \begin{cases} 1 & \text{nếu } x\notin A \\ 0 & \text{nếu } x\in A \end{cases} \tag{1.10}$$

Bởi vậy

$$\mu_{A^C}(x) = 1 - \mu_A(x). \tag{1.11}$$

Tập bù A^C của A chính là hiệu $X\setminus A$ và có cùng không gian nền X như A.

Ta còn có thể suy được ra thêm rằng hàm thuộc $\mu_{A^C}(x)$ xác định theo hai công thức (1.10) và (1.11) thỏa mãn các tính chất sau:

1) $\mu_{A^C}(x)$ chỉ phụ thuộc vào $\mu_A(x)$. (1.12a)

2) Nếu $x\in A$ thì $x\notin A^C$, hay

$$\mu_A(x)=1 \quad \Rightarrow \quad \mu_{A^C}(x) = 0. \tag{1.12a}$$

3) Nếu $x\notin A$ thì $x\in A^C$, hay

$$\mu_A(x)=0 \quad \Rightarrow \quad \mu_{A^C}(x) = 1. \tag{1.12b}$$

4) Nếu $A \subseteq B$ thì $A^C \supseteq B^C$, tức là

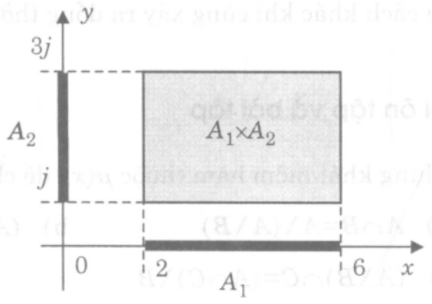
$$\mu_A(x) \le \mu_B(x) \Rightarrow \qquad \mu_{A^C}(x) \ge \mu_{B^C}(x). \tag{1.12c}$$

Công thức (1.12c) nói rằng hàm thuộc $\mu_{A^C}(x)$ là một *hàm không tăng*.

Tích của hai tập hợp

Tích $A \times B$ của *phép nhân* hai tập hợp A, B là một tập hợp mà mỗi phần tử của nó là một cặp (x, y), trong đó $x \in A$ và $y \in B$. Hai tập hợp A, B được gọi là tập thừa số của phép nhân. Trong trường hợp $A=B$ thì tích $A \times B$ thường được viết thành A^2 như các tập $\mathbb{R}^2$ (không gian Euclid 2 chiều) hay $\mathbb{C}^2$ (mặt phẳng phức).

Trong khi thực hiện phép nhân hai tập hợp A và B ta không cần phải giả thiết là chúng có chung một không gian nền. Nếu gọi X là tập nền của A và Y là tập nền của B thì tích $A \times B$ sẽ có tập nền là $X \times Y$.



Hình 1.2: Tích của hai tập hợp

Xét hai tập hợp

$A_1 = \{x \in \mathbb{R} \mid 2 \le x \le 6\}$ và

$A_2 = \{y \in j\mathbb{R} \mid j \le y \le 3j\}$.

Tích $A_1 \times A_2$ là một tập hợp được định nghĩa như sau (*hình 1.2*):

$$A_1 \times A_2 = \{(x, y) \mid x \in \mathbb{R}, y \in j\mathbb{R}, 2 \le x \le 6 \text{ và } j \le y \le 3j\} \tag{1.13}$$

và có không gian nền là tập các số phức $\mathbb{C} = \mathbb{R} \times j\mathbb{R}$.

Tích của hai tập hợp sẽ là một tập rỗng nếu như một trong hai tập thừa số là tập rỗng. Ngược lại nếu tích là tập rỗng thì ít nhất phải có một tập thừa số là tập rỗng:

$$A_1 \times A_2 = \varnothing \quad \Leftrightarrow \quad A_1 = \varnothing \text{ hoặc } A_2 = \varnothing.$$

Phép nhân tập hợp có thể thực hiện được trên nhiều tập hợp khác nhau. Ví dụ tích của n tập hợp $A_1, A_2, \cdots, A_n$ được hiểu là một tập hợp

$$A_1 \times A_2 \times \cdots \times A_n = \{(x_1, x_2, \cdots, x_n) \mid x_i \in A_i \text{ và } i = 1, 2, \cdots, n\}. \tag{1.14}$$

Phép nhân tập hợp không có tính giao hoán. Hàm thuộc của tập hợp tích $\mu_{A\times B}(x, y)$ có quan hệ với các hàm thuộc $\mu_A(x)$, $\mu_B(y)$ của hai tập thừa số A và B như sau:

$$\mu_{A\times B}(x, y) = \mu_A(x)\mu_B(y). \tag{1.15}$$

Thật vậy phần tử (x, y) chỉ thuộc $A\times B$, tức là $\mu_{A\times B}(x,y)=1$, khi và chỉ khi $x\in A$ và $y\in B$, nói cách khác khi cùng xảy ra đồng thời $\mu_A(x)=1$ và $\mu_B(y)=1$.

Câu hỏi ôn tập và bài tập

1) Sử dụng khái niệm hàm thuộc $\mu(x)$ để chứng minh các công thức sau

   a) $A\cap B = A \setminus (A\setminus B)$ b) $(A\setminus B)\cup C = (A\cup C)\setminus (B\setminus C)$

   c) $(A\setminus B)\cap C = (A\cap C)\setminus B$

2) Cho hai tập hợp A, B. Hiệu đối xứng $A\Delta B$ được hiểu là $A\Delta B = (A\setminus B)\cup(B\setminus A)$. Ký hiệu $\mu_A(x)$, $\mu_B(x)$, $\mu_{A\Delta B}(x)$, là các hàm thuộc của tập A, B, $A\Delta B$. Hãy chứng minh

   a) $B\setminus A = (A\Delta B)\cap B$ b) $\mu_{A\Delta B}(x) = \mu_A(x) + \mu_B(x) - 2\mu_A(x)\mu_B(x)$

   c) $A\cup B = A\Delta(B\setminus A)$

3) Cho n tập hợp $A_1, A_2, \cdots, A_n$. Hãy chứng minh hai công thức De Morgan

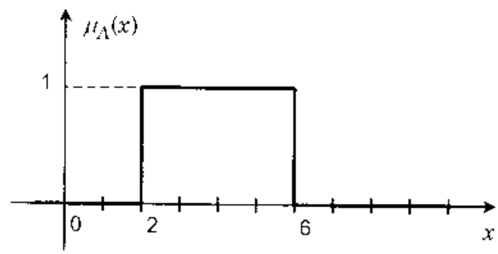
   a) $\left(\bigcup_{i=1}^{n} A_i\right)^c = \bigcap_{i=1}^{n} A_i^c$ b) $\left(\bigcap_{i=1}^{n} A_i\right)^c = \bigcup_{i=1}^{n} A_i^c$

1.2.2 Định nghĩa tập mờ

Hàm thuộc $\mu_A(x)$ định nghĩa trên tập A, trong khái niệm tập hợp kinh điển chỉ có hai giá trị là 1 nếu $x \in A$ hoặc 0 nếu $x \notin A$. *Hình 1.3* mô tả hàm thuộc của hàm $\mu_A(x)$, trong đó tập A được định nghĩa như sau:

$$A = \{x \in \mathbb{R} \mid 2 < x < 6\}. \tag{1.16}$$

Như vậy, trong lý thuyết tập hợp kinh điển, hàm thuộc hoàn toàn tương đương với định nghĩa một tập hợp. Từ định nghĩa về một tập hợp A bất kỳ ta có thể xác định được hàm thuộc $\mu_A(x)$ cho tập đó và ngược lại từ hàm thuộc $\mu_A(x)$ của tập A cũng hoàn toàn suy ra được định nghĩa cho A.



Hình 1.3: Hàm thuộc $\mu_A(x)$ của tập kinh điển A.

Cách biểu diễn hàm phụ thuộc như vậy sẽ không phù hợp với những tập được mô tả "*mờ*" như *tập B gồm các số thực dương nhỏ hơn nhiều so với 6*

$$B = \{x \in \mathbb{R} \mid x << 6\}, \tag{1.17}$$

có tập nền là $\mathbb{R}$, hoặc *tập C gồm các số thực gần bằng 3* cũng có tập nền $\mathbb{R}$

$$C = \{x \in \mathbb{R} \mid x \approx 3\}. \tag{1.18}$$

Lý do là với những định nghĩa "*mờ*" như vậy chưa đủ để xác định được một số chẳng hạn như x=3,5 có thuộc B hoặc x=2,5 có thuộc C hay không.

Nếu đã không khẳng định được x=3,5 có thuộc B hay không thì cũng không khẳng định được là số thực x=3,5 không thuộc B. Vậy thì x=3,5 thuộc B bao nhiêu phần trăm?. Giả sử rằng có câu trả lời thì lúc này hàm thuộc $\mu_B(x)$ tại điểm x=3,5 phải có một giá trị trong khoảng [0,1], tức là

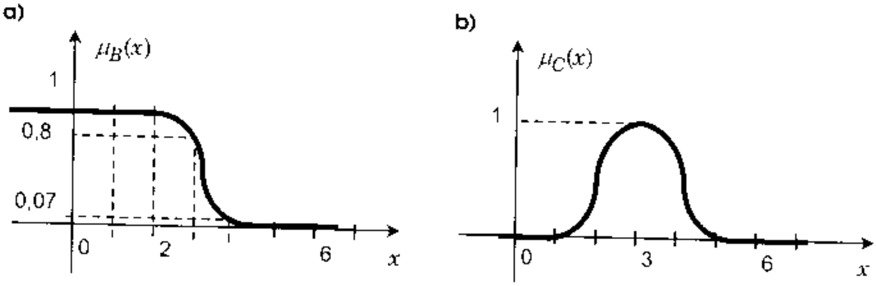
$$0 \le \mu_B(x) \le 1. \tag{1.19}$$

Nói cách khác hàm $\mu_B(x)$ không còn là hàm hai giá trị như đối với tập kinh điển nữa mà là một ánh xạ (*hình 1.4*)

$$\mu_B : X \to [0,1], \tag{1.20}$$

trong đó X là tập nền của tập "mờ".

Như vậy, khác với tập kinh điển A, từ "định nghĩa kinh điển" của tập "mờ" B hoặc C không suy ra được hàm phụ thuộc $\mu_B(x)$ hoặc $\mu_C(x)$ của chúng. Hơn thế nữa hàm phụ thuộc ở đây lại giữ một vai trò "*làm rõ định nghĩa*" cho một tập "mờ" như ví dụ ở *hình 1.4*. Do đó nó phải được nêu lên như là một điều kiện trong định nghĩa về tập "mờ".



Hình 1.4: a) Hàm phụ thuộc của tập "mờ" B.
b) Hàm phụ thuộc của tập "mờ" C.

Định nghĩa 1.1

Tập *mờ* F xác định trên tập kinh điển X là một tập mà mỗi phần tử của nó là một cặp các giá trị $(x, \mu_F(x))$ trong đó $x \in X$ và μ_F là ánh xạ

$$\mu_F : X \to [0,1]. \tag{1.21}$$

Ánh xạ μ_F được gọi là *hàm thuộc* (hoặc hàm phụ thuộc) của tập mờ F. Tập kinh điển X được gọi là *tập nền* (hay vũ trụ) của tập mờ F.

Ví dụ một tập mờ F của các số tự nhiên nhỏ hơn 6 với hàm phụ thuộc $\mu_F(x)$ có dạng như ở *hình 1.4a)* định nghĩa trên nền X sẽ chứa các phần tử sau

$$F = \{ (1, 1) , (2, 1) , (3, 0{,}8) , (4, 0{,}07) \}.$$

Số tự nhiên 1 và 2 có độ phụ thuộc

$\mu_F(1) = \mu_F(2) = 1,$

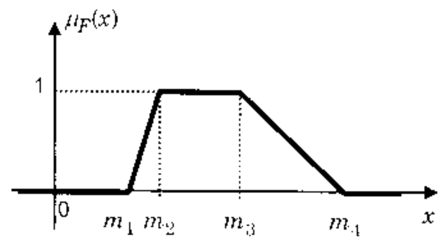
các số tự nhiên 3 và 4 có độ phụ thuộc nhỏ hơn 1

$\mu_F(3) = 0{,}8$ và $\mu_F(4) = 0{,}07.$

Những số không được liệt kê đều có độ phụ thuộc bằng 0.

Sử dụng các hàm thuộc để tính độ phụ thuộc của một phần tử x nào đó có hai cách:

- *tính trực tiếp* (nếu $\mu_F(x)$ cho trước dưới dạng công thức tường minh) hoặc
- *tra bảng* (nếu $\mu_F(x)$ cho dưới dạng bảng).



Hình 1.5: Hàm thuộc $\mu_F(x)$ có mức chuyển đổi tuyến tính.

Các hàm thuộc $\mu_F(x)$ có dạng "trơn" như ở *hình 1.4* được gọi là *hàm thuộc kiểu S*. Đối với hàm thuộc kiểu S, do các công thức biểu diễn $\mu_F(x)$ có độ phức tạp lớn, nên thời gian tính độ phụ thuộc cho một phần tử lâu. Bởi vậy trong kỹ thuật điều khiển mờ thông thường các hàm thuộc kiểu S hay được thay gần đúng bằng một hàm tuyến tính từng đoạn.

Một hàm thuộc có dạng tuyến tính từng đoạn được gọi là *hàm thuộc có mức chuyển đổi tuyến tính* (*hình 1.5*). Hàm thuộc $\mu_F(x)$ như ở *hình 1.5* với m_1=m_2 và m_3=m_4 chính là hàm thuộc của một tập kinh điển.

1.2.3 Độ cao, miền xác định và miền tin cậy của tập mờ

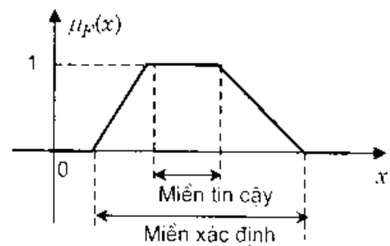
Trong những ví dụ trên các hàm thuộc đều có độ cao bằng 1. Điều đó nói rằng các tập mờ đó đều có ít nhất một phần tử với độ phụ thuộc bằng 1. Trong thực tế không phải tập mờ nào cũng có phần tử có độ phụ thuộc bằng 1, tương ứng với điều đó thì không phải mọi hàm thuộc đều có độ cao là 1.

Định nghĩa 1.2

Độ cao của một tập mờ F (định nghĩa trên tập nền X) là giá trị

$$h = \sup_{x \in X} \mu_F(x). \tag{1.22}$$

Ký hiệu $\sup_{x \in X} \mu(x)$ chỉ giá trị nhỏ nhất trong tất cả các giá trị chặn trên của hàm $\mu(x)$. Một tập mờ với ít nhất một phần tử có độ phụ thuộc bằng 1 được gọi là *tập mờ chính tắc* tức là h=1, ngược lại một tập mờ F với $h < 1$ được gọi là tập mờ *không chính tắc*.



Hình 1.6: Minh họa về miền xác định và miền tin cậy của một tập mờ.

Bên cạnh khái niệm về độ cao, mỗi tập mờ F còn có hai khái niệm quan trọng khác là

– *miền xác định* và

– *miền tin cậy*.

Định nghĩa 1.3

Miền xác định của tập mờ F (định nghĩa trên nền X), được ký hiệu bởi S là tập con của M thỏa mãn

$$S = \operatorname{supp} \mu_F(x) = \{\, x \in X \mid \mu_F(x) > 0 \,\}. \tag{1.23}$$

Ký hiệu supp$\mu_F(x)$ viết tắt của từ tiếng anh support, như công thức (1.23) đã chỉ rõ, là tập con trong X chứa các phần tử x mà tại đó hàm $\mu_F(x)$ có giá trị dương.

Định nghĩa 1.4

Miền tin cậy của tập mờ F (định nghĩa trên nền X), được ký hiệu bởi T, là tập con của M thoả mãn

$$T = \{\, x \in X \mid \mu_F(x) = 1 \,\}. \tag{1.24}$$

1.3 Các phép toán trên tập mờ

Những phép toán cơ bản trên tập mờ là *phép hợp*, *phép giao* và *phép bù*. Giống như định nghĩa về tập mờ, các phép toán trên tập mờ cũng sẽ được định nghĩa thông qua các hàm thuộc, được xây dựng tương tự như các hàm thuộc của các phép giao, hợp, bù giữa hai tập hợp kinh điển. Nói cách khác, khái niệm xây dựng những phép toán trên tập mờ được hiểu là việc xác định các hàm thuộc cho phép hợp (tuyển) $A \cup B$, giao $A \cap B$, bù (phủ định) A^C ... từ những tập mờ A, B.

Một nguyên tắc cơ bản trong việc xây dựng các phép toán trên tập mờ là không được mâu thuẫn với những phép toán đã có trong lý thuyết tập hợp kinh điển. Mặc dù không giống tập hợp kinh điển, hàm thuộc của các tập mờ $A \cup B$, giao $A \cap B$, bù A^C... được định nghĩa cùng với tập mờ, song sẽ không mâu thuẫn với các phép toán tương tự của tập hợp kinh điển nếu như chúng thỏa mãn những tính chất tổng quát được phát biểu như "tiên đề" của lý thuyết tập hợp kinh điển. Đó là các "tiên đề" (1.6) cho phép giao $A \cap B$, (1.9) cho phép hợp và (1.12) cho phép bù.

1.3.1 Phép hợp hai tập mờ

Các công thức (1.9) cho thấy một cách tổng quan những tính chất cơ bản của hàm thuộc $\mu_{A \cup B}(x)$ của hợp hai tập hợp kinh điển A, B.

Do trong định nghĩa về tập mờ hàm thuộc giữ vai trò như một thành phần cấu thành tập mờ nên các tính chất (1.9) sẽ không là điều hiển nhiên nữa. Thay vào đó chúng được sử dụng như những tiên đề để xây dựng phép hợp trên tập mờ.

Định nghĩa 1.5

Hợp của hai tập mờ A và B có cùng tập nền X là một tập mờ $A \cup B$ cũng xác định trên nền X có hàm thuộc $\mu_{A \cup B}(x)$ thỏa mãn:

a) $\mu_{A \cup B}(x)$ chỉ phụ thuộc vào $\mu_A(x)$ và $\mu_B(x)$.

b) $\mu_B(x) = 0$ với mọi $x \quad \Rightarrow \quad \mu_{A \cup B}(x) = \mu_A(x)$.

c) $\mu_{A \cup B}(x) = \mu_{B \cup A}(x)$, tức là có tính giao hoán.

d) Có tính kết hợp, tức là $\mu_{(A \cup B) \cup C}(x) = \mu_{A \cup (B \cup C)}(x)$.

e) Nếu $A_1 \subseteq A_2$ thì $A_1 \cup B \subseteq A_2 \cup B$, hay $\mu_{A \cup B}(x)$ có tính không giảm

$$\mu_{A_1}(x) \le \mu_{A_2}(x) \quad \Rightarrow \quad \mu_{A_1 \cup B}(x) \le \mu_{A_2 \cup B}(x).$$

Có thể dễ thấy được sẽ có nhiều công thức khác nhau được dùng để tính hàm thuộc $\mu_{A\cup B}(x)$ cho hợp hai tập mờ. Chẳng hạn 5 công thức sau đều có thể được sử dụng để định nghĩa hàm thuộc $\mu_{A\cup B}(x)$ của phép hợp giữa hai tập mờ:

1) $\mu_{A\cup B}(x) = \max\{\mu_A(x)\ ,\ \mu_B(x)\ \}$ *(Luật lấy max)* (1.25)

2) $$\mu_{A\cup B}(x) = \begin{cases} \max\{\mu_A(x),\mu_B(x)\} & \text{khi} \quad \min\{\mu_A(x),\mu_B(x)\} = 0 \\ 1 \quad \text{khi} \quad \min\{\mu_A(x),\mu_B(x)\} \neq 0 & \end{cases} \tag{1.26}$$

3) $\mu_{A\cup B}(x) = \min\{1,\ \mu_A(x)+\mu_B(x)\}$ *(Phép hợp Lukasiewicz),* (1.27)

4) $$\mu_{A\cup B}(x) = \frac{\mu_A(x)+\mu_B(x)}{1+\mu_A(x)+\mu_B(x)} \quad \textit{(Tổng Einstein)} \tag{1.28}$$

5) $\mu_{A\cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x)\mu_B(x)$ *(Tổng trực tiếp)* ... (1.29)

Ta sẽ chứng minh tính đúng đắn của khẳng định trên cho (1.25) làm một ví dụ, tức là phải chỉ ra rằng

$$\mu_{A\cup B}(x) = \max\{\mu_A(x)\ ,\ \mu_B(x)\}$$

thỏa mãn 5 tính chất đã nêu trong định nghĩa 1.5.

- Hiển nhiên là a) được thỏa mãn vì trong (1.25) chỉ chứa $\mu_A(x)$ và $\mu_B(x)$.
- Nếu $\mu_B(x) \equiv 0$ thì do

  $$\mu_{A\cup B}(x) = \max\{\mu_A(x)\ ,\ \mu_B(x)\} = \max\{\mu_A(x)\ ,\ 0\} \qquad \text{và} \quad \mu_A(x) \geq 0$$

  nên $\mu_{A\cup B}(x) = \max\{\mu_A(x)\ ,\ 0\} = \mu_A(x)$,

  tức là (1.25) thỏa mãn b).
- Vì

  $$\max\{\mu_A(x)\ ,\ \mu_B(x)\ \} = \max\{\mu_B(x)\ ,\ \mu_A(x)\}$$

  nên (1.25) có tính giao hoán.
- Do có

  $$\begin{aligned}\mu_{(A\cup B)\cup C}(x) &= \max\{\ \max\{\mu_A(x)\ ,\ \mu_B(x)\}\ ,\ \mu_C(x)\ \} \\ &= \max\{\mu_A(x),\mu_B(x),\mu_C(x)\ \} = \max\{\mu_A(x),\max\{\mu_B(x),\mu_C(x)\ \}\}\end{aligned}$$

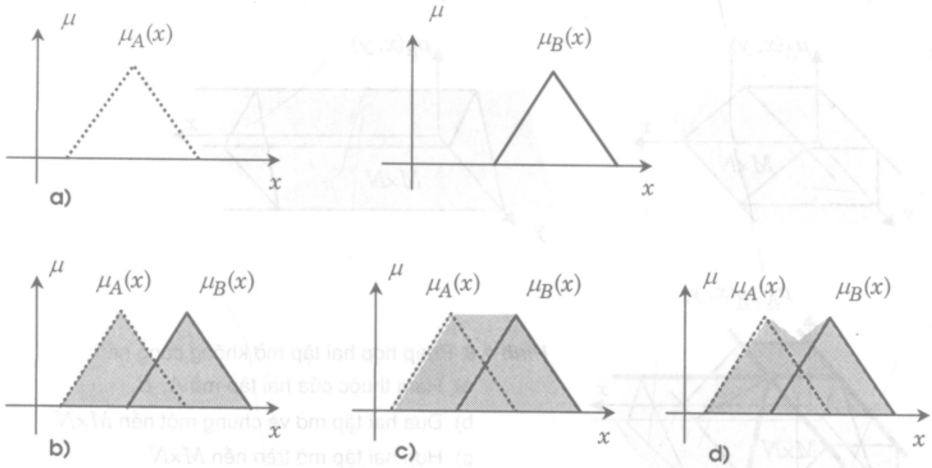
  nên (1.25) cũng có tính kết hợp, tức là thỏa mãn d).

– Với $\mu_{A_1}(x) \le \mu_{A_2}(x)$ ta được

$$\max\{\mu_{A_1}(x), \mu_B(x)\} \le \max\{\mu_{A_2}(x), \mu_B(x)\}$$

hay $\mu_{A_1 \cup B}(x) \le \mu_{A_2 \cup B}(x)$

và đó chính là điều phải chứng minh.



Hình 1.7: Hàm thuộc của hợp hai tập hợp có cùng không gian nền.

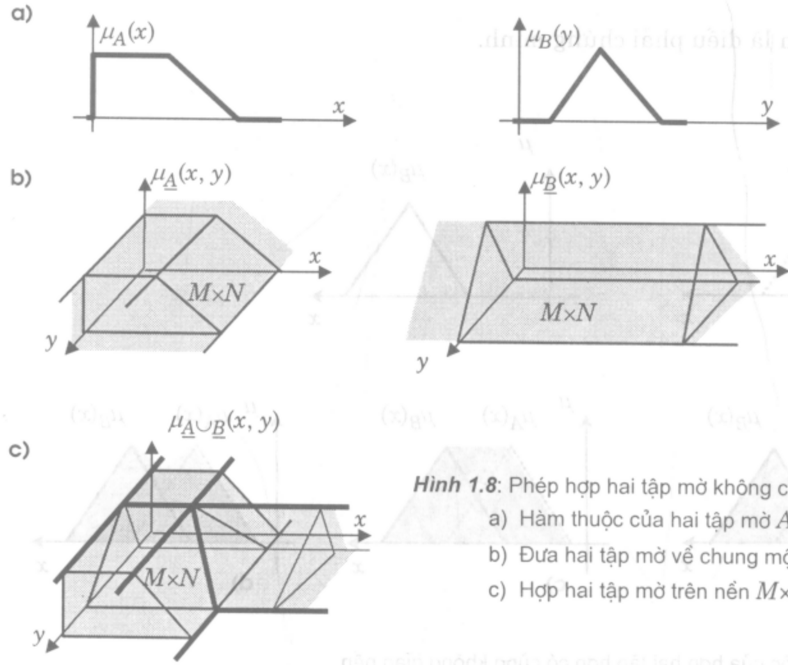
a) Hàm thuộc của hai tập mờ A và B.
b) Hợp hai tập mờ theo luật max.
c) Hợp hai tập mờ theo luật Lukasiewicz.
d) Hợp hai tập mờ theo luật tổng trực tiếp.

Một cách tổng quát thì bất cứ một ánh xạ dạng

$\mu_{A \cup B}(x): X \to [0,1]$

nếu thỏa mãn 5 tiêu chuẩn đã nêu trong định nghĩa 1.5 đều được xem như là hợp của hai tập mờ A và B có chung một tập nền X. Điều này nói rằng sẽ tồn tại rất nhiều cách xác định hợp của hai tập mờ và cho một bài toán điều khiển mờ có thể có nhiều lời giải khác nhau khi ta sử dụng các phép hợp hai tập mờ khác nhau. *Hình 1.7* là một ví dụ. Để tránh những mâu thuẩn xảy ra trong kết quả, nhất thiết trong một bài toán điều khiển ta chỉ nên thống nhất sử dụng một loại công thức cho phép hợp.

Các công thức (1.25) ÷ (1.29) cũng được mở rộng để áp dụng cho việc xác định hợp của hai tập mờ không cùng nền bằng cách đưa cả hai tập mờ về chung một tập nền là tích của hai tập nền đã cho.



Hình 1.8: Phép hợp hai tập mờ không cùng nền:

a) Hàm thuộc của hai tập mờ A, B.

b) Đưa hai tập mờ về chung một nền $M\times N$.

c) Hợp hai tập mờ trên nền $M\times N$.

Ví dụ có hai tập mờ A (định nghĩa trên tập nền M) và B (định nghĩa trên tập nền N). Ta sẽ xác định hợp $A\cup B$ của chúng theo luật max (1.25). Do hai tập nền M và N độc lập với nhau nên hàm thuộc $\mu_A(x)$, $x\in M$ của tập mờ A sẽ không phụ thuộc vào N và ngược lại $\mu_B(y)$, $y\in N$ của tập mờ B cũng sẽ không phụ thuộc vào M. Điều đó thể hiện ở chỗ trên tập nền mới là tập tích $M\times N$ hàm $\mu_A(x)$ phải là một mặt "cong" dọc theo trục y và $\mu_B(y)$ là một mặt "cong" dọc theo trục x (*hình 1.8*). Tập mờ A như vậy được định nghĩa trên hai tập nền M và $M\times N$. Để phân biệt được chúng, sau đây ký hiệu $\underline{A}$ sẽ được dùng để chỉ tập mờ A trên tập nền $M\times N$. Tương tự, ký hiệu $\underline{B}$ cũng sẽ được dùng để chỉ tập mờ B trên tập nền $M\times N$. Với ký hiệu đó thì

$$\mu_{\underline{A}}(x, y) = \mu_A(x) \qquad \text{với mọi } y\in N$$

và $$\mu_{\underline{B}}(x, y) = \mu_B(y) \qquad \text{với mọi } x\in M.$$

Sau khi đã đưa được hai tập mờ A, B về chung một tập nền là $M\times N$ thành $\underline{A}$ và $\underline{B}$ thì hàm thuộc $\mu_{\underline{A}\cup\underline{B}}(x, y)$ của tập mờ $\underline{A}\cup\underline{B}$ được xác định theo công thức (1.25).

Hợp hai tập mờ theo luật max

Hợp của hai tập mờ A với hàm thuộc $\mu_A(x)$ (định nghĩa trên tập nền M) và B với hàm thuộc $\mu_B(x)$ (định nghĩa trên tập nền N) theo luật max là một tập mờ xác định trên tập nền $M\times N$ với hàm thuộc

$$\mu_{\underline{A}\cup\underline{B}}(x, y) = \max\{\mu_{\underline{A}}(x, y)\ ,\ \mu_{\underline{B}}(x, y)\}, \tag{1.30a}$$

trong đó

$$\mu_{\underline{A}}(x, y) = \mu_A(x) \qquad \text{với mọi } y\in N$$

và $$\mu_{\underline{B}}(x, y) = \mu_B(y) \qquad \text{với mọi } x\in M.$$

Tương tự, ta cũng có định nghĩa hợp theo luật sum theo công thức (1.27) như sau:

Hợp hai tập mờ theo luật sum (Lukasiewicz)

Hợp của hai tập mờ A với hàm thuộc $\mu_A(x)$ (định nghĩa trên tập nền M) và B với hàm thuộc $\mu_B(x)$ (định nghĩa trên tập nền N) theo luật sum là một tập mờ xác định trên tập nền $M\times N$ với hàm thuộc

$$\mu_{\underline{A}\cup\underline{B}}(x, y) = \min\{1,\ \mu_{\underline{A}}(x, y)+\mu_{\underline{B}}(x, y)\} \tag{1.30b}$$

trong đó

$$\mu_{\underline{A}}(x, y) = \mu_A(x) \qquad \text{với mọi } y\in N$$

và $$\mu_{\underline{B}}(x, y) = \mu_B(y) \qquad \text{với mọi } x\in M.$$

Một cách tổng quát, do hàm thuộc $\mu_{\underline{A}\cup\underline{B}}(x, y)$ của hợp hai tập mờ A, B không cùng không gian nền chỉ phụ thuộc vào $\mu_A(x)\in[0,1]$ và $\mu_B(x)\in[0,1]$ nên ta có thể xem $\mu_{\underline{A}\cup\underline{B}}(x, y)$ là hàm của hai biến μ_A, μ_B được định nghĩa như sau

$$\mu_{\underline{A}\cup\underline{B}}(x, y) = \mu(\mu_A,\ \mu_B) : [0,1]^2 \to [0,1]. \tag{1.31}$$

Ta đi đến định nghĩa về hàm thuộc $\mu(\mu_A, \mu_B)$ của hợp hai tập mờ không cùng không gian nền:

Định nghĩa 1.6

Hàm thuộc của hợp giữa hai tập mờ A với $\mu_A(x)$ định nghĩa trên nền M và B với $\mu_B(y)$ định nghĩa trên tập nền N là một hàm hai biến $\mu(\mu_A, \mu_B) : [0,1]^2 \to [0,1]$ xác định trên nền $M \times N$ thỏa mãn:

a) $\mu_B = 0 \quad \Rightarrow \quad \mu(\mu_A, \mu_B) = \mu_A$.

b) $\mu(\mu_A, \mu_B) = \mu(\mu_B, \mu_A)$, tức là có tính giao hoán.

c) $\mu(\mu_A, \mu(\mu_B, \mu_C)) = \mu(\mu(\mu_A, \mu_B), \mu_C)$, tức là có tính kết hợp.

d) $\mu(\mu_A, \mu_B) \le \mu(\mu_C, \mu_D)$, $\forall \mu_A \le \mu_C$, $\mu_B \le \mu_D$, tức là có tính không giảm.

Một hàm hai biến $\mu(\mu_A, \mu_B)$: $[0,1]^2 \to [0,1]$ thỏa mãn các điều kiện của định nghĩa 1.6 còn được gọi là hàm *t-đối chuẩn* (*t-conorm*).

Mặc dù có nhiều cách xác định hàm thuộc của hợp hai tập mờ như vậy, song trong lý thuyết điều khiển mờ và nội dung quyển sách này sẽ chỉ tập trung chính vào hai phép hợp mờ là phép hợp max và phép hợp sum đã được phát biểu trong công thức (1.30a), (1.30b).

1.3.2 Phép giao hai tập mờ

Như đã đề cập, phép giao $A \cap B$ trên tập mờ phải được được định nghĩa sao cho không mâu thuẫn với phép giao của tập hợp kinh điển và yêu cầu này sẽ được thỏa mãn nếu chúng có được các tính chất tổng quát (1.6) của tập kinh điển.

Tương tự như đã làm với phép hợp hai tập mờ, phép giao hai tập mờ trên tập nền tổng quát hóa những tính chất (1.6) cũng chỉ được thực hiện trực tiếp nếu hai tập mờ đó có cùng tập nền. Trong trường hợp chúng không cùng một tập nền thì phải đưa chúng về chung một tập nền mới là tập tích của hai tập nền đã cho.

Định nghĩa 1.7

Giao của hai tập mờ A và B có cùng tập nền X là một tập mờ cũng xác định trên tập nền X với hàm thuộc thỏa mãn :

a) $\mu_{A \cap B}(x)$ chỉ phụ thuộc vào $\mu_A(x)$ và $\mu_B(x)$.

b) $\mu_B(x) = 1$ với mọi $x \quad \Rightarrow \quad \mu_{A \cap B}(x) = \mu_A(x)$.

c) $\mu_{A \cap B}(x) = \mu_{B \cap A}(x)$, tức là có tính giao hoán.

d) $\mu_{(A\cap B)\cap C}(x) = \mu_{A\cap(B\cap C)}(x)$, tức là có tính kết hợp.

e) $\mu_{A_1}(x) \le \mu_{A_2}(x) \Rightarrow \mu_{A_1\cap B}(x) \le \mu_{A_2\cap B}(x)$, tức là hàm không giảm.

Giống như đã trình bày về phép hợp hai tập mờ, có nhiều công thức khác nhau được dùng để tính hàm thuộc $\mu_{A\cap B}(x)$ của giao hai tập mờ và bất cứ một ánh xạ

$$\mu_{A\cap B}(x): X \to [0,1]$$

nào thỏa mãn 5 tiêu chuẩn đã nêu trong định nghĩa 1.7 trên đều được xem như là hàm thuộc của giao hai tập mờ A và B có chung nền X.

Các công thức thường dùng để tính hàm thuộc $\mu_{A\cap B}(x)$ của phép giao gồm:

1) $\mu_{A\cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}$ (1.32)

2) $\mu_{A\cap B}(x) = \begin{cases} \min\{\mu_A(x), \mu_A(x)\} & \text{nếu } \max\{\mu_A(x), \mu_A(x)\} = 1 \\ 0 \text{ nếu } \max\{\mu_A(x), \mu_A(x)\} \ne 1 \end{cases}$ (1.33)

3) $\mu_{A\cap B}(x) = \max\{0, \mu_A(x)+\mu_B(x)-1\}$ *(Phép giao Lukasiewicz)* (1.34)

4) $\mu_{A\cap B}(x) = \dfrac{\mu_A(x)\mu_B(x)}{2-\left(\mu_A(x)+\mu_B(x)\right)-\mu_A(x)\mu_B(x)}$ *(Tích Einstein)* (1.35)

5) $\mu_{A\cap B}(x) = \mu_A(x)\mu_B(x)$ *(Tích đại số)*. (1.36)

Tuy nhiên luật min (1.32) và tích đại số là hai loại luật xác định hàm thuộc của giao hai tập mờ được ưa dùng hơn cả trong kỹ thuật điều khiển mờ.

Cũng như đã làm với phép hợp hai tập mờ ta sẽ chứng minh tính đúng đắn của các công thức (1.32)÷(1.36) bằng cách chỉ ra rằng luật min (1.32)

$$\mu_{A\cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}$$

thỏa mãn 5 tính chất đã nêu trong định nghĩa 1.7.

– Hiển nhiên là a) được thỏa mãn vì trong (1.32) chỉ chứa $\mu_A(x)$ và $\mu_B(x)$.

– Nếu $\mu_B(x) \equiv 1$ thì do

$$\mu_{A\cap B}(x) = \min\{\mu_A(x), \mu_B(x)\} = \min\{\mu_A(x), 1\}$$

và $\mu_A(x) \le 1$

nên $\mu_{A\cap B}(x) = \min\{\mu_A(x), 1\} = \mu_A(x)$, tức là (1.32) thỏa mãn b).

- Vì

$$\min\{\mu_A(x)\,,\,\mu_B(x)\} = \min\{\mu_B(x)\,,\,\mu_A(x)\}$$

nên (1.32) có tính giao hoán.

- Do có

$$\mu_{(A\cap B)\cap C}(x) = \min\{\min\{\mu_A(x)\,,\,\mu_B(x)\}\,,\,\mu_C(x)\} = \min\{\mu_A(x),\mu_B(x),\mu_C(x)\}$$

$$= \min\{\mu_A(x),\, \min\{\mu_B(x),\mu_C(x)\}\} = \mu_{A\cap(B\cap C)}(x)$$

nên (1.32) cũng có tính kết hợp, tức là thỏa mãn d).

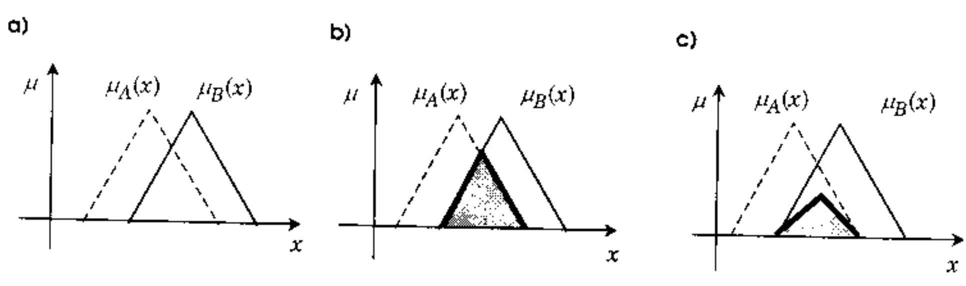
- Với $\mu_{A_1}(x) \le \mu_{A_2}(x)$ ta được $\min\{\mu_{A_1}(x),\mu_B(x)\} \le \min\{\mu_{A_2}(x),\mu_B(x)\}$

hay $\mu_{A_1\cap B}(x) \le \mu_{A_2\cap B}(x)$ và đó chính là điều phải chứng minh.

Việc có nhiều công thức xác định hàm thuộc của giao hai tập mờ đưa đến khả năng một bài toán điều khiển mờ có thể có nhiều lời giải khác nhau như *hình 1.9* dưới đây là một ví dụ.

Để tránh những mâu thuẫn trong kết quả có thể xảy ra, nhất thiết trong một bài toán điều khiển ta chỉ nên thống nhất sử dụng một loại phép giao.

Các công thức (1.32) ÷ (1.36) cũng áp dụng được cho hợp hai tập mờ không cùng tập nền bằng cách đưa cả hai tập mờ về chung một tập nền là tích của hai tập nền đã cho.



Hình 1.9: Hàm thuộc của giao hai tập hợp có cùng không gian nền.

a) Hàm thuộc của hai tập mờ *A* và *B*.

b) Giao hai tập mờ theo luật min.

c) Giao hai tập mờ theo luật tích đại số.

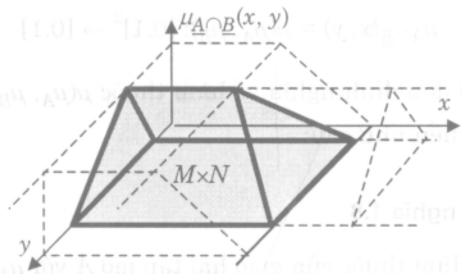
Chẳng hạn có hai tập mờ A định nghĩa trên tập nền M và B định nghĩa trên tập nền N (*hình 1.8a*). Do hai tập nền M và N độc lập với nhau nên hàm thuộc $\mu_A(x)$, $x \in M$ của tập mờ A sẽ không phụ thuộc vào N và ngược lại $\mu_B(y)$, $y \in N$ của tập mờ B cũng sẽ không phụ thuộc vào M. Trên tập nền mới là tập tích $M \times N$ hàm $\mu_A(x)$ phải là một mặt "cong" dọc theo trục y và $\mu_B(y)$ là một mặt "cong" dọc theo trục x. Tập mờ A (hoặc B) như vậy được định nghĩa trên hai tập nền M (hoặc N) và $M \times N$. Để phân biệt được chúng, ký hiệu $\underline{A}$ (hoặc $\underline{B}$) sẽ được dùng để chỉ tập mờ A (hoặc B) trên tập nền mới là $M \times N$.

Với những ký hiệu đó thì

$\mu_{\underline{A}}(x, y) = \mu_A(x)$ với mọi $y \in N$ và

$\mu_{\underline{B}}(x, y) = \mu_B(y)$ với mọi $x \in M$.

Công thức (1.32) xác định hàm thuộc của giao hai tập mờ cùng không gian nền bây giờ hoàn toàn được áp dụng với $\underline{A}$, $\underline{B}$.



Hình 1.10: Phép giao hai tập mờ không cùng nền.

Giao hai tập mờ theo luật min

Giao của tập mờ A có hàm thuộc $\mu_A(x)$ định nghĩa trên tập nền M với tập mờ B có hàm thuộc $\mu_B(x)$ định nghĩa trên tập nền N là một tập mờ xác định trên tập nền $M \times N$ có hàm thuộc

$$\mu_{\underline{A}\cap\underline{B}}(x, y) = \min\{\mu_A(x)\ ,\ \mu_B(y)\} = \min\{\mu_{\underline{A}}(x, y)\ ,\ \mu_{\underline{B}}(x, y)\}, \tag{1.37a}$$

trong đó

$\mu_{\underline{A}}(x, y) = \mu_A(x)\ \forall\ y \in N$ và $\mu_{\underline{B}}(x, y) = \mu_B(y)\ \forall\ x \in M$.

Với ví dụ về tập mờ A, B có hàm đặc tính như ở *hình 1.8a* thì tập giao của chúng trên tập nền chung $M \times N$ sẽ có hàm thuộc mô tả như ở *hình 1.10*.

Một cách hoàn toàn tương tự, nếu như áp dụng công thức tích đại số (1.36) để xác định tập giao của hai tập mờ không cùng nền ta được

Giao hai tập mờ theo luật tích đại số

Giao của tập mờ A có hàm thuộc $\mu_A(x)$ định nghĩa trên tập nền M với tập mờ B có hàm thuộc $\mu_B(x)$ định nghĩa trên tập nền N là một tập mờ xác định trên tập nền $M\times N$ có hàm thuộc

$$\mu_{\underline{A}\cap\underline{B}}(x, y) = \mu_{\underline{A}}(x, y)\cdot\mu_{\underline{B}}(x, y), \tag{1.37b}$$

trong đó

$$\mu_{\underline{A}}(x, y) = \mu_A(x)\ \forall\ y\in N \qquad \text{và} \qquad \mu_{\underline{B}}(x, y) = \mu_B(y)\ \forall\ x\in M.$$

Trong hai ví dụ trên ta thấy hàm thuộc $\mu_{\underline{A}\cap\underline{B}}(x, y)$ của hợp hai tập mờ A, B không cùng nền chỉ phụ thuộc vào giá trị các hàm $\mu_A(x)\in[0,1]$ và $\mu_B(x)\in[0,1]$. Do đó không mất tính tổng quát nếu ta xem $\mu_{\underline{A}\cup\underline{B}}(x, y)$ như một hàm của hai biến μ_A và μ_B:

$$\mu_{\underline{A}\cap\underline{B}}(x, y) = \mu(\mu_A, \mu_B)\colon [0,1]^2 \to [0,1] \tag{1.38}$$

và đi đến định nghĩa về hàm thuộc $\mu(\mu_A, \mu_B)$ của giao hai tập mờ không cùng không gian nền như sau:

Định nghĩa 1.8

Hàm thuộc của giao hai tập mờ A với $\mu_A(x)$ định nghĩa trên nền M và B với $\mu_B(y)$ định nghĩa trên tập nền N là một hàm hai biến $\mu(\mu_A, \mu_B)\colon [0,1]^2 \to [0,1]$ xác định trên nền $M\times N$ thỏa mãn:

a) $\mu_B = 1 \quad \Rightarrow \quad \mu(\mu_A, \mu_B) = \mu_A$.

b) $\mu(\mu_A, \mu_B) = \mu(\mu_B, \mu_A)$, tức là có tính giao hoán.

c) $\mu(\mu_A, \mu(\mu_B, \mu_C)) = \mu(\mu(\mu_A, \mu_B), \mu_C)$, tức là có tính kết hợp.

d) $\mu(\mu_A, \mu_B) \le \mu(\mu_C, \mu_D)$, $\forall\, \mu_A \le \mu_C$, $\mu_B \le \mu_D$, tức là có tính không giảm.

Một hàm hai biến $\mu(\mu_A, \mu_B)\colon [0,1]^2 \to [0,1]$ thỏa mãn các điều kiện trên được gọi là hàm *t- chuẩn* (*t-norm*).

1.3.3 Phép bù của một tập mờ

Phép bù (còn gọi là phủ định) của một tập mờ được suy ra từ các tính chất (1.12) của phép bù trong lý thuyết tập hợp kinh điển như sau:

Định nghĩa 1.9

Tập bù của tập mờ A định nghĩa trên nền X là một tập mờ A^C cũng xác định trên tập nền X với hàm thuộc thỏa mãn

a) $\mu_{A^C}(x)$ chỉ phụ thuộc vào $\mu_A(x)$.

b) Nếu $x \in A$ thì $x \notin A^C$, hay $\mu_A(x)=1 \Rightarrow \mu_{A^C}(x)=0$.

c) Nếu $x \notin A$ thì $x \in A^C$, hay $\mu_A(x)=0 \Rightarrow \mu_{A^C}(x)=1$.

d) Nếu $A \subseteq B$ thì $A^C \supseteq B^C$, hay $\mu_A(x) \leq \mu_B(x) \Rightarrow \mu_{A^C}(x) \geq \mu_{B^C}(x)$.

Do hàm thuộc $\mu_{A^C}(x)$ của A^C chỉ phụ thuộc vào $\mu_A(x)$ nên ta có thể xem $\mu_{A^C}(x)$ như là một hàm của $\mu_A \in [0,1]$. Từ đó có định nghĩa tổng quát hơn về phép bù mờ như sau:

Định nghĩa 1.10

Tập bù của tập mờ A định nghĩa trên nền X là một tập mờ A^C cũng xác định trên tập nền X với hàm thuộc

$$\mu(\mu_A): [0,1] \rightarrow [0,1]$$

thỏa mãn

a) $\mu(1)=0$ và $\mu(0)=1$.

b) $\mu_A \leq \mu_B \Rightarrow \mu(\mu_A) \geq \mu(\mu_B)$, tức là hàm không tăng.

Nếu hàm một biến $\mu(\mu_A)$ còn liên tục và

$$\mu_A < \mu_B \Rightarrow \mu(\mu_A) > \mu(\mu_B),$$

thì phép bù mờ trên còn được gọi là *phép bù mờ chặt* (*strictly*).

Một phép bù mờ chặt sẽ là *phép bù mờ mạnh* (*strongly*), nếu

$$\mu(\mu(\mu_A)) = \mu_A, \text{ tức là } \left(A^C\right)^C = A.$$

Hàm thuộc $\mu(\mu_A)$ của một phép bù mờ mạnh được gọi là *hàm phủ định mạnh*.

Phép bù mờ mạnh

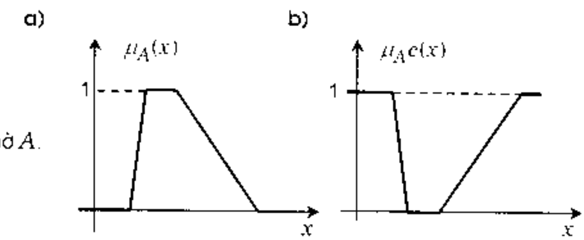
Phép bù mờ của một tập mờ A hay dùng trong điều khiển mờ là phép bù có tập mờ A^C với hàm thuộc

$$\mu_{A^C}(x) = 1 - \mu_A(x). \tag{1.39}$$

Nếu $\mu_A(x)$ là một hàm liên tục thì hàm thuộc (1.39) của tập bù A^C là một hàm phủ định mạnh. Thật vậy,

- Do $\mu_A(x)$ liên tục nên $\mu_{A^C}(x)$ cũng là một hàm liên tục.
- Nếu $\mu_{A_1}(x) < \mu_{A_2}(x)$ thì hiển nhiên có $\mu_{A_1^C}(x) > \mu_{A_2^C}(x)$.
- $\mu_{(A^C)^C}(x) = 1 - \mu_{A^C}(x) = 1 - \big(1 - \mu_A(x)\big) = \mu_A(x)$

Hình 1.11 là một ví dụ minh họa về hàm lên thuộc của phép phủ định mạnh (1.39).



Hình 1.11: Tập bù mạnh A^C của tập mờ A.
a) Hàm thuộc của tập mờ A;
b) Hàm thuộc của tập mờ A^C.

Tính đối ngẫu

Cho hai tập mờ A (trên không gian nền M) và B (trên không gian nền N) với các hàm thuộc tương ứng là $\mu_A(x)$, $\mu_B(x)$. Gọi $A \cup B$ là tập mờ hợp của chúng. Theo định nghĩa 1.6 tập mờ $A \cup B$ sẽ có hàm thuộc $\mu_{A\cup B}(\mu_A, \mu_B)$ thỏa mãn

$$\mu_{A\cup B} : [0,1]^2 \to [0,1]$$

là một hàm *t-đối chuẩn*. Sử dụng hàm phủ định

$$\eta(\xi) = 1 - \xi$$

ta sẽ có

$$\eta(\mu_{A\cup B}) = 1 - \mu_{A\cup B}(\eta(\mu_A),\ \eta(\mu_B)) = 1-(1-\mu_A, 1-\mu_B) \tag{1.40}$$

là một hàm *t-chuẩn*. Phần chứng minh dành cho bài tập.

Tính đối ngẫu giữa t-chuẩn và t-đối chuẩn cho phép xây dựng được một phép giao mờ từ một phép hợp mờ tương ứng.

Câu hỏi ôn tập và bài tập

1) Chứng minh rằng nếu $\mu_{A\cup B}$ là hàm thuộc của hợp hai tập mờ A với $\mu_A(x)$ định nghĩa trên tập nền M và B với $\mu_A(x)$ định nghĩa trên tập nền N, thì

$$\mu_{A\cap B} = \eta(\mu_{A\cup B}) = 1 - (1-\mu_A, 1-\mu_B)$$

xác định theo công thức (1.40) sẽ là hàm thuộc của giao hai tập mờ A và B.

2) Thiết lập hàm thuộc phân loại con người theo người thấp người tầm thước và người cao.

3) Thiết lập mô hình phân loại sinh viên qua các tập mờ sinh viên cần cù, sinh viên thông minh và sinh viên lười.

4) Cho các tập mờ A, B, C được định nghĩa trên nền $X= [0,10]$ với các hàm thuộc sau

$$\mu_A(x)= \frac{x}{x+2}, \quad \mu_B(x) = 2-x, \quad \mu_C(x) = \frac{1}{1+10(x-2)^2}$$

Hãy xác định hàm thuộc của các tập mờ sau dưới dạng biểu thức và đồ thị:

a) A^c b) B^c c) C^c d) $A\cup B$ e) $B\cup C$ f) $A\cup C$

g) $A\cap B$ h) $B\cap C$ i) $A\cap C$ j) $A\cap B\cap C$ k) $A\cup B\cup C$

l) $A\cap B^c$ m) $(A\cap B)^c$ n) $(A\cap C)^c$

5) Cho A là một tập mờ xác định trên nền X. Hãy chỉ ra rằng biểu thức $A\cap A^c=X$ không đúng như đối với tập hợp kinh điển.

6) Chứng minh rằng tập mờ A, B với các hàm thuộc

$$\mu_A(x) = \frac{x}{x+2}, \quad \mu_B(x) = \sqrt{\frac{x}{x+2x}}$$

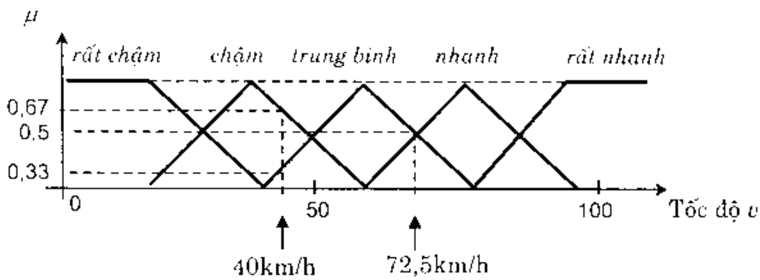
thoả mãn hai công thức De Morgan.

1.4 Biến ngôn ngữ và giá trị của nó

Quay lại với ví dụ về lái ô tô. Trong ví dụ đó đại lượng tốc độ có những *giá trị* được nhắc đến dưới dạng ngôn ngữ như:

- *rất chậm*,
- *chậm*,
- *trung bình*,
- *nhanh* và
- *rất nhanh*.

Mỗi *giá trị ngôn ngữ* đó của biến tốc độ được xác định bằng một tập mờ định nghĩa trên tập nền là tập các số thực dương chỉ giá trị vật lý x (đơn vị là km/h) của biến *tốc độ v* như 40km/h, 50km/h, ... (*hình 1.12*).



Hình 1.12: Mô tả các giá trị ngôn ngữ bằng tập mờ.

Hàm thuộc tương ứng của chúng được ký hiệu bằng

$$\mu_{rất\ chậm}(x),\ \mu_{chậm}(x),\ \mu_{trung\ bình}(x),\ \mu_{nhanh}(x) \text{ và } \mu_{rất\ nhanh}(x).$$

Như vậy, biến tốc độ v có hai miền giá trị khác nhau:

- miền các giá trị ngôn ngữ

  N = {*rất chậm*, *chậm*, *trung bình*, *nhanh*, *rất nhanh*},

- miền các giá trị vật lý (miền các giá trị rõ)

  $V = \{ x \in \mathbb{R} \mid x \geq 0 \}$,

và mỗi giá trị ngôn ngữ (mỗi phần tử của N) lại được mô tả bằng một tập mờ có tập nền là miền các giá trị vật lý V.

Biến tốc độ v, xác định trên miền các giá trị ngôn ngữ N, được gọi là *biến ngôn ngữ*. Do tập nền các tập mờ mô tả giá trị ngôn ngữ của *biến ngôn ngữ tốc độ* lại chính là tập V các giá trị vật lý của biến nên từ một giá trị vật lý $x \in V$ có được một vector μ gồm các độ phụ thuộc của x như sau:

$$x \mapsto \mu = \begin{pmatrix} \mu_{rất\ chậm}(x) \\ \mu_{chậm}(x) \\ \mu_{trung\ bình}(x) \\ \mu_{nhanh}(x) \\ \mu_{rất\ nhanh}(x) \end{pmatrix}. \tag{1.41}$$

Ánh xạ (1.41) có tên gọi là *quá trình Fuzzy hóa* (hay *mờ hóa*) của giá trị rõ x. Ví dụ, kết quả *Fuzzy hóa* giá trị vật lý x = 40km/h (giá trị rõ) của biến tốc độ sẽ là:

$$40\text{km/h} \mapsto \begin{pmatrix} 0 \\ 0{,}67 \\ 0{,}33 \\ 0 \\ 0 \end{pmatrix} \text{ (\textit{hình 1.12})},$$

hoặc của x =72,5km/h là

$$72{,}5\text{km/h} \mapsto \begin{pmatrix} 0 \\ 0 \\ 0{,}5 \\ 0{,}5 \\ 0 \end{pmatrix}.$$

1.5 Luật hợp thành mờ

1.5.1 Mệnh đề hợp thành

Trên đây, biến ngôn ngữ (ví dụ biến v chỉ tốc độ xe) được xác định thông qua tập các giá trị mờ của nó. Cùng là một đại lượng vật lý chỉ tốc độ nhưng biến v có hai dạng thể hiện

- là biến vật lý với các giá trị rõ như v = 40km/h hay v = 72,5km/h, ··· (miền xác định là tập kinh điển),
- là biến ngôn ngữ với các giá trị mờ như *rất chậm*, *chậm*, *trung bình*, ··· (miền xác định là tập các tập mờ).

Để phân biệt chúng, sau đây ký hiệu la mã sẽ được dùng để chỉ biến ngôn ngữ thay vì ký hiệu thường. Chẳng hạn biến ngôn ngữ χ sẽ có nhiều giá trị ngôn ngữ khác nhau là các tập mờ với hàm thuộc $\mu_{A_1}(x)$, $\mu_{A_2}(x)$, $\mu_{A_3}(x)$, ...

Cho hai biến ngôn ngữ χ và γ. Nếu biến χ nhận giá trị (mờ) A với hàm thuộc $\mu_A(x)$ và γ nhận giá trị (mờ) B có hàm thuộc $\mu_B(y)$ thì biểu thức

$$\chi = A, \tag{1.42a}$$

được gọi là *mệnh đề điều kiện* và

$$\gamma = B, \tag{1.42b}$$

là *mệnh đề kết luận*.

Ký hiệu mệnh đề (1.42a) là p và (1.42b) là q thì *mệnh đề hợp thành*

$p \Rightarrow q$ (từ p suy ra q),

hoàn toàn tương ứng với luật điều khiển (*mệnh đề hợp thành một điều kiện*)

$$\text{NẾU } \chi = A \text{ THÌ } \gamma = B. \tag{1.42c}$$

Mệnh đề hợp thành trên là một ví dụ đơn giản về bộ điều khiển mờ. Nó cho phép từ một giá trị đầu vào x_0 hay cụ thể hơn là từ độ phụ thuộc $\mu_A(x_0)$ đối với tập mờ A của giá trị đầu vào x_0 xác định được hệ số thỏa mãn mệnh đề kết luận q của giá trị đầu ra y. Hệ số thỏa mãn mệnh đề kết luận này được gọi là giá trị của mệnh đề hợp thành khi đầu vào bằng A và giá trị của mệnh đề hợp thành (1.42c)

$A \Rightarrow B$ (từ A suy ra B),

là một giá trị mờ. Biểu diễn giá trị mờ đó là tập hợp C thì mệnh đề hợp thành mờ (1.42c) chính là ánh xạ

$$\mu_A(x_0) \mapsto \mu_C(y).$$

1.5.2 Mô tả mệnh đề hợp thành mờ

Ánh xạ $\mu_A(x_0) \mapsto \mu_C(y)$ chỉ ra rằng mệnh đề hợp thành là một tập mà mỗi phần tử là một giá trị $(\mu_A(x_0), \mu_C(y))$, tức là mỗi phần tử là một tập mờ. Mô tả mệnh đề hợp thành tức là mô tả ánh xạ trên.

Quay lại mệnh đề logic kinh điển, giữa mệnh đề hợp thành $p \Rightarrow q$ và các mệnh đề điều kiện p, kết luận q có quan hệ sau:

p	q	$p \Rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

nói cách khác mệnh đề hợp thành $p \Rightarrow q$ sẽ có giá trị logic của $\sim p \vee q$, trong đó ~ chỉ phép phủ định và $\vee$ chỉ phép tính logic HOẶC.

Như vậy mệnh đề hợp thành kinh điển $p \Rightarrow q$ là một biểu thức logic có giá trị $R_{p \Rightarrow q}$ thỏa mãn:

a) $p=0 \quad \Rightarrow \quad R_{p \Rightarrow q} = 1.$

b) $q=1 \quad \Rightarrow \quad R_{p \Rightarrow q} = 1.$

c) $p=1$ và $q=0 \quad \Rightarrow \quad R_{p \Rightarrow q} = 0.$

So sánh các tính chất a) và c) ta rút ra được

d) $p_1 \le p_2 \quad \Rightarrow \quad R_{p_1 \Rightarrow q} \ge R_{p_2 \Rightarrow q}$.

Tương tự như vậy, từ b) và c) ta có

e) $q_1 \le q_2 \quad \Rightarrow \quad R_{p \Rightarrow q_1} \le R_{p \Rightarrow q_2}$.

Năm tính chất trên tạo thành bộ "tiên đề" cho việc xác định giá trị logic của mệnh đề hợp thành kinh điển. Bây giờ ta xét đến mệnh đề hợp thành mờ, tức là mệnh đề hợp thành có cấu trúc

NẾU $\chi = A$ THÌ $\gamma = B$. (1.43a)

hay

$\mu_A(x) \Rightarrow \mu_B(y)$, với $\mu_A, \mu_B \in [0, 1]$, (1.43b)

trong đó $\mu_A(x)$ là hàm thuộc của tập mờ đầu vào A định nghĩa trên tập nền X và $\mu_B(y)$ là hàm thuộc của B trên tập nền Y.

Định nghĩa 1.11 (Suy diễn đơn thuần)

Giá trị của mệnh đề hợp thành mờ (1.43) là một tập mờ định nghĩa trên nền Y (không gian nền của B) và có hàm thuộc

$\mu_{A \Rightarrow B}(y) : Y \rightarrow [0,1]$

thỏa mãn

a) $\mu_{A\to B}(y)$ chỉ phụ thuộc vào $\mu_A(x)$ và $\mu_B(y)$.

b) $\mu_A(x) = 0 \quad \Rightarrow \quad \mu_{A\to B}(y) = 1$.

c) $\mu_B(y) = 1 \quad \Rightarrow \quad \mu_{A\to B}(y) = 1$.

d) $\mu_A(x) = 1$ và $\mu_B(y) = 0 \quad \Rightarrow \quad \mu_{A\to B}(y) = 0$.

e) $\mu_{A_1}(x) \le \mu_{A_2}(x) \quad \Rightarrow \quad \mu_{A_1\to B}(y) > \mu_{A_2\to B}(y)$.

f) $\mu_{B_1}(y) < \mu_{B_2}(y) \quad \Rightarrow \quad \mu_{A\to B_1}(y) \le \mu_{A\to B_2}(y)$.

Như vậy bất cứ một hàm $\mu_{A\to B}(y)$ nào thỏa mãn những tính chất trên đều có thể được sử dụng làm hàm thuộc cho tập mờ C là kết quả của mệnh đề hợp thành (1.43). Các hàm thuộc cho mệnh đề hợp thành mờ $A \Rightarrow B$ thường hay dùng bao gồm:

1) $\mu_{A\to B}(x, y) = \max\{\min\{\mu_A(x), \mu_B(y)\}, 1 - \mu_A(x)\}$ công thức Zadeh,

2) $\mu_{A\to B}(x, y) = \min\{1, 1 - \mu_A(x) + \mu_B(y)\}$ công thức Lukasiewicz,

3) $\mu_{A\to B}(x, y) = \max\{1 - \mu_A(x), \mu_B(y)\}$ công thức Kleene-Dienes.

Thật vậy, chẳng hạn như hàm thuộc $\mu_{A\to B}(x, y)$ xác định theo công thức Kleene-Dienes có

- Với mọi $\mu_B(y)$ và $\mu_A(x)=0$ thì $\mu_{A\to B}(x, y) = \max\{1, \mu_B(y)\} = 1$.

  Với mọi $\mu_A(x)$ và $\mu_B(y)=1$ thì $\mu_{A\to B}(x, y) = \max\{1 - \mu_A(x), 1\} = 1$.

- Khi $\mu_A(x) = 1$ và $\mu_B(y)=0$ thì $\mu_{A\to B}(x, y) = \max\{0, 0\} = 0$.

- $\mu_{A_1}(x) < \mu_{A_2}(x) \quad \Rightarrow \quad 1 - \mu_{A_1}(x) \ge 1 - \mu_{A_2}(x)$

$$\Rightarrow \quad \max\{1 - \mu_{A_1}(x), \mu_B(y)\} \ge \max\{1 - \mu_{A_2}(x), \mu_B(y)\}$$

$$\Rightarrow \quad \mu_{A_1\to B}(x, y) \ge \mu_{A_2\to B}(x, y).$$

- $\mu_{B_1}(y) < \mu_{B_2}(y) \quad \Rightarrow \quad \max\{1 - \mu_A(x), \mu_{B_1}(y)\} \le \max\{1 - \mu_A(x), \mu_{B_2}(y)\}$.

$$\Rightarrow \quad \mu_{A\to B_1}(x, y) \le \mu_{A\to B_2}(x, y).$$

Do mệnh đề hợp thành kinh điển $p \Rightarrow q$ luôn có giá trị đúng (giá trị logic 1) khi p sai nên sự chuyển đổi tương đương từ mệnh đề hợp thành $p \Rightarrow q$ kinh điển sang mệnh đề hợp thành mờ $A \Rightarrow B$ như định lý suy diễn 1.11 đã nêu sẽ sinh ra một nghịch lý khi ứng dụng trong điều khiển. Có thể thấy nghịch lý đó ở chỗ: mặc dù mệnh đề điều kiện

$$\chi = A,$$

không được thỏa mãn (có độ phụ thuộc bằng 0, tức là $\mu_A(x)=0$) nhưng mệnh đề kết luận

$$\gamma = B,$$

lại có độ thỏa mãn cao nhất $\mu_B(y) = 1$. Điều này dẫn tới mâu thuẫn, ví dụ như khi cài đặt mệnh đề

NẾU *ánh sáng = tối* THÌ *đèn = bật*.

Trong trường hợp trời nắng có

ánh sáng = nắng $\Rightarrow$ độ thỏa mãn $\mu_{tối}(x) = 0$

và như vậy đèn vẫn cứ được bật, do mệnh đề hợp thành có độ thoả mãn

$\mu_{tối \Rightarrow bật}(x, y)$ luôn bằng 1.

Đã có nhiều ý kiến được đề nghị nhằm khắc phục nhược điểm trên của định lý suy diễn 1.11 song nguyên tắc của *Mamdani*

"*Độ phụ thuộc của kết luận không được lớn hơn độ phụ thuộc của điều kiện*"

là có tính thuyết phục hơn cả và hiện đang được sử dụng nhiều nhất để mô tả mệnh đề hợp thành mờ trong điều khiển.

Biểu diễn nguyên tắc Mamdani dưới dạng công thức ta được

$$\mu_A(x) \ge \mu_{A\Rightarrow B}(y). \tag{1.44}$$

Do hàm $\mu_{A\Rightarrow B}(y)$ của tập mờ kết quả $B'=A\Rightarrow B$ chỉ phụ thuộc vào $\mu_A(x)$ và $\mu_B(y)$ và cũng như đã làm với phép hợp, giao ... hai tập mờ ta sẽ coi $\mu_{A\Rightarrow B}(y)$ như là một hàm của hai biến μ_A và μ_B , tức là

$$\mu_{A\Rightarrow B}(y) = \mu\,(\mu_A, \mu_B)$$

thì định nghĩa giả định 1.11 với sự sửa đổi lại theo nguyên tắc Mamdani sẽ được phát biểu như sau:

Định nghĩa 1.12 (Phép suy diễn mờ)

Giá trị của mệnh đề hợp thành mờ (1.43) là một tập mờ B' định nghĩa trên nền Y (không gian nền của B) và có hàm thuộc

$$\mu(\mu_A, \mu_B): [0,1]^2 \to [0,1]$$

thoa mãn

a) $\mu_A \geq \mu(\mu_A, \mu_B)$ với mọi $\mu_A, \mu_B \in [0,1]$.

b) $\mu(\mu_A, 0) = 0$ với mọi $\mu_A \in [0,1]$.

c) $\mu_{A_1} \leq \mu_{A_2} \Rightarrow \mu(\mu_{A_1}, \mu_B) \leq \mu(\mu_{A_2}, \mu_B)$.

d) $\mu_{B_1} \leq \mu_{B_2} \Rightarrow \mu(\mu_A, \mu_{B_1}) \leq \mu(\mu_A, \mu_{B_2})$.

Từ nguyên tắc của Mamdani và với định lý 1.12 có được các công thức xác định hàm thuộc cho mệnh đề hợp thành $B'=A\Rightarrow B$. Một trong số chúng là

1) $\mu(\mu_A, \mu_B) = \min\{\mu_A, \mu_B\}$ (1.45)

2) $\mu(\mu_A, \mu_B) = \mu_A \mu_B$ (1.46)

Thật vậy, chẳng hạn như với công thức (1.45) ta có

- Với mọi $\mu_A \in [0,1]$ có $\mu_A \geq \min\{\mu_A, \mu_B\} = \mu(\mu_A, \mu_B)$.
- $\mu(\mu_A, 0) = \min\{0, 0\} = 0$.
- $\mu_{A_1} \leq \mu_{A_2} \Rightarrow \min\{\mu_{A_1}, \mu_B\} \leq \min\{\mu_{A_2}, \mu_B\}$
  $\Rightarrow \mu(\mu_{A_1}, \mu_B) \leq \mu(\mu_{A_2}, \mu_B)$.
- $\mu_{B_1} \leq \mu_{B_2} \Rightarrow \min\{\mu_A, \mu_{B_1}\} \leq \min\{\mu_A, \mu_{B_2}\}$
  $\Rightarrow \mu(\mu_A, \mu_{B_1}) \leq \mu(\mu_A, \mu_{B_2})$.

Các công thức trên là hai công thức thường được sử dụng nhiều nhất trong kỹ thuật điều khiển mờ để mô tả mệnh đề hợp thành $A\Rightarrow B$. Chúng có tên chung là *quy tắc hợp thành*. Hai quy tắc hợp thành theo Mamdani là (1.45) và (1.46) sẽ là hai quy tắc hợp thành chính được sử dụng trong quyển sách này.

Quy tắc hợp thành MIN

Giá trị của mệnh đề hợp thành mờ (1.43) là một tập mờ B' định nghĩa trên nền Y (không gian nền của B) và có hàm thuộc

$$\mu_{B'}(y) = \min\{\mu_A \,,\, \mu_B(y)\} \tag{1.47}$$

Quy tắc hợp thành PROD

Giá trị của mệnh đề hợp thành mờ (1.43) là một tập mờ B' định nghĩa trên nền Y (không gian nền của B) và có hàm thuộc

$$\mu_{B'}(y) = \mu_A\, \mu_B(y) \tag{1.48}$$

Thoạt mới nhìn, hai quy tắc hợp thành trên (còn được gọi là phép suy diễn mờ) có dạng gần giống như công thức (1.37a) và (1.37b) xác định hàm thuộc $\mu_{A\Rightarrow B}(x, y)$ của tập giao hai tập mờ. Tuy nhiên chúng lại khác nhau ở bản chất là trong khi tập mờ kết quả của quy tắc hợp thành $\mu_{B'}(y)$ được định nghĩa trên nền của B, còn của $\mu_{A\Rightarrow B}(x, y)$ lại được định nghĩa trên tập nền tích của hai tập nền của A và B. Ngoài ra $\mu_{B'}(y)$ chỉ được xác định khi đã biết cụ thể một giá trị của μ_A, tức là $\mu_{B'}(y)$ phụ thuộc vào giá trị rõ x_0 ở đầu vào còn $\mu_{A\Rightarrow B}(x, y)$ thì không.

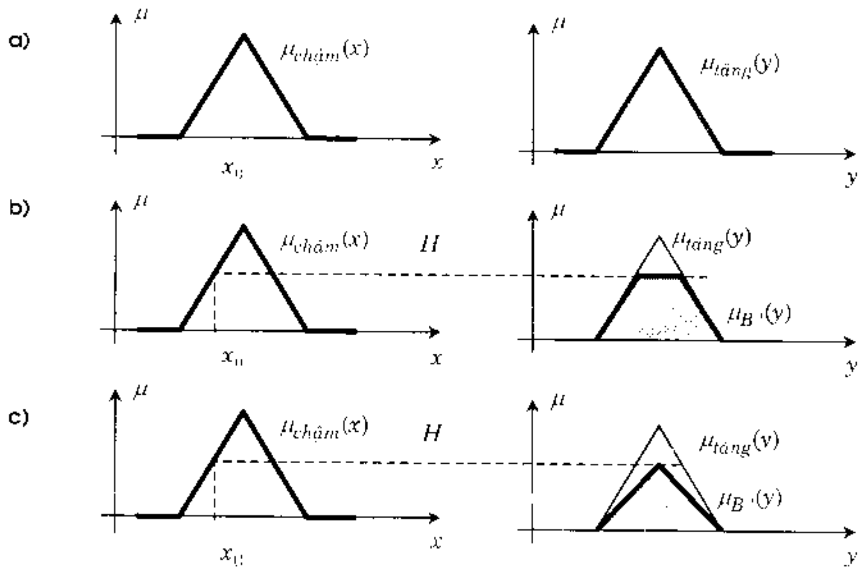
Giả sử rằng biến ngôn ngữ χ chỉ tốc độ và γ chỉ sự tác động ga xe. Luật điều khiển cho xe chạy với tốc độ trung bình không đổi sẽ tương đương với mệnh đề hợp thành mờ một điều kiện đầu vào

$$\text{NẾU } \chi = \textit{chậm} \text{ THÌ } \gamma = \textit{tăng} \tag{1.49}$$

với $\mu_{chậm}(x)$, $\mu_{tang}(y)$ như trong *hình 1.13a)*. Kết quả của mệnh đề hợp thành (1.49) khi sử dụng quy tắc MIN (1.47) cho một giá trị rõ $x=x_0$ đầu vào sẽ là một tập mờ B' có tập nền cùng với tập nền của $\mu_{tang}(y)$ và hàm thuộc $\mu_{B'}(y)$ là phần dưới của hàm $\mu_{tang}(y)$ bị cắt bởi đường $H=\mu_{chậm}(x_0)$ (xem *hình 1.18b)*).

Hình 1.18c) biểu diễn hàm thuộc của B' cho mệnh đề hợp thành (1.49) được xác định với quy tắc PROD.

Như vậy ta có hai quy tắc hợp thành xác định giá trị mờ B' của mệnh đề hợp thành. Nếu hàm thuộc $\mu_{B'}(y)$ của B' thu được theo quy tắc MIN thì mệnh đề hợp thành có tên gọi là mệnh đề hợp thành MIN. Cũng như vậy mệnh đề hợp thành sẽ được gọi là PROD, nếu $\mu_{B'}(y)$ xác định theo quy tắc PROD.



Hình 1.18 a) Hàm thuộc $\mu_{chậm}(x)$ và $\mu_{tăng}(x)$

b) $\mu_{B'}(y)$ xác định theo quy tắc hợp thành MIN.

c) $\mu_{B'}(y)$ xác định theo quy tắc hợp thành PROD.

Ký hiệu giá trị mờ đầu ra là B' ứng với một giá trị rõ x_0 tại đầu vào thì hàm thuộc của B' với quy tắc hợp thành MIN sẽ là

$$\mu_{B'}(y) = \min\{\mu_A(x_0), \mu_B(y)\}. \tag{1.50}$$

Gọi

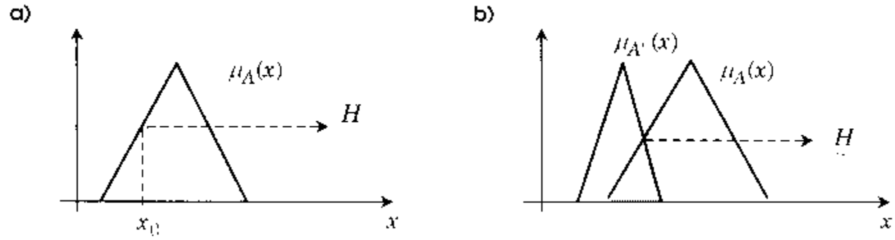
$$H = \mu_A(x_0) \tag{1.51}$$

là *độ thỏa mãn mệnh đề điều kiện* hay ngắn gọn là *độ thỏa mãn* thì

$$\mu_{B'}(y) = \min\{H, \mu_B(y)\}. \tag{1.52}$$

Với quy tắc hợp thành PROD, hàm thuộc của B' sẽ là

$$\mu_{B'}(y) = \mu_A(x_0)\mu_B(y) = H \cdot \mu_B(y). \tag{1.53}$$



Hình 1.19 Mô tả độ thỏa mãn.

a) Giá trị đầu vào rõ b) Giá trị đầu vào mờ

Trong trường hợp tín hiệu dầu vào A' là một giá trị mờ với hàm thuộc $\mu_{A'}(x)$, dầu ra B' cũng là một giá trị mờ có hàm thuộc $\mu_{B'}(y)$ là phần dưới cua hàm $\mu_B(y)$ bị chặn trên bởi độ thỏa mãn H được xác định theo nguyên tắc "*tình huống xấu nhất*" như sau:

$$H = \max_x \min\{\mu_{A'}(x), \mu_A(x)\} \quad \text{(xem } \textit{hình 1.19}\text{)}. \tag{1.54}$$

1.5.3 Luật hợp thành mờ

Hàm thuộc $\mu_{B'}(y)$ trong ví dụ trên với một giá trị vật lý rõ $x=x_0$ có cùng tập nền với $\mu_{tang}(y)$. Tổng quát lên thì hàm thuộc $\mu_{A\Rightarrow B}(y)$ của mệnh đề hợp thành $A\Rightarrow B$, bây giờ sẽ được ký hiệu ngắn gọn lại thành R, tại một giá trị rõ $x=x_0$ là một hàm thuộc cho một giá trị mờ nào đó của biến ngôn ngữ γ.

Luật hợp thành là tên chung gọi mô hình R biểu diễn một hay nhiều hàm thuộc cho một hay nhiều mệnh đề hợp thành, nói cách khác *luật hợp thành* được hiểu là một tập hợp của nhiều mệnh đề hợp thành. Một luật hợp thành chỉ có một mệnh đề hợp thành được gọi là *luật hợp thành đơn*. Ngược lại nếu nó có nhiều hơn một mệnh đề hợp thành, ta sẽ gọi nó là *luật hợp thành kép*. Phần lớn các hệ mờ trong thực tế đều có mô hình là luật hợp thành kép.

Ta hãy xét một ví dụ về luật hợp thành R biểu diễn mô hình lái ô tô gồm 3 mệnh đề hợp thành R_1, R_2, R_3 cho biến tốc độ χ và biến ga γ như sau

R_1: NẾU χ = *chậm* THÌ γ = *tăng* hoặc

R_2: NẾU χ = *trung bình* THÌ γ = *giữ nguyên* hoặc

R_3: NẾU χ = *nhanh* THÌ γ = *giảm*.

Với mỗi một giá trị vật lý x_0 của biến tốc độ đầu vào thì thông qua phép suy diễn mờ ta có ba tập mờ B_1', B_2', B_3' từ ba mệnh đề hợp thành R_1, R_2, R_3 của luật hợp thành R. Lần lượt ta gọi các hàm thuộc của ba tập mờ kết quả đó là $\mu_{B_1'}(y), \mu_{B_2'}(y)$ và $\mu_{B_3'}(y)$. Giá trị của luật hợp thành R ứng với x_0 được hiểu là tập mờ R' thu được qua phép hợp ba tập mờ B_1', B_2' và B_3':

$$R' = B_1' \cup B_2' \cup B_3'. \tag{1.55}$$

Nếu các hàm thuộc $\mu_{B_1'}(y), \mu_{B_2'}(y), \mu_{B_3'}(y)$ thu được theo quy tắc MIN và phép hợp (1.55) được thực hiện theo luật max (xem (1.25) và (1.30a)) thì R có tên gọi là luật hợp thành max–MIN. Cũng như vậy R sẽ còn có những tên gọi khác như:

- luật hợp thành max PROD, nếu $\mu_{B_1'}(y), \mu_{B_2'}(y), \mu_{B_3'}(y)$ được xác định theo quy tắc hợp thành PROD và phép hợp (1.55) là phép hợp theo luật max.
- luật hợp thành sum-MIN, nếu $\mu_{B_1'}(y), \mu_{B_2'}(y), \mu_{B_3'}(y)$ được xác định theo quy tắc hợp thành MIN và phép hợp (1.55) là phép hợp Lukasiewicz (xem (1.27), (1.30b)).
- luật hợp thành sum-PROD, nếu $\mu_{B_1'}(y), \mu_{B_2'}(y), \mu_{B_3'}(y)$ được xác định theo quy tắc hợp thành PROD và phép hợp (1.55) là phép hợp Lukasiewicz (xem (1.27), (1.30b)).

Tóm lại, để xác định hàm thuộc $\mu_{R'}(y)$ của giá trị đầu ra R' của một luật hợp thành có n mệnh đề hợp thành $R_1, R_2, \ldots, R_n$ phải thực hiện các bước:

1) Xác định độ thỏa mãn $H_1, H_2, \ldots, H_n$ theo (1.51) hoặc (1.54).

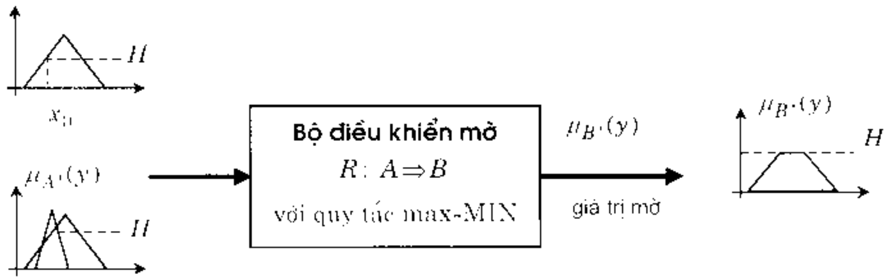
2) Tính $\mu_{B_1'}(y), \mu_{B_2'}(y), \cdots, \mu_{B_n'}(y)$ theo (1.52) hoặc (1.53).

3) Xác định $\mu_{R'}(y)$ theo (1.25), (1.30a) hoặc (1.27), (1.30b).

Nếu xem luật hợp thành R chỉ có một mệnh đề hợp thành

R_1: NẾU $\chi = A$ THÌ $\gamma = B$

như là luật điều khiển của *bộ điều khiển mờ* một vào một ra (SISO) thì đầu ra sẽ là một giá trị mờ có hàm thuộc $\mu_{B'}(y)$ – *hình 1.20*.



Hình 1.20: Bộ điều khiển mờ với quy tắc max-MIN.

Một luật hợp thành có các mệnh đề điều kiện và kết luận là những mệnh đề đơn, ví dụ như:

R_1: NẾU $\chi = A_1$ THÌ $\gamma = B_1$ hoặc

R_2: NẾU $\chi = A_2$ THÌ $\gamma = B_2$ hoặc

$\vdots$

R_n: NẾU $\chi = A_n$ THÌ $\gamma = B_n$.

được gọi là luật hợp thành có cấu trúc SISO (*một vào, một ra*). Ngược lại luật hợp thành có m biến ngôn ngữ vào $\chi_1, \chi_2, \cdots, \chi_m$ và một biến ngôn ngữ ra γ với cấu trúc dạng

R_1: NẾU $\chi_1 = A_{11}$ VÀ $\chi_2 = A_{12}$ VÀ $\cdots$ VÀ $\chi_m = A_{1m}$ THÌ $\gamma = B_1$ hoặc

R_2: NẾU $\chi_1 = A_{21}$ VÀ $\chi_2 = A_{22}$ VÀ $\cdots$ VÀ $\chi_m = A_{2m}$ THÌ $\gamma = B_2$ hoặc

$\vdots$

R_n: NẾU $\chi_1 = A_{n1}$ VÀ $\chi_2 = A_{n2}$ VÀ $\cdots$ VÀ $\chi_m = A_{nm}$ THÌ $\gamma = B_n$.

có tên gọi là luật hợp thành MISO (*nhiều vào, một ra*).

1.5.4 Thuật toán thực hiện luật hợp thành đơn max-MIN, max-PROD có cấu trúc SISO

Luật hợp thành max-MIN

Luật hợp thành max-MIN là tên gọi mô hình (ma trận) R của luật hợp thành chỉ mà giá trị biến mờ của nó được xác định theo quy tắc max-MIN.

Xét luật hợp thành SISO chỉ có một mệnh đề hợp thành

R_1: NẾU $\chi = A$ THÌ $\gamma = B$.

Trước tiên hai hàm thuộc $\mu_A(x)$ và $\mu_B(y)$ được rời rạc hóa với tần số rời rạc đủ nhỏ để không bị mất thông tin. Chẳng hạn trong ví dụ về biến vận tốc v (biến ngôn ngữ), hai giá trị mờ $\mu_{chậm}(x)$, $\mu_{tang}(y)$ được rời rạc hóa tại các điểm

$x \in \{0,1 ; 0,2 ; 0,3 ; 0,4 ; 0,5\}$

$y \in \{0,5 ; 0,6 ; 0,7 ; 0,8 ; 0,9\}$.

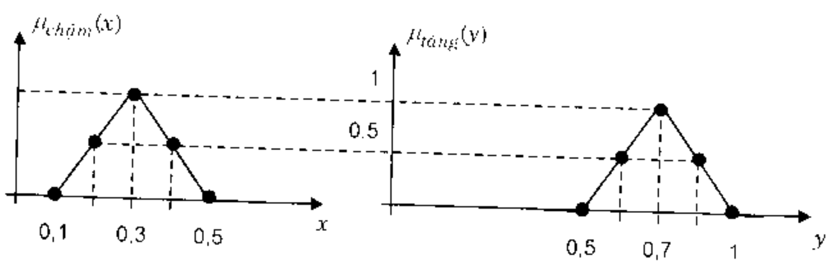
Với các điểm rời rạc này thì theo (1.51) và (1.52), khi đầu vào là giá trị rõ x_0=0,2 hàm thuộc $\mu_{B'}(y)$ tại điểm y=0,7 sẽ là

$$\mu_{B'}(0,7)\big|_{0,2} = \mu_R(0,2 ; 0,7) = \min\{\mu_{chậm}(0,2), \mu_{tăng}(0,7)\} = \min\{0,5 ; 1\} = 0,5$$

hoặc

$$\mu_{B'}(0,7)\big|_{0,3} = \mu_R(0,3 ; 0,7) = \min\{\mu_{chậm}(0,3), \mu_{tăng}(0,7)\} = \min\{1 ; 1\} = 1$$

$\vdots$



Hình 1.21: Rời rạc hóa hàm thuộc.

Nhóm tất cả các giá trị có được của $\mu_{B'}(y)\big|_x = \mu_R(x, y)$, gồm 5×5=25 giá trị, thành ma trận R (được gọi là *luật hợp thành* max-MIN) gồm 5 hàng, 5 cột

R		y				
		0,5	0,6	0,7	0,8	0,9
	0,1	0	0	0	0	0
→	0,2	0	0,5	0,5	0,5	0
x	0,3	0	0,5	1	0,5	0
	0,4	0	0,5	0,5	0,5	0
	0,5	0	0	0	0	0

thì khi tín hiệu đầu vào là một giá trị rõ $x_0=0,2$ tín hiệu mờ đầu ra B' sẽ có hàm thuộc dạng rời rạc

$$\mu_{B'}(y) = \mu_R(0{,}2, y) = \{0\,;\,0{,}5\,;\,0{,}5\,;\,0{,}5\,;\,0\},$$

tương ứng với các phần tử trong hàng x=0,2.

Cách biểu diễn này rất thuận tiện cho việc xác định hàm thuộc của tín hiệu ra dưới dạng nhân ma trận. Ví dụ với tập 5 phần tử cho tín hiệu đầu vào

$$x \in \{0{,}1\,;\,0{,}2\,;\,0{,}3\,;\,0{,}4\,;\,0{,}5\}$$

thì ứng với $x_0 = 0{,}2$ (phần tử thứ 2) là vector

$$\underline{a}^T = (0\,;\,1\,;\,0\,;\,0\,;\,0) \tag{1.56}$$

và do đó

$$\mu_{B'}(y) = \mu_R(x_0, y) = \underline{a}^T \cdot R = \{0\,;\,0{,}5\,;\,0{,}5\,;\,0{,}5\,;\,0\}. \tag{1.57}$$

Tổng quát lên cho một giá trị rõ x_0 bất kỳ

$$x_0 \in X = \{0{,}1\,;\,0{,}2\,;\,0{,}3\,;\,0{,}4\,;\,0{,}5\}$$

tại đầu vào, vector chuyển vị $\underline{a}$ sẽ có dạng

$$\underline{a}^T = (a_1\,,\,a_2\,,\,a_3\,,\,a_4\,,\,a_5)$$

trong đó chỉ có một phần tử a_i duy nhất có chỉ số i là chỉ số của x_0 trong X có giá trị bằng 1, các phần tử còn lại đều bằng 0. Hàm thuộc (rời rạc) $\mu_{B'}(y)$ được xác định với

$$\mu_{B'}(y) = \underline{a}^T \cdot R = (a_1, a_2, a_3, a_4, a_5)\begin{pmatrix} r_{11} & \cdots & r_{15} \\ \vdots & & \vdots \\ r_{51} & \cdots & r_{55} \end{pmatrix} = (l_1, l_2, l_3, l_4, l_5) \quad (1.58)$$

trong đó

$$l_k = \sum_{i=1}^{5} a_i r_{ik}$$

Để tránh phải cài đặt thuật toán nhân ma trận của đại số tuyến tính cho việc tính hàm thuộc $\mu_{B'}(y)$ theo (1.58) và qua đó tăng tốc độ xử lý, phép tính nhân ma trận kiểu (1.58) được thay bởi luật max-min của Zadeh với max (phép lấy cực đại) thay vào vị trí phép cộng và min (phép lấy cực tiểu) thay vào vị trí phép nhân như sau

$$l_k = \max_{1 \le i \le 5} \min \{a_i, r_{ik}\}. \quad (1.59)$$

Kết quả hai phép tính (1.58) và (1.59) với đầu vào là một giá trị rõ hoàn toàn giống nhau.

Luật hợp thành max-PROD

Cũng giống như đã làm với luật hợp thành max-MIN, ma trận R của luật hợp thành max-PROD được xây dựng gồm các hàng là m giá trị rời rạc của đầu ra $\mu_{B'}(y_1)$, $\mu_{B'}(y_2)$, $\cdots$, $\mu_{B'}(y_m)$ cho n giá trị rõ đầu vào x_1, x_2, $\cdots$, x_n. Như vậy ma trận R sẽ có n hàng và m cột.

Lại làm với ví dụ (1.49) cho 5 giá trị đầu vào

$$\{x_1, x_2, x_3, x_4, x_5\} = \{0{,}1\ ;\ 0{,}2\ ;\ 0{,}3\ ;\ 0{,}4\ ;\ 0{,}5\}$$

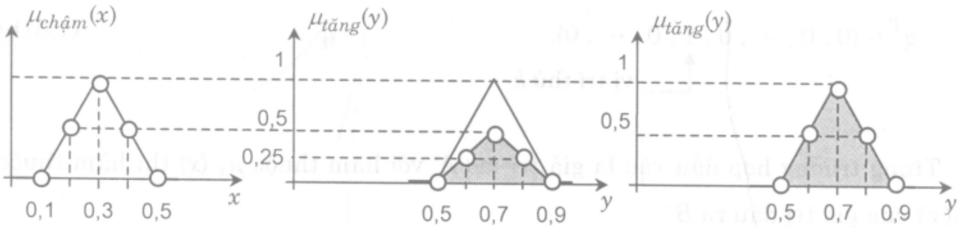
thì với từng giá trị x_i năm giá trị của hàm thuộc đầu ra tương ứng $\mu_{B'}(0{,}5)$, $\mu_{B'}(0{,}6)$, $\mu_{B'}(0{,}7)$, $\mu_{B'}(0{,}8)$ và $\mu_{B'}(0{,}9)$ sẽ như sau (xem *hình 1.22*)

R		y				
		0,5	0,6	0,7	0,8	0,9
i=1	0,1	0	0	0	0	0
i=2	0,2	0	0,25	0,5	0,25	0
i=3	0,3	0	0,5	1	0,5	0
i=4	0,4	0	0,25	0,5	0,25	0
i=5	0,5	0	0	0	0	0

Từ ma trận R trên (được gọi *là luật hợp thành* max-PROD), hàm thuộc $\mu_{B'}(y)$ của giá trị đầu ra khi đầu vào là giá trị rõ x_4=0,4 cũng được xác định bằng công thức (1.59), tức là

$$\underline{a}^T = (0\ ;\ 0\ ;\ 0\ ;\ 1\ ;\ 0)$$

và $\quad \mu_{B'}(y) = \mu_R(x_4,\ y) = \underline{a}^T \cdot R = \{0\ ;\ 0{,}25\ ;\ 0{,}5\ ;\ 0{,}25\ ;\ 0\}$



Hình 1.22: Xây dựng R theo quy tắc max-PROD.

Để rút ngắn thời gian tính và cũng để mở rộng công thức trên cho trường hợp đầu vào là giá trị mờ, phép nhân ma trận $\underline{a}^T \cdot R$ cũng được thay bằng luật max-min (1.59) của Zadeh như đã làm cho luật hợp thành max-MIN.

Thuật toán xây dựng R

Phương pháp xây dựng R cho mệnh đề hợp thành một điều kiện $\hat{R}: A \Rightarrow B$, theo max-MIN hay max-PROD, để xác định hàm thuộc cho giá trị mờ B' đầu ra đã được trình bày với ví dụ về mệnh đề hợp thành (1.49) trên đây hoàn toàn có thể mở rộng tương tự cho một mệnh đề hợp thành SISO bất kỳ nào khác dạng

NẾU $\chi = A$ THÌ $\gamma = B$,

trong đó ma trận, hay luật hợp thành R không nhất thiết phải là một ma trận vuông như đã làm trong ví dụ trên. Số chiều của R phụ thuộc vào số điểm lấy mẫu của $\mu_A(x)$ và $\mu_B(y)$ khi rời rạc các hàm thuộc tập mờ A và B.

Chẳng hạn với n điểm mẫu $x_1, x_2, \cdots, x_n$ của hàm $\mu_A(x)$ và m điểm mẫu $y_1, y_2, \cdots, y_m$ của hàm $\mu_B(y)$ thì luật hợp thành R là một ma trận n hàng m cột như sau

$$R = \begin{pmatrix} \mu_R(x_1, y_1) & \cdots & \mu_R(x_1, y_m) \\ \vdots & & \vdots \\ \mu_R(x_n, y_1) & \cdots & \mu_R(x_n, y_m) \end{pmatrix} = \begin{pmatrix} r_{11} & \cdots & r_{1m} \\ \vdots & & \vdots \\ r_{n1} & \cdots & r_{nm} \end{pmatrix} \tag{1.60}$$

Hàm thuộc $\mu_{B'}(y)$ của giá trị đầu ra ứng với giá trị rõ đầu vào x_k được xác định theo

$$\mu_{B'}(y) = \underline{a}^T \cdot R \tag{1.61a}$$

với

$$\underline{a}^T = (0\ ,\ 0\ ,\ \cdots\ ,\ 0\ ,\ \underset{\uparrow\ \text{vị trí thứ } k}{1}\ ,\ 0\ ,\ \cdots\ ,\ 0). \tag{1.61b}$$

Trong trường hợp đầu vào là giá trị mờ A' với hàm thuộc $\mu_{A'}(x)$ thì hàm thuộc $\mu_{B'}(y)$ của giá trị đầu ra B'

$$\mu_{B'}(y) = (\ l_1\ ,\ l_2\ ,\ \cdots\ ,\ l_m\ )$$

cũng được tính theo công thức (1.61) và

$$l_k = \max_{1 \le i \le n} \min\ \{a_i\ ,\ r_{ik}\}, \qquad k = 1, 2, \cdots, m, \tag{1.62}$$

trong đó $\underline{a}$ là vector gồm các giá trị rời rạc của hàm thuộc $\mu_{A'}(x)$ của A' tại các điểm

$$x \in X = \{x_1, x_2, \cdots, x_n\}.$$

tức là

$$\underline{a}^T = \big(\mu_{A'}(x_1)\ ,\ \mu_{A'}(x_2)\ ,\ \cdots,\ \mu_{A'}(x_n)\big).$$

Ưu điểm nổi bật của luật max-min (1.62) của Zadeh là có thể xác định ngay được R thông qua tích *dyadic*, tức là tích của một vector với một vector chuyển vị. Chẳng hạn với n điểm rời rạc $x_1, x_2, \cdots, x_n$ của tập nền của A và m điểm rời rạc $y_1, y_2, \cdots, y_m$ của tập nền của B thì từ hai vector

$$\underline{\mu}_A^T = \big(\mu_A(x_1),\ \mu_A(x_2),\ \cdots\ ,\ \mu_A(x_n)\big) \tag{1.63a}$$

và $$\underline{\mu}_B^T = \big(\mu_B(y_1),\ \mu_B(y_2),\ \cdots\ ,\ \mu_B(y_m)\big) \tag{1.63b}$$

suy ngay ra được

$$R = \mu_A \cdot \underline{\mu}_B^T \quad . \tag{1.63c}$$

trong đó nếu quy tắc áp dụng là max-MIN thì phép nhân phải được thay bằng phép tính lấy cực tiểu (min), với quy tắc max-PROD thì thực hiện phép nhân như bình thường.

Hãy lấy việc xác định R của

$$\mu_{B'}(y)\big|_x = \mu_R(x, y)$$

cho luật hợp thành max-MIN (1.49) làm ví dụ. Với 5 điểm rời rạc của X (tập nền của A) cho trong *hình 1.21*

$$\{x_1, x_2, x_3, x_4, x_5\} = \{0{,}1 ; 0{,}2 ; 0{,}3 ; 0{,}4 ; 0{,}5\}$$

và 5 điểm rời rạc của Y (tập nền của B)

$$\{y_1, y_2, y_3, y_4, y_5\} = \{0{,}5 ; 0{,}6 ; 0{,}7 ; 0{,}8 ; 0{,}9\}$$

thì

$$\underline{\mu}_A^T = (0 \quad 0{,}5 \quad 1 \quad 0{,}5 \quad 0)$$

và

$$\underline{\mu}_B^T = (0 \quad 0{,}5 \quad 1 \quad 0{,}5 \quad 0).$$

Khi sử dụng quy tắc max-MIN (phép nhân được thay bằng min), luật hợp thành max-MIN sẽ như sau

$$R = \begin{pmatrix} 0 \\ 0{,}5 \\ 1 \\ 0{,}5 \\ 0 \end{pmatrix} (0 \quad 0{,}5 \quad 1 \quad 0{,}5 \quad 0) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0{,}5 & 0{,}5 & 0{,}5 & 0 \\ 0 & 0{,}5 & 1 & 0{,}5 & 0 \\ 0 & 0{,}5 & 0{,}5 & 0{,}5 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

và với quy tắc max-PROD (luật hợp thành max-PROD)

$$R = \begin{pmatrix} 0 \\ 0{,}5 \\ 1 \\ 0{,}5 \\ 0 \end{pmatrix} (0 \quad 0{,}5 \quad 1 \quad 0{,}5 \quad 0) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0{,}25 & 0{,}5 & 0{,}25 & 0 \\ 0 & 0{,}5 & 1 & 0{,}5 & 0 \\ 0 & 0{,}25 & 0{,}5 & 0{,}25 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

1.5.5 Thuật toán xác định luật hợp thành đơn có cấu trúc MISO

Một mệnh đề hợp thành với d mệnh đề điều kiện

$$\text{NẾU } \chi_1 = A_1 \text{ VÀ } \chi_2 = A_2 \text{ VÀ } \cdots \text{ VÀ } \chi_d = A_d \text{ THÌ } \gamma = B \tag{1.64}$$

bao gồm d biến ngôn ngữ đầu vào $\chi_1, \chi_2, \cdots, \chi_d$ và một biến đầu ra γ cũng được mô hình hóa giống như việc mô hình hóa mệnh đề hợp thành có một điều kiện, trong đó liên kết VÀ giữa các mệnh đề (hay giá trị mờ) được thực hiện bằng phép giao các tập mờ $A_1, A_2, \cdots, A_d$ với nhau theo công thức (1.37). Kết quả của phép giao sẽ là độ thỏa mãn H. Các bước xây dựng luật hợp thành R như sau

- rời rạc hóa miền xác định hàm thuộc $\mu_{A_1}(x_1), \mu_{A_2}(x_2), \cdots, \mu_{A_d}(x_d), \mu_B(y)$ của các mệnh đề điều kiện và mệnh đề kết luận.
- xác định độ thỏa mãn H cho từng vector các giá trị rõ đầu vào là vector tổ hợp d điểm mẫu thuộc miền xác định của các hàm thuộc $\mu_{A_i}(x_i)$, i=1,$\cdots$, d. Chẳng hạn với một vector các giá trị rõ đầu vào

  $$\underline{x} = \begin{pmatrix} c_1 \\ \vdots \\ c_d \end{pmatrix},$$

  trong đó c_i, i=1,$\cdots$, d là một trong các điểm mẫu miền xác định của $\mu_{A_i}(x_i)$, thì

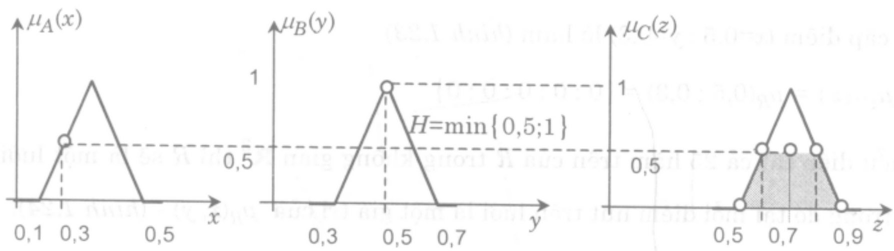
  $$H = \min\{\, \mu_{A_1}(c_1), \mu_{A_2}(c_2), \cdots, \mu_{A_d}(c_d) \,\}. \tag{1.65}$$

- lập R gồm các hàm thuộc giá trị mờ đầu ra cho từng vector các giá trị đầu vào theo nguyên tắc:

  $\mu_{B'}(y) = \min\{H, \mu_B(y)\}$ nếu quy tắc sử dụng là max-MIN hoặc

  $\mu_{B'}(y) = H \cdot \mu_B(y)$ nếu quy tắc sử dụng là max-PROD.

Không như luật hợp thành có cấu trúc SISO, luật hợp thành R của (1.63) với d mệnh đề điều kiện không thể biểu diễn dưới dạng ma trận được nữa mà thành một lưới trong không gian d+1 chiều. Nguyên nhân nằm ở chỗ các tập mờ đầu vào A_1, A_2, $\cdots$ A_d không cùng một không gian nền nên qua phép giao mờ tập mờ thu được sẽ phải được định nghĩa trên nền mới là tập tích của d không gian nền đã cho.



Hình 1.23: Xây dựng R cho luật hợp thành MISO với hai mệnh đề điều kiện.

Hãy xét một mệnh đề hợp thành với hai mệnh đề điều kiện

NẾU $\alpha = A$ VÀ $\beta = B$ THÌ $\gamma = C$

làm một ví dụ (xem *hình 1.23*). Luật hợp thành R của nó có dạng sau

R: $A \wedge B \Rightarrow C$.

Các bước xây dựng R như sau:

1) *Rời rạc hóa các hàm thuộc*:

a) Hàm thuộc $\mu_A(x)$ được rời rạc hóa tại 5 điểm

$x \in \{0{,}1\ ;\ 0{,}2\ ;\ 0{,}3\ ;\ 0{,}4\ ;\ 0{,}5\}$.

b) Hàm thuộc $\mu_B(y)$ được rời rạc hóa tại 5 điểm

$y \in \{0{,}3\ ;\ 0{,}4\ ;\ 0{,}5\ ;\ 0{,}6\ ;\ 0{,}7\}$.

c) Hàm thuộc $\mu_C(z)$ được rời rạc hóa tại 5 điểm

$z \in \{0{,}5\ ;\ 0{,}6\ ;\ 0{,}7\ ;\ 0{,}8\ ;\ 0{,}9\}$.

2) *Lập R gồm các hàm thuộc cho từng vector giá trị đầu vào*:

Như vậy sẽ có tất cả 5×5=25 cặp điểm giá trị đầu vào và ứng với từng cặp điểm đầu vào là một hàm thuộc $\mu_{C'}(z)$ của biến mờ đầu ra C'.

Ví dụ cho cặp điểm (x=0,2 ; y=0,5) là hàm

$$\mu_{C'}(z) = \mu_R(0{,}2\ ;\ 0{,}5) = \{0\ ;\ 0{,}5\ ;\ 0{,}5\ ;\ 0{,}5\ ;\ 0\},$$

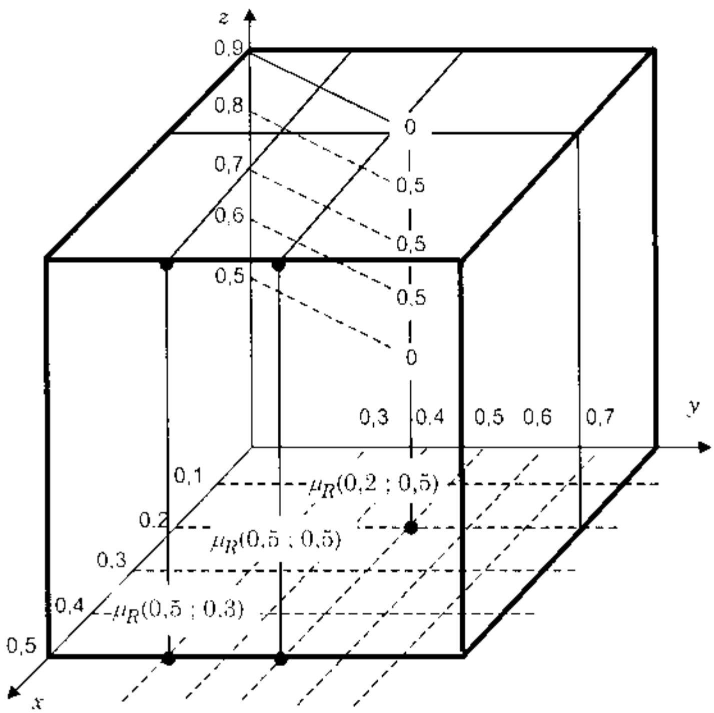
cho cặp điểm (x=0,5 ; y=0,5) là hàm

$$\mu_{C'}(z) = \mu_R(0{,}5\ ;\ 0{,}5) = \{0\ ;\ 0\ ;\ 0\ ;\ 0\ ;\ 0\}$$

và cho cặp điểm (x=0,5 ; y=0,3) là hàm (*hình 1.23*)

$$\mu_{C'}(z) = \mu_R(0{,}5\ ;\ 0{,}3) = \{0\ ;\ 0\ ;\ 0\ ;\ 0\ ;\ 0\}$$

Biểu diễn tất cả 25 hàm trên của R trong không gian $\mathbb{R}^3$ thì R sẽ là một lưới 3 chiều, trong đó tại mỗi điểm nút trên lưới là một giá trị của $\mu_R(x, y)$ - (*hình 1.24*).



Hình 1.24: Biểu diễn R trong lưới 3 chiều.

1.5.6 Thuật toán xác định luật hợp thành kép max-MIN, max-PROD

Trong thực tế *ít có một bộ điều khiển mờ nào chỉ làm việc với một mệnh đề hợp thành* mà thông thường với nhiều mệnh đề hợp thành, hay còn gọi là một tập các luật điều khiển R_k. Phần trên đã trình bầy cách mô hình hóa một mệnh đề hợp thành theo quy tắc MIN để có luật hợp thành max-MIN hoặc theo PROD để có được luật hợp thành max-PROD. Mục này được dành riêng để mô tả phương pháp liên kết các mệnh đề hợp thành R_k lại với nhau trong một luật hợp thành chung và cũng qua đó mà nêu được ý nghĩa của ký hiệu "max" sử dụng trong tên gọi luật hợp thành như max-MIN hay max-PROD.

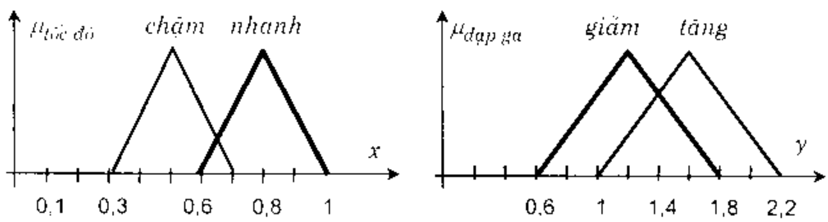
Luật hợp thành có hai mệnh đề hợp thành

Xét một luật hợp thành gồm hai mệnh đề hợp thành của ví dụ về lái ô tô

R_1: NẾU χ = *chậm* THÌ γ = *tăng* hoặc (1.66a)

R_2: NẾU χ = *nhanh* THÌ γ = *giảm*, (1.66b)

trong đó biến ngôn ngữ χ chỉ tốc độ xe và γ chỉ sự tác động vào bàn đạp ga xe. Hàm thuộc của giá trị mờ *chậm*, *nhanh* cho biến tốc độ và *tăng*, *giảm* cho biến bàn đạp ga được mô tả trong *hình 1.25*.



Hình 1.25: Hàm thuộc của các giá trị *nhanh*, *chậm* cho biến tốc độ và *tăng*, *giảm* cho biến bàn đạp ga.

Ký hiệu R' là giá trị của luật hợp thành R thì

$$R' = R_1' \cup R_2', \tag{1.67}$$

trong đó R_1', R_2' là giá trị của từng mệnh đề hợp thành.

Ký hiệu hàm thuộc của R_1' là $\mu_{R_1'}(y)$ và của R_2' là $\mu_{R_2'}(y)$, thì theo (1.30a) hàm thuộc $\mu_{R'}(y)$ của R' sẽ như sau

$$\mu_{R'}(y) = \max\{\, \mu_{R_1'}(y), \mu_{R_2'}(y) \,\}. \tag{1.68}$$

Cũng như đã làm với luật có một mệnh đề hợp thành, phương pháp triển khai hợp hai luật điều khiển (1.66) sau đây sẽ được mô tả trước tiên với một giá trị rõ x_0 tại đầu vào.

Đối với luật điều khiển R_1 thì (*hình 1.26a*)

- độ thỏa mãn: $H_1 = \mu_{chậm}(x_0)$,
- giá trị mờ đầu ra B_1: $\mu_{R_1'}(y) = \min\{H_1, \mu_{tăng}(y)\}$.

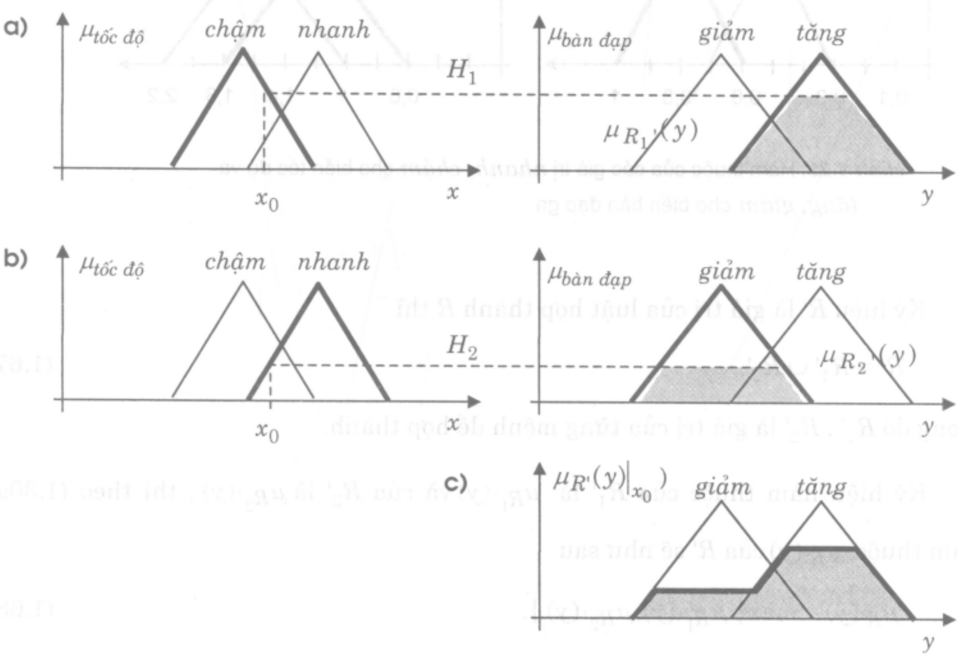
Đối với luật điều khiển R_2 thì (*hình 1.26b*)

- độ thỏa mãn: $H_2 = \mu_{nhanh}(x_0)$
- giá trị mờ đầu ra B_2: $\mu_{R_2'}(y) = \min\{H_2, \mu_{giảm}(y)\}$

Từ đây có được theo (1.68)

$$\mu_{R'}(y)\Big|_{x_0} = \max\{\ \mu_{R_1'}(y), \mu_{R_2'}(y)\ \}. \qquad (1.69)$$

và đó chính là hàm thuộc của giá trị mờ đầu ra R' của luật gồm hai mệnh đề hợp thành (1.66) khi đầu vào là một giá trị rõ x_0 (*hình 1.26c*).



Hình 1.26: Hàm thuộc của hợp hai luật điều khiển.
a) Xác định hàm thuộc đầu ra của luật điều khiển thứ nhất.
b) Xác định hàm thuộc đầu ra của luật điều khiển thứ hai.
c) Hàm thuộc đầu ra của luật hợp thành.

Để triển khai (1.68), tức là xác định luật hợp thành R, trước hết hai tập nền X và Y của các giá trị *chậm*, *nhanh* (cho biến tốc độ) và của *tăng*, *giảm* (cho biến bàn đạp ga xe) được rời rạc hóa, giả sử tại các điểm:

$X = \{x_1, x_2, \cdots, x_n\}$ $\qquad$ n điểm mẫu,

$Y = \{y_1, y_2, \cdots, y_m\}$ $\qquad$ m điểm mẫu.

Bốn vector những giá trị của hàm thuộc $\mu_{chậm}(x)$, $\mu_{nhanh}(x)$, $\mu_{tăng}(y)$, $\mu_{giảm}(y)$ khi Fuzzy hóa các điểm đó sẽ là

$$\underline{\mu}^T_{chậm} = \big(\mu_{chậm}(x_1), \mu_{chậm}(x_2), \cdots, \mu_{chậm}(x_n)\big), \tag{1.70a}$$

$$\underline{\mu}^T_{nhanh} = \big(\mu_{nhanh}(x_1), \mu_{nhanh}(x_2), \cdots, \mu_{nhanh}(x_n)\big), \tag{1.70b}$$

$$\underline{\mu}^T_{tăng} = \big(\mu_{tăng}(y_1), \mu_{tăng}(y_2), \cdots, \mu_{tăng}(y_m)\big) \tag{1.70c}$$

và

$$\underline{\mu}^T_{giảm} = \big(\mu_{giảm}(y_1), \mu_{giảm}(y_2), \cdots, \mu_{giảm}(y_m)\big). \tag{1.70d}$$

Từ đây suy ra (xem công thức (1.63))

$$R_1 = \underline{\mu}_{chậm}\, \underline{\mu}^T_{tăng} = \begin{pmatrix} r^1_{11} & \cdots & r^1_{1m} \\ \vdots & & \vdots \\ r^1_{n1} & \cdots & r^1_{nm} \end{pmatrix}, \tag{1.71a}$$

$$R_2 = \underline{\mu}_{nhanh}\, \underline{\mu}^T_{giảm} = \begin{pmatrix} r^2_{11} & \cdots & r^2_{1m} \\ \vdots & & \vdots \\ r^2_{n1} & \cdots & r^2_{nm} \end{pmatrix} \tag{1.71b}$$

và do đó luật hợp thành sẽ có giá trị chung là

$$R = R_1 \cup R_2 = \begin{pmatrix} \max\{r^1_{11}, r^2_{11}\} & \cdots & \max\{r^1_{1m}, r^2_{1m}\} \\ \vdots & & \vdots \\ \max\{r^1_{n1}, r^2_{n1}\} & \cdots & \max\{r^1_{nm}, r^2_{nm}\} \end{pmatrix}. \tag{1.72}$$

Chú ý rằng việc thực hiện phép nhân *dyadic* (1.71) phụ thuộc vào quy tắc sử dụng khi mô hình hóa. Nếu dụng quy tắc max-MIN thì thay cho phép nhân phải dùng phép tính lấy cực tiểu min.

Thuật toán xây dựng luật hợp thành có nhiều mệnh đề hợp thành

Tổng quát hóa phương pháp mô hình hóa trên cho p mệnh đề hợp thành gồm

R_1: NẾU $\chi = A_1$ THÌ $\gamma = B_1$ hoặc (1.73a)

R_2: NẾU $\chi = A_2$ THÌ $\gamma = B_2$ hoặc (1.73b)

$\vdots$

R_p: NẾU $\chi = A_p$ THÌ $\gamma = B_p$ (1.73c)

trong đó các giá trị mờ $A_1, A_2, \cdots, A_p$ có cùng tập nền X và $B_1, B_2, \cdots, B_p$ có cùng tập nền Y.

Gọi hàm thuộc của A_k và B_k là $\mu_{A_k}(x)$ và $\mu_{B_k}(y)$ với k=1, 2, ⋯ , p. Thuật toán triển khai

$$R = R_1 \cup R_2 \cup \cdots \cup R_p$$

sẽ như sau:

1) Rời rạc hóa X tại n điểm $x_1, x_2, \cdots, x_n$ và Y tại m điểm $y_1, y_2, \cdots, y_m$,

2) Xác định các vector $\underline{\mu}_{A_k}$ và $\underline{\mu}_{B_k}$, k=1, 2, ⋯ , p theo

$$\underline{\mu}_{A_k}^T = \left(\mu_{A_k}(x_1), \mu_{A_k}(x_2), \cdots, \mu_{A_k}(x_n)\right)$$

$$\underline{\mu}_{B_k}^T = \left(\mu_{B_k}(y_1), \mu_{B_k}(y_2), \cdots, \mu_{B_k}(y_m)\right), \ldots$$

tức là Fuzzy hóa các điểm rời rạc của X và Y,

3) Xác định mô hình cho luật điều khiển

   a) $R_k = \underline{\mu}_{A_k} \cdot \underline{\mu}_{B_k}^T = \left(r_{ij}^k\right)$, i= 1, ⋯, n và j= 1, ⋯ , n.

   b) trong đó phép nhân được thay bằng phép tính lấy cực tiểu min khi sử dụng nguyên tắc max-MIN.

4) Xác định luật hợp thành $R = \max\{\, r_{ij}^k \mid k = 1, 2, \cdots, p \,\}$. (1.74)

Chú ý: từng mệnh đề nên được mô hình hóa thống nhất theo một quy tắc chung, ví dụ hoặc theo quy tắc max-MIN hoặc theo max-PROD ⋯. Khi đó các luật điều khiển R_k sẽ có một tên chung là luật hợp thành max-MIN hay luật hợp thành max-PROD. Tên chung này cũng sẽ là tên gọi của luật hợp thành R.

Ví dụ: Ví dụ sau mô tả việc triển khai R theo quy tắc max-MIN cho hai luật điều khiển là hai mệnh đề hợp thành (1.66). Các hàm thuộc tương ứng cho trong *hình 1.25*.

1) Rời rạc hóa X và Y:

$X = \{0{,}3 ; 0{,}4 ; 0{,}5 ; 0{,}6 ; 0{,}7 ; 0{,}8 ; 0{,}9 ; 1{,}0\}$,

$Y = \{0{,}6 ; 0{,}8 ; 1{,}0 ; 1{,}2 ; 1{,}4 ; 1{,}6 ; 1{,}8 ; 2{,}0 ; 2{,}2\}$,

2) Fuzzy hóa các điểm rời rạc trên thành 4 vector

$$\underline{\mu}_{chậm} = \begin{pmatrix} 0 \\ 0{,}5 \\ 1 \\ 0{,}5 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \ \underline{\mu}_{nhanh} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0{,}5 \\ 1 \\ 0{,}5 \\ 0 \end{pmatrix}, \ \underline{\mu}_{giảm} = \begin{pmatrix} 0 \\ 0{,}33 \\ 0{,}66 \\ 1 \\ 0{,}66 \\ 0{,}33 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \ \underline{\mu}_{tăng} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0{,}33 \\ 0{,}66 \\ 1 \\ 0{,}66 \\ 0{,}33 \\ 0 \end{pmatrix}.$$

3) Tính R_1 và R_2

$$R_1 = \underline{\mu}_{chậm} \cdot (\underline{\mu}_{tăng})^T = \begin{pmatrix} 0 \\ 0{,}5 \\ 1 \\ 0{,}5 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0{,}33 & 0{,}66 & 1 & 0{,}66 & 0{,}33 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0{,}33 & 0{,}5 & 0{,}5 & 0{,}5 & 0{,}33 & 0 \\ 0 & 0 & 0 & 0{,}33 & 0{,}66 & 1 & 0{,}66 & 0{,}33 & 0 \\ 0 & 0 & 0 & 0{,}33 & 0{,}5 & 0{,}5 & 0{,}5 & 0{,}33 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$R_2 = \mu_{nhanh} \mu^T_{giảm} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0,5 \\ 1 \\ 0,5 \\ 0 \end{pmatrix} \begin{pmatrix} 0 & 0,33 & 0,66 & 1 & 0,66 & 0,33 & 0 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0, & 0 \\ 0 & 0,33 & 0,5 & 0,5 & 0,5 & 0,33 & 0 & 0 & 0 \\ 0 & 0,33 & 0,66 & 1 & 0,66 & 0,33 & 0 & 0 & 0 \\ 0 & 0,33 & 0,5 & 0,5 & 0,5 & 0,33 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

4) Xác định $R = R_1 \cup R_2$

$$R = \max(R_1 . R_2) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,33 & 0,5 & 0,5 & 0,5 & 0,33 & 0 \\ 0 & 0 & 0 & 0,33 & 0,66 & 1 & 0,66 & 0,33 & 0 \\ 0 & 0 & 0 & 0,33 & 0,5 & 0,5 & 0,5 & 0,33 & 0 \\ 0 & 0,33 & 0,5 & 0,5 & 0,5 & 0,33 & 0 & 0 & 0 \\ 0 & 0,33 & 0,66 & 1 & 0,66 & 0,33 & 0 & 0 & 0 \\ 0 & 0,33 & 0,5 & 0,5 & 0,5 & 0,33 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Với một giá trị rõ x=0,6 đầu vào (phần tử thứ 4 trong dãy điểm rời rạc của X) thì đầu ra sẽ là một giá trị mờ B' có hàm thuộc sau (lấy từ hàng thứ 4 trong R)

$\mu_{B'}(y) = \{0 ; 0 ; 0 ; 0,33 ; 0,5 ; 0,5 ; 0,5 ; 0,33 ; 0\}$.

1.5.7 Thuật toán xác định luật hợp thành sum-MIN và sum-PROD

Chương mục trên đã mô tả phương pháp xây dựng luật hợp thành chung R cho một tập gồm nhiều mệnh đề hợp thành R_k được liên kết với nhau bằng toán tử HOẶC theo (1.30a) và do đó xuất hiện ký hiệu max trong tên gọi của luật hợp thành cũng như quy tắc được sử dụng như luật hợp thành max-MIN hay luật hợp thành max-PROD.

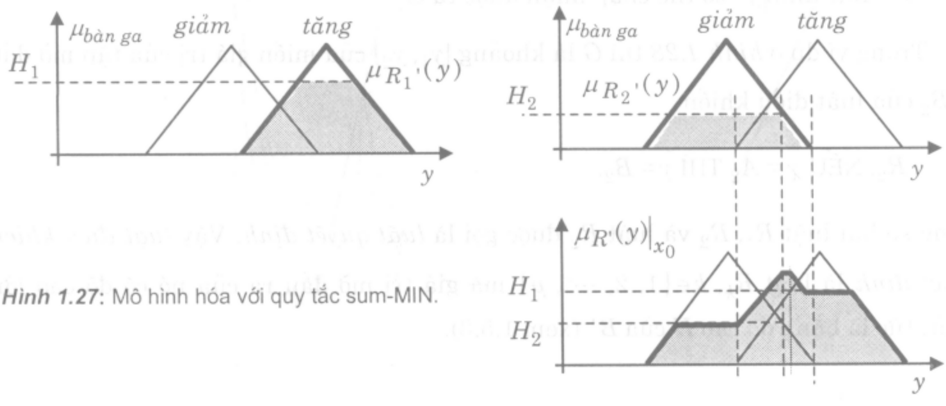
Kiểu liên kết nhiều mệnh đề hợp thành, hay còn gọi là luật điều khiển R_k, bằng toán tử HOẶC theo công thức (1.30a) không có tính thống kê. Chẳng hạn như khi đa số các mệnh đề hợp thành R_k có cùng một giá trị đầu ra nhưng vì không phải là giá trị lớn nhất nên sẽ không được để ý tới và bị mất trong kết quả chung.

Có nhiều cách khắc phục nhược điểm này. Một trong những phương pháp phổ biến là sử dụng phép hợp *Lukasiewicz* (1.30b) thay cho (1.30a) để liên kết các mệnh đề hợp thành R_k lại với nhau

$$R = \min\left\{1, \sum_{k=1}^{p} R_k\right\}, \tag{1.75}$$

trong đó phép lấy cực tiểu min được thực hiện giữa số 1 và từng phần tử của ma trận tổng.

Vì trong (1.75) R được xác định bằng cách cộng các R_k của các mệnh đề hợp thành nên luật hợp thành R theo liên kết Lukasiewicz (1.30b) sẽ có tên gọi là sum-MIN hoặc sum-PROD (sum là chữ viết tắt của từ tiếng Anh *sumc* – tổng) thay vì max-MIN hay max-PROD.



Hình 1.27: Mô hình hóa với quy tắc sum-MIN.

Thuật toán triển khai R theo quy tắc sum-MIN hay sum-PROD cũng bao gồm các bước như khi triển khai với quy tắc max-MIN hoặc max-PROD đã được trình bày trong mục 1.5.5, chỉ riêng tại bước 4 thì công thức (1.75) được sử dụng thay cho (1.74). *Hình 1.27* là một ví dụ về mô hình hóa R gồm hai mệnh đề hợp thành (1.66) theo quy tắc sum-MIN (so sánh với *hình 1.26* là kết quả khi sử dụng quy tắc max-MIN).

1.6 Giải mờ (rõ hoá)

Bộ điều khiển mờ tổng hợp theo kiểu *hình 1.20*, cho dù với một hoặc với nhiều luật điều khiển (mệnh đề hợp thành), cũng chưa thể áp dụng được trong điều khiển đối tượng, vì đầu ra luôn là một giá trị mờ B'. Một bộ điều khiển mờ hoàn chỉnh cần phải có thêm khâu giải mờ (quá trình rõ hóa tập mờ đầu ra B').

Giải mờ là quá trình xác định một giá trị rõ y' nào đó có thể chấp nhận được từ hàm thuộc $\mu_{B'}(y)$ của giá trị mờ B' (tập mờ). Có hai phương pháp giải mờ chính là *phương pháp cực đại* và *phương pháp điểm trọng tâm* sẽ được trình bày dưới đây, trong đó tập nền của tập mờ B' được ký hiệu thống nhất là Y.

1.6.1 Phương pháp cực đại

Theo tư tưởng cho rằng giá trị rõ y' đại diện cho tập mờ phải là giá trị có "xác suất" thuộc tập mờ lớn nhất, thì phương pháp cực đại để giải mờ sẽ gồm hai bước:

a) Xác định miền chứa giá trị rõ y'. Giá trị rõ y' là giá trị mà tại đó hàm thuộc đạt giá trị cực đại (độ cao H của tập mờ B'), tức là miền

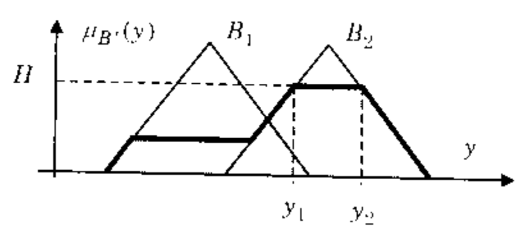
$$G = \{ y \in Y \mid \mu_{B'}(y) = H \}.$$

b) Xác định y' có thể chấp nhận được từ G.

Trong ví dụ ở *hình 1.28* thì G là khoảng $[y_1, y_2]$ của miền giá trị của tập mờ đầu ra B_2 của luật điều khiển

R_2: NẾU $\chi = A_2$ THÌ $\gamma = B_2$.

trong số hai luật R_1, R_2 và luật R_2 được gọi là *luật quyết định*. Vậy *luật điều khiển quyết định* là luật R_k, $k \in \{1, 2, \ldots, p\}$ mà giá trị mờ đầu ra của nó có độ cao lớn nhất, tức là bằng độ cao H của B' (xem 1.5.3).



Hình 1.28: Giải mờ bằng phương pháp cực đại.

Để thực hiện bước hai có ba nguyên lý:

- nguyên lý trung bình,
- nguyên lý cận trái và
- nguyên lý cận phải.

Nếu ký hiệu

$$y_1 = \inf_{y\in G}(y) \quad \text{và} \quad y_2 = \sup_{y\in G}(y). \tag{1.76}$$

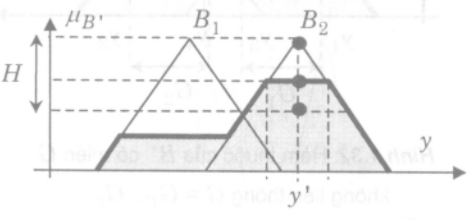
thì y_1 chính là điểm cận trái và y_2 là cận phải của G.

Nguyên lý trung bình

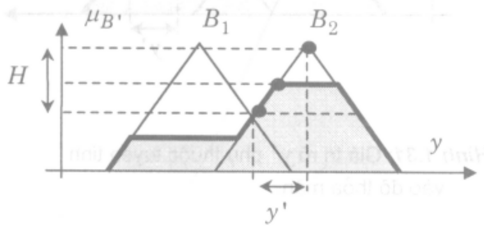
Theo nguyên lý trung bình, giá trị rõ y' sẽ là

$$y' = \frac{y_1 + y_2}{2}. \tag{1.77}$$

Nguyên lý này thường được dùng khi G là một miền liên thông và như vậy y' cũng sẽ là giá trị có độ phụ thuộc lớn nhất. Trong trường hợp B' gồm các hàm thuộc dạng đều thì giá trị rõ y' (1.77) không phụ thuộc vào độ thỏa mãn của luật điều khiển quyết định (xem *hình 1.29*).



Hình 1.29: Giá trị rõ y' không phụ thuộc vào đáp ứng vào của luật điều khiển quyết định.



Hình 1.30: Giá trị rõ y' phụ thuộc tuyến tính với đáp ứng vào của luật điều khiển quyết định.

Nguyên lý cận trái

Giá trị rõ y' được lấy bằng cận trái y_1 của G theo (1.76). Giá trị rõ lấy theo nguyên lý cận trái này sẽ phụ thuộc tuyến tính vào độ thỏa mãn của luật điều khiển quyết định.

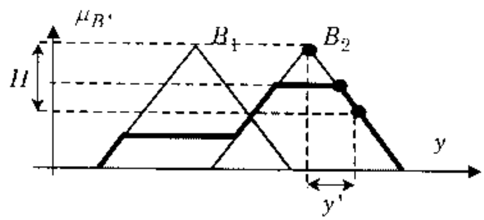
Nguyên lý cận phải

Giá trị rõ y' được lấy bằng cận phải y_2 của G theo (1.76). Cũng giống như nguyên lý cận trái, giá trị rõ y'ở đây phụ thuộc tuyến tính vào đáp ứng vào của luật điều khiển quyết định.

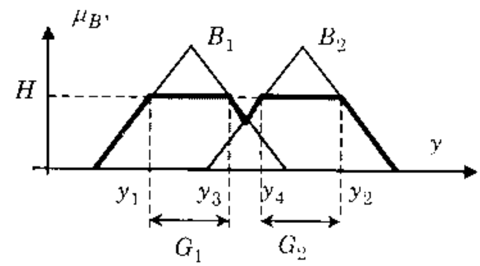
Ghi chú

Sai lệch của ba giá trị rõ, xác định theo trung bình, cận trái hay cận phải sẽ càng lớn nếu độ thỏa mãn H của luật điều khiển quyết định càng nhỏ.

Một câu hỏi đặt ra chung cho cả ba nguyên lý trên là y' sẽ được chọn như thế nào khi G không phải là một miền liên thông?, tức là khi có nhiều luật hợp thành có cùng một đáp ứng vào cho những giá trị quyết định khác nhau của biến ngôn ngữ đầu ra. Chẳng hạn nếu vẫn cứ áp dụng nguyên lý trung bình thì có thể giá trị rõ y' sẽ là giá trị có độ phụ thuộc nhỏ hơn H, hoặc nếu sử dụng nguyên lý cận trái hay phải thì các trường hợp còn lại là y_3 và y_1 thì sao? (*hình 1.32*).



Hình 1.31: Giá trị rõ y' phụ thuộc tuyến tính vào độ thỏa mãn.



Hình 1.32: Hàm thuộc của B' có miền G không liên thông $G = G_1 \cup G_2$.

Đối với những trường hợp như vậy, thông thường một khoảng con liên thông trong G sẽ được chọn làm *khoảng liên thông có mức ưu tiên cao nhất*,ví dụ là G_1, sau đó áp dụng một trong ba nguyên lý đã biết với miền G_1 thay cho G.

Trong các hình vẽ minh họa trên, từ *hình 1.28* đến *hình 1.32*, tập mờ B' thu được là do đã sử dụng quy tắc max-MIN. Đối với luật hợp thành max-PROD, miền G sẽ chỉ có một điểm duy nhất và do đó cả ba nguyên lý trung bình, cận trái, cận phải sẽ cho ra cùng một kết quả.

1.6.2 Phương pháp điểm trọng tâm

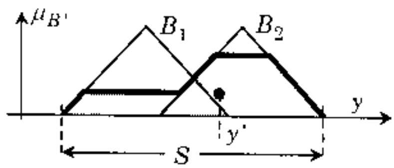
Phương pháp điểm trọng tâm sẽ cho ra kết quả y' là hoành độ của điểm trọng tâm miền được bao bởi trục hoành và đường $\mu_{B'}(y)$ - xem *hình 1.33*.

Công thức xác định y' theo phương pháp điểm trọng tâm như sau:

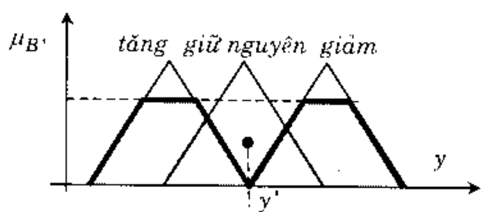
$$y' = \frac{\int_S y\mu_{B'}(y)dy}{\int_S \mu_{B'}(y)dy} \qquad (1.78)$$

trong đó S là miền xác định của tập mờ B'.

Công thức (1.78) cho phép xác định giá trị y' với sự tham gia của tất cả các tập mờ đầu ra của mọi luật điều khiển một cách bình đẳng và chính xác, tuy nhiên lại không để ý được tới độ thỏa mãn của luật điều khiển quyết định và thời gian tính (1.78) lâu. Ngoài ra một trong những nhược điểm cơ bản của phương pháp điểm trọng tâm là có thể giá trị y' xác định được lại có độ phụ thuộc nhỏ nhất, thậm chí bằng 0 (xem *hình 1.34* làm ví dụ). Bởi vậy để tránh những trường hợp như vậy, khi định nghĩa hàm thuộc cho từng giá trị mờ của một biến ngôn ngữ nên để ý sao cho miền xác định của các giá trị mờ đầu ra là một miền liên thông.



Hình 1.33: Giá trị rõ y' là hoành độ của điểm trọng tâm.



Hình 1.34: Xác định giá trị rõ y' theo phương pháp điểm trọng tâm khi miền giá trị của tập mờ không liên thông.

Phương pháp điểm trọng tâm cho luật hợp thành sum-MIN

Giả sử có q luật điều khiển được triển khai. Vậy thì mỗi một giá trị mờ B' tại đầu ra của bộ điều khiển sẽ là tổng của q giá trị mờ đầu ra của từng luật hợp thành. Ký hiệu các giá trị mờ đầu ra của luật điều khiển thứ k là $\mu_{B'_k}(y)$ với k=1, 2, ···, q thì với quy tắc sum-MIN, hàm thuộc $\mu_{B'}(y)$ sẽ là

$$\mu_{B'}(y) = \sum_{k=1}^{q} \mu_{B'_k}(y) . \quad (1.79)$$

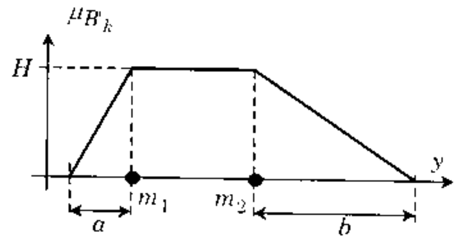
Thay (1.79) vào (1.78), sau đó đổi chỗ của tổng và tích phân cho nhau (hoàn toàn có nghĩa, vì tổng và tích phân đều hội tụ) thì công thức tính y' sẽ được đơn giản như sau

$$y' = \frac{\int_S \left(y \sum_{k=1}^{q} \mu_{B'_k}(y) \right) dy}{\int_S \sum_{k=1}^{q} \mu_{B'_k}(y) dy} = \frac{\sum_{k=1}^{q} \left(\int_S y \mu_{B'_k}(y) dy \right)}{\sum_{k=1}^{q} \left(\int_S \mu_{B'_k}(y) dy \right)} = \frac{\sum_{k=1}^{q} M_k}{\sum_{k=1}^{q} A_k} . \quad (1.80)$$

trong đó

$$M_k = \int_S y \mu_{B'_k}(y) dy \quad \text{và} \quad A_k = \int_S \mu_{B'_k}(y) dy . \quad (1.81)$$

Hình 1.35: Tập mờ có hàm thuộc hình thang.



Xét riêng cho các hàm thuộc $\mu_{B'_k}(y)$ dạng hình thang như trong *hình 1.35* thì

$$M_k = \frac{H}{6}\left(3m_2^2 - 3m_1^2 + b^2 - a^2 + 3m_2 b + 3m_1 a\right), \quad (1.82a)$$

$$A_k = \frac{H}{2}\left(2m_2 - 2m_1 + a + b\right). \quad (1.82b)$$

Công thức (1.82) rất tiện lợi để tính nhanh y'. Chẳng hạn để tính giá trị rõ y' cho tập mờ đầu ra B' gồm hai luật điều khiển R_1 và R_2 với

$$\mu_{B'}(y) = \mu_{B'_1}(y) + \mu_{B'_2}(y)$$

cho trong *hình 1.36* theo (1.81) thì khi áp dụng (1.82), có ngay được

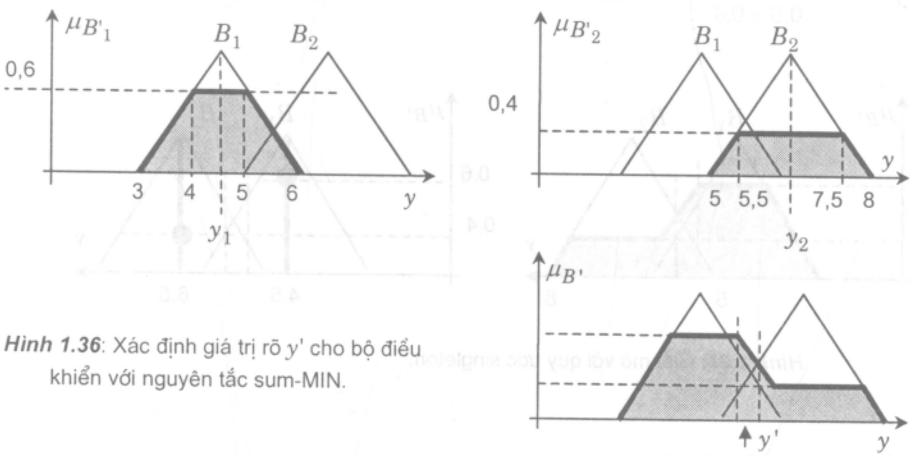
$M_1 = 5{,}4$ và $A_1 = 1{,}2$

$M_2 = 6{,}5$ và $A_2 = 1$.

Suy ra

$$y' = \frac{5{,}4 + 6{,}5}{1{,}2 + 1} = 5{,}41.$$

Chú ý: Mặc dù công thức (1.80) chỉ được xây dựng cho luật hợp thành kiểu sum-MIN, song trong thực tế nó vẫn được dùng cho cả luật hợp thành max-MIN.



Hình 1.36: Xác định giá trị rõ y' cho bộ điều khiển với nguyên tắc sum-MIN.

Phương pháp độ cao

Sử dụng công thức (1.80) cho cả hai loại luật hợp thành max-MIN và sum-MIN với thêm một giả thiết là mỗi tập mờ $\mu_{B'_k}(y)$ được xấp xỉ bằng một cặp giá trị (y_k, H_k) duy nhất (*singleton*), trong đó H_k là độ cao của $\mu_{B'_k}(y)$ và y_k là một điểm mẫu trong miền giá trị của $\mu_{B'_k}(y)$ có

$$\mu_{B'_k}(y_k) = H_k$$

thì

$$y' = \frac{\sum_{k=1}^{q} y_k H_k}{\sum_{k=1}^{q} H_k}. \tag{1.83}$$

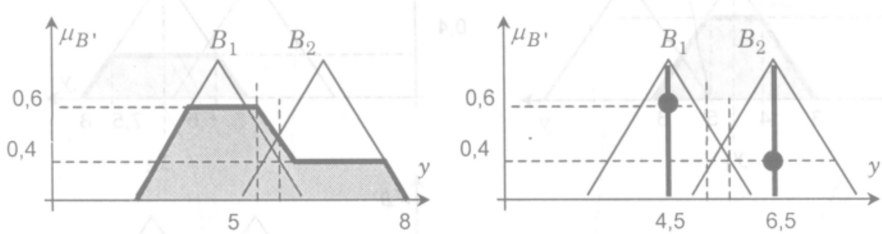
Công thức (1.83) có tên gọi là công thức tính xấp xỉ y' theo *phương pháp độ cao* và không chỉ áp dụng cho luật hợp thành max-MIN, sum-MIN mà còn có thể cho cả những luật hợp thành khác như max-PROD hay sum-PROD.

Áp dụng (1.83) cho tập mờ B' trong *hình 1.36* được

- B'_1 được xấp xỉ bằng singleton (4,5 ; 0,6) và
- B'_2 được xấp xỉ bằng singleton (6,5 ; 0,4).

Suy ra

$$y' = \frac{4{,}5 \cdot 0{,}6 + 6{,}5 \cdot 0{,}4}{0{,}6 + 0{,}4} = 5{,}3.$$



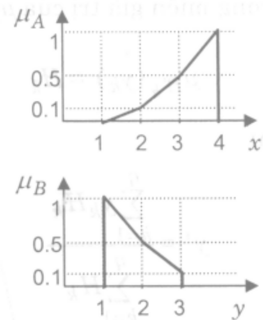
Hình 1.37: Giải mờ với quy ước singleton.

Câu hỏi ôn tập và bài tập

1) Hãy chứng minh các công thức giải mờ (1.82).

2) Hãy chỉ ra rằng hàm thuộc của giá trị mờ đầu ra của một bộ điều khiển mờ bất kỳ sử dụng công thức max-PROD để cài đặt luật hợp thành đều nhỏ hơn hàm thuộc của giá trị mờ đầu ra của cùng bộ điều khiển đó nhưng luật hợp thành được cài đặt theo nguyên tắc max-MIN.

3) Cho hai tập nền $U=\{1,2,3,4\}$ và $V=\{1,2,3\}$ với hai biến ngôn ngữ χ, ψ. Hãy xác định tập mờ B' đầu ra của mệnh đề hợp thành NẾU $\chi = A$ THÌ $\psi = B$, khi đầu vào là một giá trị rõ $x_0=2{,}5 \in U$, trong đó các giá trị ngôn ngữ A, B là hai tập mờ với hai hàm thuộc $\mu_A(x)$, $x \in U$ và $\mu_B(y)$, $y \in V$ cho trong hình bên.



4) Cho bộ điều khiển mờ với luật hợp thành.

R_1: Nếu $x = A_1$ và $y = B_1$ thì $z = C_1$

R_2: Nếu $x = A_2$ và $y = B_2$ thì $z = C_2$

R_3: Nếu $x = A_3$ và $y = B_3$ thì $z = C_3$.

trong đó các tập mờ A_1, A_2, A_3, B_1, B_2, B_3, C_1, C_2, C_3, được định nghĩa như sau:

$$\mu_{A_1}(x) = \begin{cases} \dfrac{3+x}{3} & \text{khi } -3 \le x \le 0 \\ 1 & \text{khi } 0 \le x \le 3 \\ \dfrac{6-x}{3} & \text{khi } 3 < x < 6 \end{cases} \qquad \mu_{A_2}(x) = \begin{cases} \dfrac{x-2}{3} & \text{khi } 2 \le x \le 5 \\ \dfrac{6-x}{4} & \text{khi } 5 \le x < 9 \end{cases}$$

$$\mu_{A_3}(x) = \begin{cases} \dfrac{x-6}{4} & \text{khi } 6 \le x \le 10 \\ \dfrac{13-x}{3} & \text{khi } 10 \le x \le 13 \end{cases} \qquad \mu_{B_1}(y) = \begin{cases} \dfrac{y-1}{4} & \text{khi } 1 < y \le 5 \\ \dfrac{7-y}{2} & \text{khi } 5 \le y \le 7 \end{cases}$$

$$\mu_{B_2}(y) = \begin{cases} \dfrac{y-5}{3} & \text{khi } 5 \le y \le 8 \\ \dfrac{12-y}{4} & \text{khi } 8 \le y \le 12 \end{cases} \qquad \mu_{B_3}(y) = \begin{cases} \dfrac{y-8}{4} & \text{khi } 8 \le y \le 12 \\ \dfrac{15-y}{3} & \text{khi } 12 \le y \le 15 \end{cases}$$

$$\mu_{C_1}(z) = \begin{cases} \dfrac{3-z}{3} & \text{khi } -3 \le z \le -1 \\ 1 & \text{khi } -1 \le z \le 1 \\ \dfrac{3-z}{2} & \text{khi } 1 \le z \le 3 \end{cases} \qquad \mu_{C_2}(z) = \begin{cases} \dfrac{z-1}{3} & \text{khi } 1 \le z \le 4 \\ \dfrac{7-z}{3} & \text{khi } 4 \le z \le 7 \end{cases}$$

$$\mu_{C_3}(z) = \begin{cases} \dfrac{z-5}{3} & \text{khi } 5 \le z \le 7 \\ 1 & \text{khi } 7 \le z \le 9 \\ \dfrac{3-z}{2} & \text{khi } 9 < z \le 11 \end{cases}$$

Với hai giá trị rõ đầu vào x_0=3 và y_0=6 hãy:

a) Xác định giá trị mờ R' của đầu ra của bộ điều khiển nếu luật hợp thành được cài đặt theo nguyên tắc max-MIN.

b) Vẽ hàm thuộc $\mu_{R'}(z)$ của R'.

c) Xác định giá trị rõ z_0 đầu ra nếu bộ điều khiển được giải mờ theo phương pháp điểm trọng tâm.

5) Hãy xác định luật hợp thành cho bộ điều khiển mờ để điều khiển nhiệt độ của một lò hơi với ba biến ngôn ngữ: nhiệt độ (biến đầu vào), van khí đốt (biến đầu ra 1) và van không khí (biến đầu ra 2) sao cho nhiệt độ lò hơi được duy trì trong lân cận 540°C. Hai van đều mở ở mức giữa (50%), vùng nhiệt độ cần khống chế là từ 440°C đến 640°C.

2 TÍNH PHI TUYẾN CỦA HỆ MỜ

Kỹ thuật điều khiển mờ, kể từ khi ra đời bằng một mô hình điều khiển máy hơi nước của Mamdani vào giữa thập kỷ 70 cho tới nay, đã được biết đến như là phần bổ sung lý tưởng cho kỹ thuật điều khiển kinh điển trong lĩnh vực phi tuyến, khi mà nền tảng lý thuyết phi tuyến xây dựng trên sự phát triển lý thuyết tuyến tính kinh điển chỉ phần nào ứng dụng được trong một phạm vi hẹp cho những hệ có những tính chất phi tuyến đặc thù như hệ lưỡng tuyến tính (*bilinear*), hệ autonom, hệ có tham số biến đổi chậm ....

Trong phần này sẽ trình bầy những kiến thức cơ bản về tính phi tuyến của hệ điều khiển mờ, bắt đầu bằng việc phân tích sự phụ thuộc của tính phi tuyến một khâu mờ vào tập các giá trị ngôn ngữ vào/ra, các khâu điều khiển mờ điển hình ... cho đến việc tổng hợp một khâu điều khiển mờ mong muốn.

2.1 Phân loại các khâu điều khiển mờ

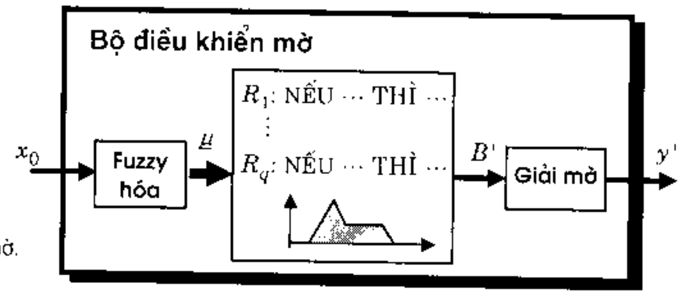
Một bộ điều khiển mờ, như đã được mô tả trong chương 1, có ba khâu cơ bản gồm (*hình 2.1*):

- khâu Fuzzy hóa có nhiệm vụ chuyển đổi một giá trị rõ đầu vào x_0 thành một vector $\underline{\mu}$ gồm các độ phụ thuộc của giá trị rõ đó theo các giá trị mờ (tập mờ) đã định nghĩa cho biến ngôn ngữ đầu vào,
- khâu thực hiện luật hợp thành, có tên gọi là *thiết bị hợp thành*, xử lý vector $\underline{\mu}$ và cho ra giá trị mờ B' của biến ngôn ngữ đầu ra,
- khâu giải mờ, có nhiệm vụ chuyển đổi tập mờ B' thành một giá trị rõ y' chấp nhận được cho đối tượng (*tín hiệu điều chỉnh*).

Các bộ điều khiển mờ sẽ được phân loại dựa trên quan hệ vào/ra toàn cục của tín hiệu vào x_0 và tín hiệu ra y' biểu diễn ánh xạ $x_0 \mapsto y'$. Quan hệ toàn cục đó có tên gọi là *quan hệ truyền đạt*.

Cũng giống như việc phân loại một khâu điều khiển kinh điển, việc phân loại quan hệ truyền đạt của một bộ điều khiển mờ dựa vào 7 tiêu chuẩn:

a) tĩnh hay động,
b) tuyến tính hay phi tuyến,
c) tham số tập trung hay tham số rải,
d) liên tục hay rời rạc,
e) tham số tĩnh hay tham số động,
f) tiền định hay ngẫu nhiên,
g) ổn định hay không ổn định.



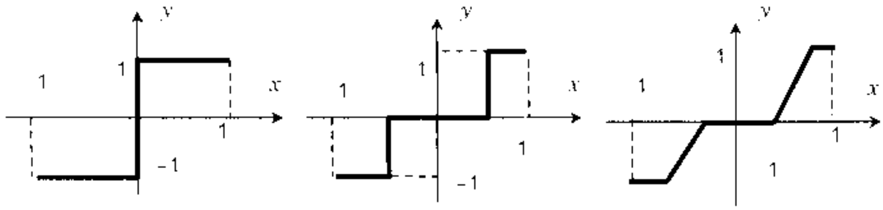
Hình 2.1: Cấu trúc bên trong của một bộ điều khiển mờ.

Nếu nhìn lại từng khâu của bộ điều khiển mờ gồm các khâu Fuzzy hóa, thiết bị hợp thành và giải mờ trong *hình 2.1*, thì thấy rằng trong quan hệ vào/ra giá trị y' tại đầu ra chỉ phụ thuộc vào một mình giá trị x_0 của đầu vào chứ không phụ thuộc vào các giá trị đã qua của tín hiệu $x(t)$, tức là chỉ phụ thuộc vào giá trị của $x(t)$ tại đúng thời điểm đó. Do đó bộ điều khiển mờ thực chất là một *bộ điều khiển tĩnh* và quan hệ truyền đạt hoàn toàn được mô tả đầy đủ bằng đường đặc tính $y(x)$ như các đường đặc tính của khâu relay 2 hoặc 3 trạng thái quen biết trong kỹ thuật điều khiển phi tuyến kinh điển (*hình 2.2*).

Tuy nhiên, trong một số trường hợp đặc biệt, qua thay đổi dạng hàm thuộc của các giá trị ngôn ngữ vào ra, hoặc nhờ việc nối thêm các khâu tích, vi phân vào phía trước bộ điều khiển làm vai trò tiền xử lý tín hiệu, thì bộ điều khiển chung nhận được sẽ lại có tính gần tĩnh (*quasi static*) giống như khâu relay có trễ hoặc có tính động như bộ điều khiển mờ PID sẽ trình bày sau trong chương 3.

Bộ điều khiển mờ với quan hệ truyền đạt $y(x)$ được gọi là phi tuyến (tĩnh), nếu $y(x)$ là một hàm phi tuyến. Dưới đây, tính phi tuyến của quan hệ truyền đạt, phụ thuộc vào tập các giá trị mờ của biến ngôn ngữ vào/ra, sẽ được trình bày thành ba phần:

- thay đổi tính phi tuyến bằng tập các giá trị mờ đầu vào,
- thay đổi tính phi tuyến bằng tập các giá trị mờ đầu ra và
- tạo bộ điều khiển mờ giả tĩnh, hay bộ điều khiển mờ có trễ.



Hình 2.2: Đường đặc tính của các khâu relay kinh điển.

2.1.1 Quan hệ truyền đạt và các tập mờ của biến ngôn ngữ đầu vào

Cũng như đã làm trong chương 1, sự phụ thuộc của quan hệ truyền đạt vào tập các hàm thuộc biến ngôn ngữ đầu vào, tức là miền xác định biến ngôn ngữ đầu vào, được giới thiệu nhập đề trên cơ sở một ví dụ. Các kết quả của ví dụ sau đó sẽ được tổng quát hóa.

Xét một ví dụ với biến ngôn ngữ đầu vào χ và đầu ra γ cùng chỉ một số thực có các giá trị mờ như sau:

- số thực xấp xỉ −1, gọi là số *âm*,
- số thực gần bằng 0, gọi là số *không*,
- số thực xấp xỉ 1, gọi là số *dương*,

trong đó tập các hàm thuộc đầu ra $\mu_{âm}(y)$, $\mu_{không}(y)$, $\mu_{dương}(y)$, là cố định và cho trong *hình 2.4*. Tính phi tuyến của $y(x)$ sẽ được xét cho ba trường hợp khác nhau về dạng miền xác định của các hàm thuộc $\mu_{âm}(x)$, $\mu_{không}(x)$, $\mu_{dương}(x)$ cho trong *hình 2.3*.

Luật hợp thành của bộ điều khiển mờ là luật max–MIN, khâu giải mờ được chọn làm việc theo phương pháp điểm trọng tâm.

Luật hợp thành R của bộ điều khiển gồm ba mệnh đề hợp thành (hay còn gọi *luật điều khiển*) như sau:

R_1: NẾU χ = âm THÌ γ = âm HOẶC

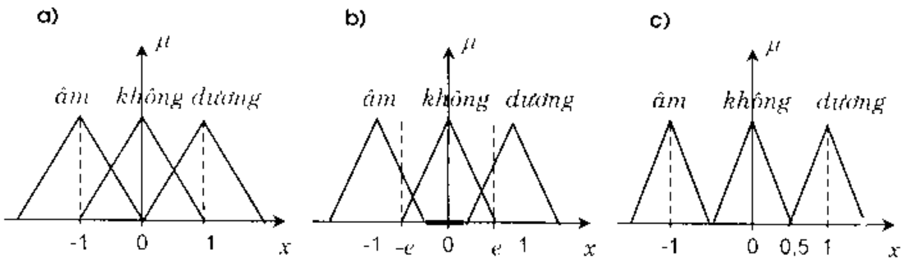
R_2: NẾU χ = không THÌ γ = không HOẶC

R_3: NẾU χ = dương THÌ γ = dương.

Về hình thức, R có dạng

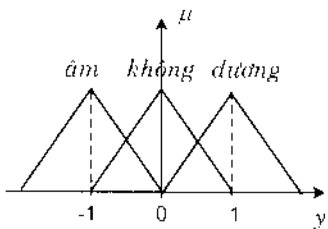
$$R = R_1 \cup R_2 \cup R_3$$

là một luật tỷ lệ thuận. Giá trị đầu vào χ càng lớn thì giá trị đầu ra γ càng lớn. Việc mô hình hóa R được tiến hành với thuật toán 4 bước đã trình bày trong mục 1.5.5 của chương 1.



Hình 2.3: Ba trường hợp khác nhau về các tập mờ của biến đầu vào χ.
a) Trường hợp 1 b) Trường hợp 2 c) Trường hợp 3

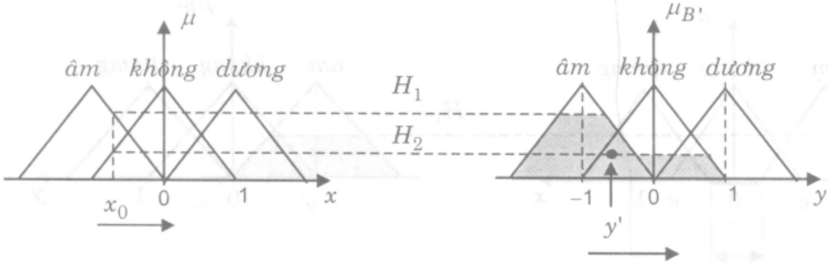
Hình 2.4. Miền xác định của các tập mờ thuộc biến đầu ra γ



Trường hợp 1

Với một giá trị rõ x_0 trong khoảng [-1,1] của đầu vào luôn có 2 trong 3 mệnh đề hợp thành tích cực, tức là có độ thỏa mãn lớn hơn 0. Giả sử hai mệnh đề đó là R_1 và R_2. Ký hiệu H_1 là độ thoả mãn của R_1 và H_2 là độ thỏa mãn của R_2. Cho x_0 tăng dần, H_1 sẽ giảm dần và H_2 tăng dần làm cho điểm trọng tâm của B' cũng dần chuyển dịch một cách tỷ lệ sang phải (*hình 2.5*).

Cứ tiếp tục tăng x_0 cho tới khi H_1=0, tức là R_1 trở thành không tích cực thì cũng tại thời điểm đó R_3 bắt đầu tích cực và H_3, độ thỏa mãn của R_3 cũng tăng dần lên theo. Điểm trọng tâm của B' vì thế vẫn tiếp tục dịch chuyển sang phải.

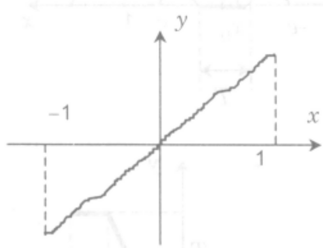


Hình 2.5: Tăng giá trị rõ đầu vào làm tăng giá trị rõ đầu ra.

Hình 2.6 biểu diễn đầu ra y' theo đầu vào x_0 và đó cũng là đường đặc tính $y(x)$ của quan hệ truyền đạt. Bỏ qua sự lượn sóng "không đáng kể" trong hình thì $y(x)$ có thể được xem như là một hàm tuyến tính và bộ điều khiển mờ với miền xác định trong *hình 2.3a* là một bộ điều khiển mờ "tuyến tính".

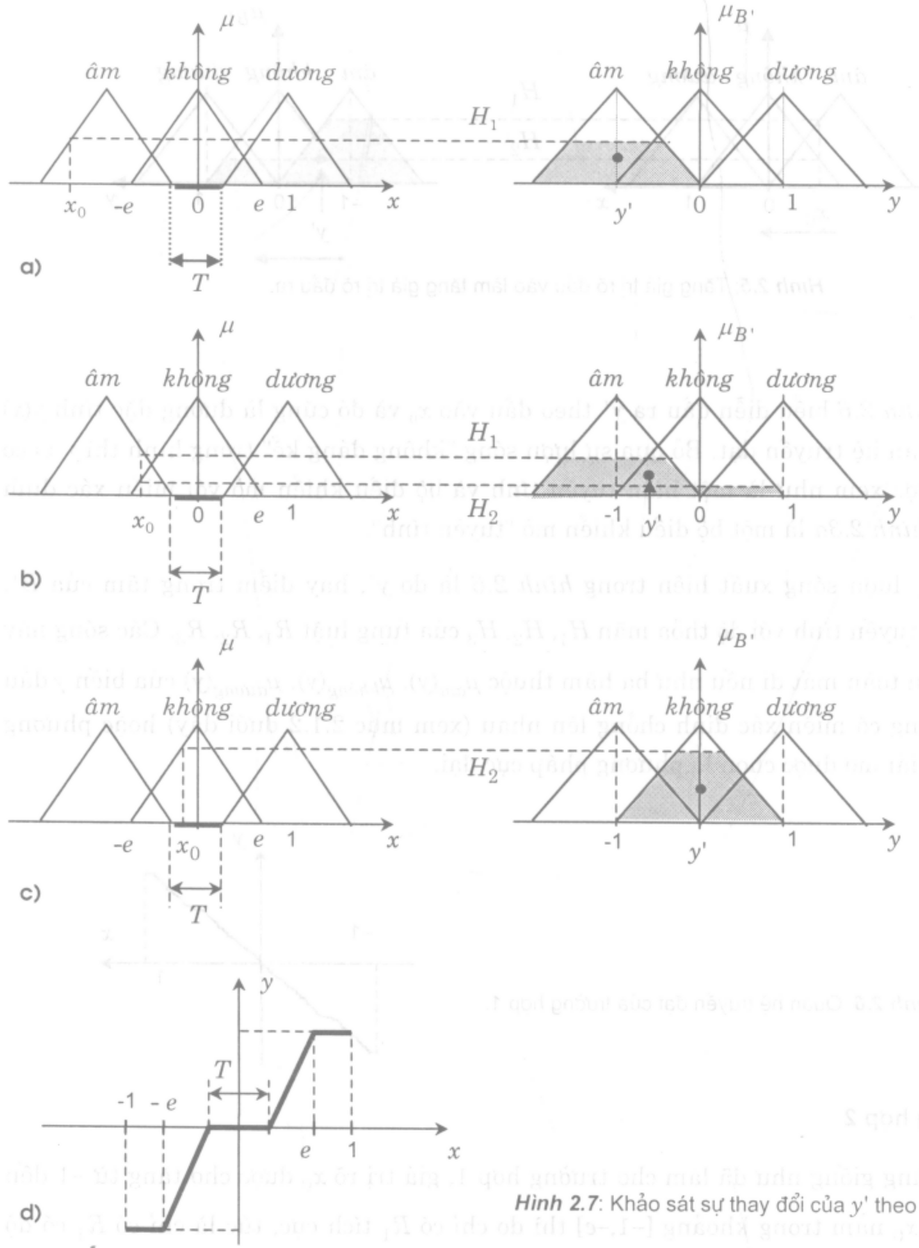
Sự lượn sóng xuất hiện trong *hình 2.6* là do y', hay điểm trọng tâm của B', không tuyến tính với độ thỏa mãn H_1, H_2, H_3 của từng luật R_1, R_2, R_3. Các sóng này sẽ hoàn toàn mất đi nếu như ba hàm thuộc $\mu_{âm}(y)$, $\mu_{không}(y)$, $\mu_{dương}(y)$ của biến y đầu ra không có miền xác định chồng lên nhau (xem mục 2.1.2 dưới đây) hoặc phương pháp giải mờ được chọn là phương pháp cực đại.

Hình 2.6: Quan hệ truyền đạt của trường hợp 1.



Trường hợp 2

Cũng giống như đã làm cho trường hợp 1, giá trị rõ x_0 được cho tăng từ –1 đến 1. Khi x_0 nằm trong khoảng [–1,–e] thì do chỉ có R_1 tích cực, tức là chỉ có R_1 có độ thỏa mãn H_1 lớn hơn 0, nên sẽ không phụ thuộc vào x_0, điểm trọng tâm của B' luôn nằm trên trục cố định là đường cao tam giác $\mu_{âm}(y)$ và do đó y' có giá trị bằng −1 (*hình 2.7a*).



Hình 2.7: Khảo sát sự thay đổi của y' theo x_0.

Cho x_0 tiếp tục tăng dần từ $-e$ đến điểm biên phía trái của miền T (*hình 2.7b*), lúc đó ngoài R_1 còn có R_2 tích cực nên điểm trọng tâm của B' sẽ dịch chuyển dần theo sang phải phụ thuộc vào các độ thỏa mãn H_1 và H_2 của R_1, R_2 (H_2 càng tăng thì điểm trọng tâm của B' càng dịch về phía $\mu_{không}(y)$, tức là càng dịch sang phải). Lúc này giá trị y', giống như trong trường hợp 1, thay đổi "tuyến tính" theo x_0.

Khi x_0 đạt tới giá trị của điểm biên trái miền T thì $y'=0$.

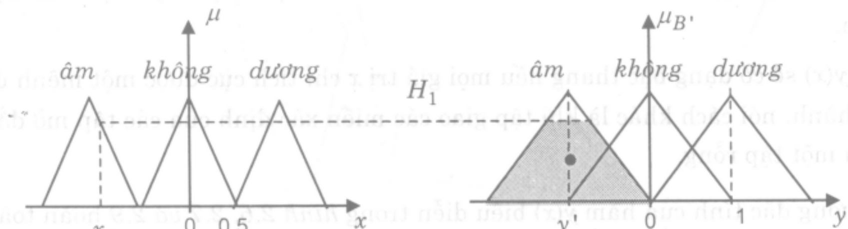
Khi x_0 thuộc miền T, do chỉ có một mình R_2 tích cực nên điểm trọng tâm của B' sẽ chỉ dịch chuyển trên đường cao tam giác $\mu_{không}(y)$ và do đó giá trị rõ y' bằng 0 (*hình 2.7c*).

Nếu x_0 tiếp tục tăng dần từ điểm biên phải của T tới e thì y' cũng tăng tuyến tính theo x_0 vì lúc này cả R_2 và R_3 đều tích cực.

Với $x_0 > e$, do chỉ có một mình R_3 tích cực nên điểm trọng tâm của B' lại nằm trên một trục cố định và y' có giá trị bằng 1.

Đường đặc tính $y(x)$ của trường hợp 2 được mô tả trong *hình 2.7d*.

Trường hợp 3



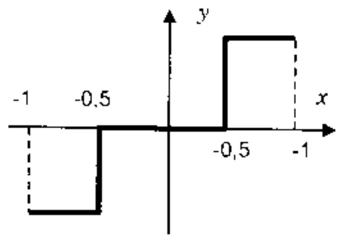
Hình 2.8: Khảo sát sự thay đổi của y' theo x_0 cho trường hợp 3.

Trong trường hợp này, vì luôn chỉ có một trong số 3 luật R_1, R_2, và R_3 là tích cực (*hình 2.8*) nên tập mờ B' bao giờ cũng chỉ thuộc một trong 3 tam giác $\mu_{âm}(y)$, $\mu_{không}(y)$, $\mu_{dương}(y)$. Do đó y' sẽ có giá trị

$$y' = \begin{cases} -1 & \text{nếu} \quad x_0 \le -0{,}5 \\ 0 & \text{nếu} \quad |x_0| < 0{,}5 \\ 1 & \text{nếu} \quad x_0 \ge 0{,}5 \end{cases} \qquad (2.1)$$

Hình 2.9 biểu diễn đường đặc tính $y(x)$ cho trường hợp 3.

Hình 2.9: Đường đặc tính của quan hệ truyền đạt trong trường hợp 3.



Ví dụ trên đây cho thấy tính phi tuyến của quan hệ truyền đạt $y(x)$ phụ thuộc vào sự phân bố miền xác định của tập các giá trị mờ đầu vào. Tổng kết lại cả ba trường hợp trên đưa tới:

- Hàm $y(x)$ sẽ trơn, nếu mọi điểm x thuộc miền xác định của *ít nhất hai tập mờ đầu vào*, tức là một giá trị x bất kỳ sẽ làm tích cực được ít nhất hai mệnh đề hợp thành.
- Hàm $y(x)$ sẽ nhận giá trị hằng không đổi trong khoảng x chỉ thuộc miền xác định của *một* tập mờ đầu vào, tức là khi x chỉ tích cực được một mệnh đề hợp thành.
- Hàm $y(x)$ sẽ có dạng bậc thang nếu mọi giá trị x chỉ tích cực được một mệnh đề hợp thành, nói cách khác là khi tập giao các miền xác định của các tập mờ đầu vào là một tập rỗng.

Các đường đặc tính của hàm $y(x)$ biểu diễn trong *hình 2.6, 2.7 và 2.9* hoàn toàn định tính. Hàm $y(x)$ cũng có thể được xác định một cách chính xác nhờ vào công thức (1.48) qua các bước:

1) Xác định độ thỏa mãn H_q của q mệnh đề hợp thành từ giá trị x bằng cách Fuzzy hóa x,
2) Tính các tham số α, β, m_1 và m_2 của từng tập mờ B'_k theo H_q.
3) Tính A_k và M_k theo (1.50).
4) Tính y' từ A_k và M_k theo (1.48).

2.1.2 Quan hệ truyền đạt và các tập mờ của biến ngôn ngữ đầu ra

Lý do xuất hiện các "sóng nhỏ" trong đường đặc tính $y(x)$ của trường hợp 1 (*hình 2.6*) là sự phụ thuộc không hoàn toàn tuyến tính của y' vào độ thỏa mãn H_k, $k=1, 2, 3$ của các mệnh đề hợp thành R_1, R_2 và R_3. Có nhiều cách để có thể thu được một đường đặc tính $y(x)$ thực sự tuyến tính như:

- giải mờ theo phương pháp cực đại với nguyên lý cận trái hoặc cận phải,
- sử dụng quy ước *singleton* trong giải mờ hoặc trong định nghĩa tập mờ cho biến ngôn ngữ đầu ra y (xem công thức (1.51) và *hình 1.37* thuộc chương 1).

Hãy xét kỹ hơn về trường hợp sau. Xuất phát từ

$$y' = \frac{(-1)\cdot H_1 + 0\cdot H_2 + 1\cdot H_3}{H_1 + H_2 + H_3} = \frac{H_3 - H_1}{H_1 + H_2 + H_3}\,, \tag{2.2}$$

trong đó H_1, H_2 và H_3 là những độ thỏa mãn của R_1, R_2, R_3 cho một giá trị rõ x tại đầu vào. Do luôn chỉ có hai trong số ba mệnh đề R_1, R_2, R_3 được tích cực (độ thỏa mãn khác 0) nên

$$H_1 + H_2 + H_3 = 1 \text{ (hằng số)}. \tag{2.3}$$

Ngoài ra

$H_1 = 0$ khi $x \geq 0$

và $H_3 = 0$ khi $x \leq 0$,

do đó

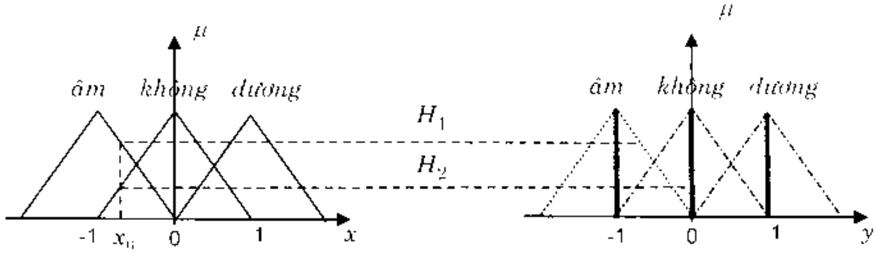
$$y' = \begin{cases} -H_1 & \text{khi } x \geq 0 \\ H_3 & \text{khi } x \leq 0 \end{cases}, \tag{2.4}$$

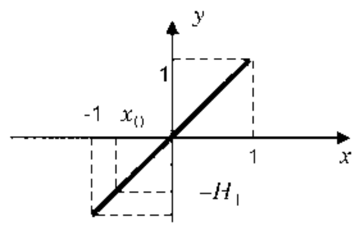
và như vậy, đường đặc tính của quan hệ truyền đạt là đường thẳng $y=x$ (xem hình *2.10*) vì với các tập mờ đầu vào $\mu_{\hat{a}m}(x)$, $\mu_{kh\hat{o}ng}(x)$, $\mu_{d\text{ư}\sigma ng}(x)$ như đã định nghĩa (*hình 2.10*) thì độ thỏa mãn H_1 và H_3 phụ thuộc x theo

$$H_1 = \begin{cases} -x & \text{khi } -1 \leq x \leq 0 \\ 0 & \text{khi } x > 0 \end{cases}$$

$$H_3 = \begin{cases} x & \text{khi } 0 \leq x \leq 1 \\ 0 & \text{khi } x < 0 \end{cases}.$$

Với phương pháp singleton trên đây, các hàm thuộc đầu ra dạng tam giác cân như *hình 2.4* về bản chất đã được thay bởi các hàm Kronecker. Việc thay đổi các tập mờ đầu ra thành hàm Kronecker như vậy đã làm mất đi các "lượn sóng" trong đường đặc tính $y(x)$ của quan hệ truyền đạt và do đó đã biến đổi tính gần tuyến tính của $y(x)$ thành thực sự tuyến tính.





Hình 2.10: Xác định y' theo quy ước singleton.

Vậy thì tính phi tuyến của $y(x)$ phụ thuộc như thế nào vào tập các tập mờ đầu ra?. Có khả năng thu được relay 2 hoặc 3 vị trí bằng cách thay đổi tập mờ đầu ra giống như khi thay đổi tập mờ đầu vào hay không?. Hãy xét 2 trường hợp:

– giữ nguyên dạng các tập mờ đầu ra, chỉ thay đổi phần miền xác định chồng lên nhau của chúng bằng cách tịnh tiến hàm thuộc dọc theo trục y,
– thay đổi dạng (độ rộng) các tập mờ đầu ra.

Tịnh tiến hàm thuộc đầu ra dọc theo trục *y*

Giả sử $-1 \le x \le 0$. Đi từ công thức (1.48) để tính y' thì sau khi tịnh tiến $\mu_{âm}(y)$ sang trái và $\mu_{dương}(y)$ sang phải một khoảng là a (*hình 2.11*), giá trị y' từ công thức (1.48) sẽ thay đổi một sai lệch Δ như sau:

$$\Delta = \frac{H_1(3H_2 - 2)a}{A_1 + A_2} = H_1(3H_2 - 2)b\ . \qquad (2.5a)$$

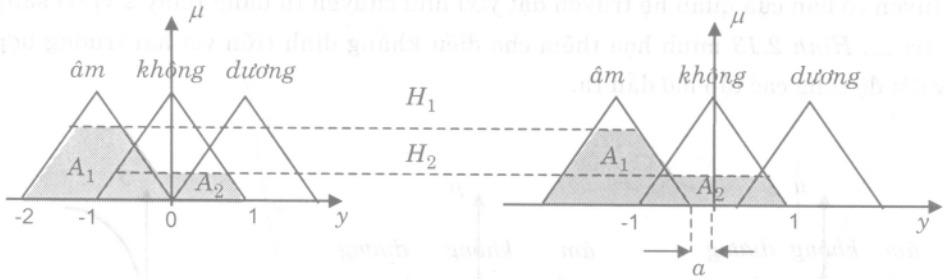
trong đó

$$b = \frac{a}{A_1 + A_2} = \frac{a}{(2-H_1)H_1 + (2-H_2)H_2}.$$

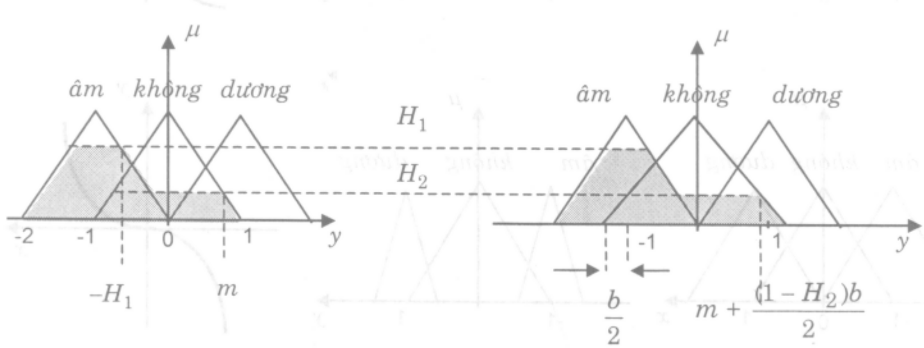
Tương tự, khi $0 \le x \le 1$ thì

$$\Delta = H_2(1+H_2)b\ . \tag{2.5b}$$

Cả hai công thức (2.5a) và (2.5b) cho thấy sai lệch Δ phụ thuộc tuyến tính theo a. Do đó việc tịnh tiến các hàm thuộc đầu ra nhằm thay đổi phần miền xác định chung của chúng không làm thay đổi được đặc tính cơ bản của quan hệ truyền đạt $y(x)$ như từ tuyến tính sang phi tuyến hoặc từ dạng relay 2 vị trí sang dạng relay 3 vị trí. Sự thay đổi duy nhất ở đây tạo ra được chỉ là tăng hoặc giảm những "sóng lượn" của đường đặc tính $y(x)$.



Hình 2.11: Tính sai lệch Δ do tịnh tiến $\mu_{âm}(y)$.



Hình 2.12: Tính sai lệch Δ do thay đổi độ rộng của $\mu_{không}(y)$.

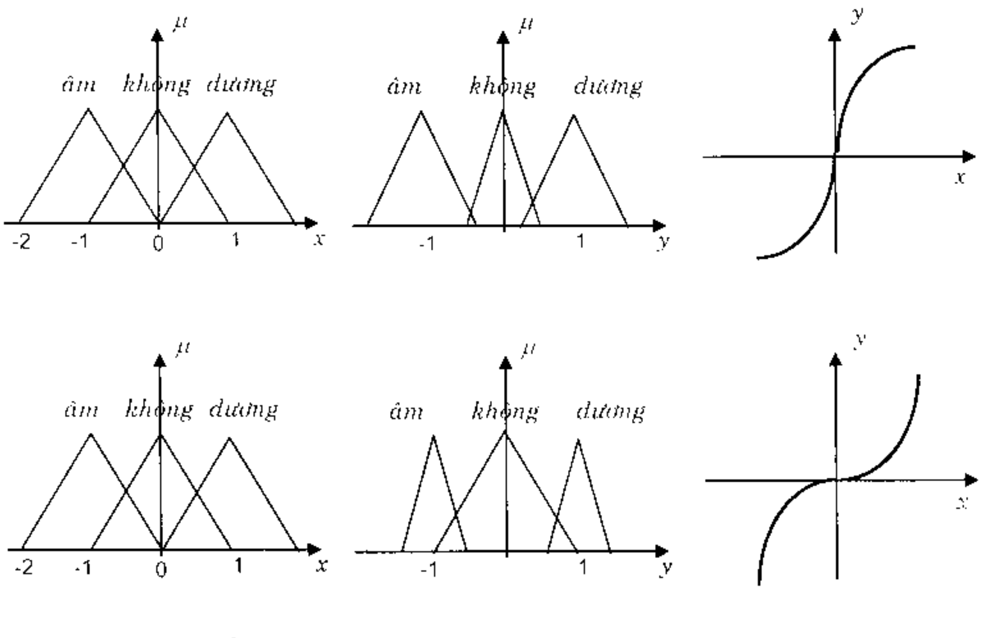
Thay đổi độ rộng các hàm thuộc đầu ra

Giả sử độ rộng của $\mu_{không}(y)$ thay đổi từ 2 lên 2+b còn độ rộng của $\mu_{âm}(y)$ và $\mu_{dương}(y)$ được giữ nguyên (*hình 2.12*). Sai lệch Δ của y' khi đó sẽ là

$$\Delta = \frac{M_1}{\frac{(2+b)(2-H_2)}{2} H_2 - A_1} - \frac{M_1}{A_2 + H_1}, \tag{2.6}$$

trong đó các hằng số M_1, A_1 và A_2 được tính theo (1.50a) và không phụ thuộc vào b.

Công thức (2.6) chỉ rằng Δ liên tục và trơn theo b nên không thể dựa vào b để làm tăng hoặc giảm số điểm không liên tục hoặc không trơn của $y(x)$. Do đó cũng giống như việc thay đổi phần miền xác định chung của $\mu_{âm}(y)$, $\mu_{không}(y)$, $\mu_{dương}(y)$, sự thay đổi độ rộng các hàm thuộc đầu ra cũng sẽ không làm thay đổi được đặc tính phi tuyến cơ bản của quan hệ truyền đạt $y(x)$ như chuyển từ dạng relay 2 vị trí sang 3 vị trí .... *Hình 2.13* minh họa thêm cho điều khẳng định trên với hai trường hợp thay đổi độ rộng các tập mờ đầu ra.



Hình 2.13: Thay đổi độ rộng của tập mờ đầu ra và $y(x)$ tương ứng.

2.1.3 Bộ điều khiển mờ hai vị trí có trễ

Nguyên tắc làm việc của khâu relay 2 vị trí có trễ với đặc tính tĩnh $y(x)$

$$y = \begin{cases} 1 & \text{khi} \quad x \geq a \\ \operatorname{sgn}\left(\dfrac{dx}{dt}\right) & \text{khi} \quad |x| < a \\ -1 & \text{khi} \quad x \leq -a \end{cases}, \tag{2.7}$$

trong đó $x(t)$ là tín hiệu vào và $y(t)$ là tín hiệu ra được mô tả bằng lời như sau:

- là khâu relay hai vị trí thông thường, khi $|x| \geq a$,
- trong khoảng $|x| < a$ đầu ra relay giữ nguyên giá trị cũ trước đó.

So sánh với nguyên lý của bộ điều khiển ba vị trí đã được trình bày trong mục 2.1.1 khi

là khâu relay hai vị trí thông thường, với $|x| \geq a$.

- trong khoảng $|x| < a$ đầu ra có giá trị bằng 0,

thì để tạo được ra khoảng trễ thay cho y=0 khi $|x| < a$, các mệnh đề điều kiện của luật hợp thành R phải không được phủ kín toàn bộ miền giá trị ngôn ngữ đầu vào, hay nói một cách khác theo ngôn ngữ toán học thì tập R của các mệnh đề hợp thành R_k *không đầy đủ*.

Định nghĩa 2.1

Một tập $R = \{ R_k | k \in I$, trong đó I là tập chỉ số$\}$ của các mệnh đề hợp thành (luật điều khiển) R_k được gọi là *đầy đủ* khi mọi giá trị rõ x_0 đầu vào bao giờ cũng tích cực được ít nhất một mệnh đề R_k, tức là

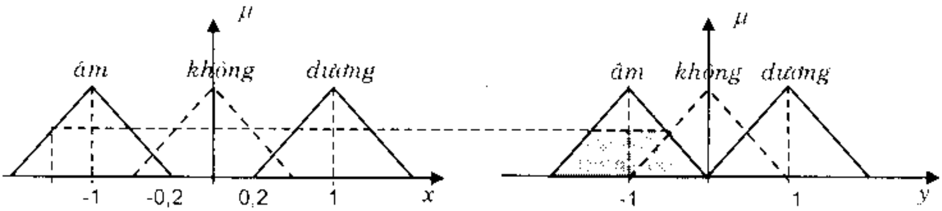
$\forall x_0, \exists k \subset I : H_k > 0$, H_k là độ thỏa mãn của R_k.

Ví dụ luật hợp thành R chỉ gồm hai mệnh đề hợp thành

a) R_1: NẾU χ = *âm* THÌ γ = *âm* HOẶC

b) R_2: NẾU χ = *dương* THÌ γ = *dương*,

không đầy đủ vì không phủ kín cả ba giá trị ngôn ngữ đầu vào là *âm*, *không* và *dương*. *Hình 2.14* trình bầy các hàm thuộc của các giá trị χ=*âm*, χ=*không*, χ=*dương* của biến ngôn ngữ đầu vào và γ=*âm*, γ=*không*, γ=*dương* của biến ngôn ngữ đầu ra, trong đó hai hàm $\mu_{không}(x)$ và $\mu_{không}(y)$ được vẽ bằng đường gạch nét để chỉ rằng chúng không tham gia vào trong luật hợp thành R.



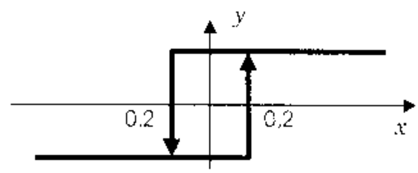
Hình 2.14: Tạo bộ điều khiển mờ hai vị trí có trễ.

Xét bộ điều khiển mờ với luật hợp thành R trên có khâu giải mờ làm việc theo nguyên lý điểm trọng tâm. Khi x được tăng dần từ -1 đến -0,2, do chỉ có một mình R_1 tích cực nên đầu ra luôn có giá trị bằng -1. Cũng như vậy đầu ra luôn có giá trị 1 khi x nằm trong khoảng [0,2 : 1] vì chỉ có một mình R_2 tích cực . Trong khoảng miền giá trị vật lý không được phủ (-0,2 : 0,2), cùng với nguyên tắc là đầu ra y sẽ giữ nguyên giá trị cũ trước đó, thì

$$y = \begin{cases} 1 & \text{khi } x \text{ tăng} \\ 1 & \text{khi } x \text{ giảm} \end{cases}.$$

Hình 2.14 biểu diễn đường đặc tính $y(x)$ của quan hệ truyền đạt.

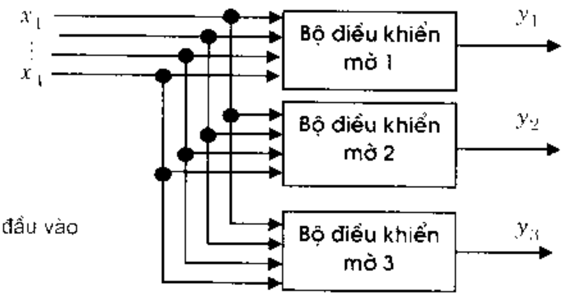
Hình 2.14: Bộ điều khiển mờ hai vị trí có trễ.



2.2 Xây dựng công thức quan hệ truyền đạt

Phần trên đã trình bày sự phụ thuộc của quan hệ truyền đạt $y(x)$ giữa tín hiệu (rõ) đầu vào x và tín hiệu (rõ) đầu ra y của một bộ điều khiển mờ vào việc định nghĩa các giá trị ngôn ngữ (tập mờ) trên cơ sở những ví dụ cụ thể. Các kết quả thu được hoàn toàn có thể mở rộng cho một bộ điều khiển mờ bất kỳ như bộ điều khiển mờ nhiều vào nhiều ra (MIMO).

Việc xây dựng công thức tổng quát $y(x)$ cho quan hệ truyền đạt bộ điều khiển MIMO sau đây chỉ cần thực hiện cho bộ điều khiển mờ với nhiều đầu vào và một đầu ra (bộ MISO) là đủ vì một bộ điều khiển mờ có nhiều đầu ra bất kỳ đều có thể được thay bằng một tập các bộ điều khiển với một đầu ra (*hình 2.15*).



Hình 2.8: Bộ điều khiển mờ với 4 đầu vào và 3 đầu ra.

Luật điều khiển của bộ điều khiển mờ nhiều đầu vào và một đầu ra có dạng

$$R_k:\ \text{NẾU } \chi_1 = A_1{}^k \text{ VÀ } \chi_2 = A_2{}^k \text{ VÀ } \cdots \text{ VÀ } \chi_d = A_d{}^k \text{ THÌ } \gamma = B_k \tag{2.8}$$

trong đó $k = 1, 2, \cdots, n$ và các tập mờ $A_m{}^k$, $m = 1, 2, \cdots, d$ có cùng cơ sở X. Luật điều khiển (2.8) còn có tên gọi là luật chuẩn (*canonical*) vì nó bao hàm rất nhiều những dạng luật điều khiển khác như:

$$R:\ \text{NẾU } \chi_1 = A_1 \text{ VÀ } \cdots \text{ VÀ } \chi_m = A_m \text{ HOẶC } \chi_{m+1} = A_{m+1} \text{ VÀ}$$
$$\text{VÀ } \cdots \text{ VÀ } \chi_d = A_d \text{ THÌ } \gamma = B$$

hay

$$R:\ \text{NẾU } \chi_1 = A_1 \text{ VÀ } \chi_2 = A_2 \text{ VÀ } \cdots \text{ VÀ } \chi_m = A_m \text{ THÌ } \gamma = B$$

nếu $m < d$....

2.2.1 Quan hệ vào ra của thiết bị hợp thành

Một tập (luật hợp thành) R của n luật điều khiển (2.8) được gọi là

- *đủ* (complete), nếu không có một giá trị rõ $x_0 \in X$ nào của đầu vào làm cho độ thỏa mãn mọi luật R_k của R bằng 0, tức là

$$\forall x_0 \in X,\ \exists m \in \{1, 2, \cdots, d\} : \mu_{A_m^k}(x_0) \neq 0,\ \forall k \in \{1, 2, \cdots, n\} \tag{2.9}$$

- *nhất quán* (consistent), nếu không có hai luật điều khiển nào có cùng mệnh đề điều kiện nhưng lại khác mệnh đề kết luận.

Với các bước triển khai đã trình bày trong mục 1.5.4, quan hệ vào ra của thiết bị hợp thành được thực hiện qua:

1) *Bước 1*: Tìm tập mờ đầu ra của R_k

Ký hiệu $\underline{x}$ là một vector d chiều có phần tử thứ m là một giá trị rõ bất kỳ thuộc tập mờ A_m^k, tức là

$$\underline{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix},$$

trong đó x_m là một giá trị thuộc miền xác định của $\mu_{A_m^k}(x)$.

Độ thỏa mãn H_k của luật R_k được tính theo

a) $H_k = \min\{\, \mu_{A_1^k}(x_1),\ \mu_{A_2^k}(x_2),\ \cdots,\ \mu_{A_d^k}(x_d)\,\}$, (2.10)

nếu sử dụng (1.12) để thực hiện phép giao trong mệnh đề điều kiện của R_k

b) $H_k = \prod_{m=1}^{d} \mu_{A_m^k}(x_m)$ (2.11)

nếu sử dụng công thức "*Tích đại số*" để thực hiện phép giao trong mệnh đề điều kiện của R_k.

Từ đó tập mờ đầu ra B'_k sẽ có hàm thuộc

a) $\mu_{B'_k}(y) = \min\{\, H_k, \mu_{B_k}(y)\,\}$ (2.12)

nếu sử dụng quy tắc triển khai max-MIN hoặc sum-MIN để cài đặt R_k.

b) $\mu_{B'_k}(y) = H_k \cdot \mu_{B_k}(y)$ (2.12)

nếu sử dụng quy tắc triển khai max-PROD hoặc sum-PROD.

2) *Bước 2*: Tìm tập mờ đầu ra của R

Sau khi đã có được d tập mờ đầu ra cho từng luật điều khiển R_k là

$\mu_{B'_k}(y)$, $k = 1, 2, \cdots, d$.

tập mờ đầu ra chung B' của thiết bị hợp thành

$$R = \bigcup_{k=1}^{n} R_k \tag{2.13}$$

được xác định như sau:

a) $\mu_{B'}(y) = \max\{\ \mu_{B'_k}(y), k = 1, 2, \ldots, n\ \}$

nếu sử dụng (1.10) để thực hiện phép hợp (2.13),

b) $\mu_{B'}(y) = \min\left\{1\ ,\ \sum_{k=1}^{n} \mu_{B'_k}(y)\right\}$

nếu sử dụng công thức Lukasiewicz để thực hiện phép hợp (2.13).

Từ những công thức (2.10) ÷ (2.12) của bước 1 và (2.14), (2.15) của bước 2 dễ dàng suy ra được công thức biểu diễn quan hệ vào ra $\underline{x} \mapsto \mu_{B'}(y)$ của thiết bị hợp thành. Cho những nguyên tắc triển khai, công thức áp dụng thực hiện phép giao và hợp trên tập mờ khác nhau thì có công thức biểu diễn quan hệ vào ra khác nhau. Ví dụ nếu áp dụng "tích đại số" cho phép giao, nguyên tắc triển khai max–MIN để cài đặt luật điều khiển và công thức (1.10) cho phép hợp thì

$$\mu_{B'}(y) = \max_{1 \le k \le n}\left\{\min\left\{\prod_{m=1}^{d} \mu_{A_m^k}(x_m)\ ,\ \mu_{B_k}(y)\right\}\right\} \tag{2.16}$$

hoặc cho nguyên tắc triển khai sum–PROD, phép giao theo (1.12) và công thức Lukasiewicz cho phép hợp thì

$$\mu_{B'}(y) = \min\left\{1\ ,\ \sum_{k=1}^{n}\left[\mu_{B_k}(y) \min_{1 \le m \le d} \mu_{A_m^k}(x_m)\right]\right\}. \tag{2.17}$$

2.2.2 Quan hệ vào ra của khâu giải mờ

Công thức tổng quát biểu diễn quan hệ vào ra $\mu_{B'}(y) \mapsto y'$ của khâu giải mờ dễ dàng suy ra được từ các công thức (1.44) ÷ (1.51) đã trình bày trong mục 1.6. Nếu ký hiệu H là độ cao của B', G là miền giá trị vật lý y có độ phụ thuộc bằng H và S là miền xác định của B' thì:

1) $$y' = \frac{\inf_{y \in G} y + \sup_{y \in G} y}{2} \tag{2.18}$$

cho phương pháp cực đại theo nguyên lý trung bình,

2) $$y' = \inf_{y \in G} y \tag{2.19}$$

cho phương pháp cực đại theo nguyên lý cận trái,

3) $$y' = \sup_{y \in G} y \tag{2.20}$$

cho phương pháp cực đại theo nguyên lý cận phải.

4) $$y' = \frac{\int_S y\mu_{B'}(y)dy}{\int_S \mu_{B'}(y)dy} \tag{2.21}$$

cho phương pháp điểm trọng tâm,

5) $$y' = \frac{\sum_{k=1}^{n}\left(\int_S y\mu_{B'_k}(y)dy\right)}{\sum_{k=1}^{n}\left(\int_S \mu_{B'_k}(y)dy\right)} = \frac{\sum_{k=1}^{n} M_k}{\sum_{k=1}^{n} A_k} \tag{2.22}$$

cho phương pháp điểm trọng tâm và nguyên tắc triển khai sum–MIN, trong đó M_k và A_k được tính theo (1.50),

6) $$y' = \frac{\sum_{k=1}^{n} y_k H_k}{\sum_{k=1}^{n} H_k} \tag{2.23}$$

cho phương pháp điểm trọng tâm và nguyên tắc triển khai sum–MIN với quy ước singleton (phương pháp độ cao), trong đó y_k là điểm mẫu thỏa mãn $\mu_{B'_k}(y_k) = H_k$.

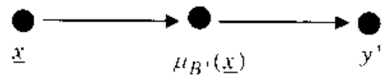
Chú ý rằng các công thức (2.22) và (2.23) cũng có thể được sử dụng khi nguyên tắc triển khai không phải là sum–MIN.

2.2.3 Quan hệ truyền đạt $y(\underline{x})$

Quan hệ truyền đạt $y(\underline{x})$ của bộ điều khiển mờ nhận được thông qua việc ghép nối tiếp hai ánh xạ $\underline{x} \mapsto \mu_{B'}(y)$ và $\mu_{B'}(y) \mapsto y'$ với nhau để có $\underline{x} \mapsto y'$.

Công thức biểu diễn ánh xạ tích nhận được phụ thuộc vào thiết bị hợp thành và phương pháp giải mờ được sử dụng.

Hình 2.9: Tích của hai ánh xạ.



Ví dụ nếu áp dụng "tích đại số" cho phép giao, nguyên tắc triển khai max MIN để cài đặt luật điều khiển, công thức (1.10) cho phép hợp và giải mờ theo phương pháp điểm trọng tâm thì quan hệ truyền đạt có dạng sau

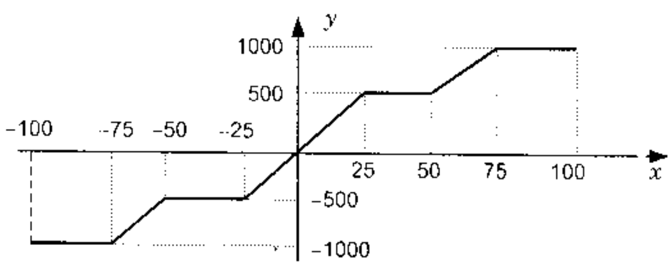
$$y' = \frac{\int\limits_S y\left(\max\limits_{1\le k\le n}\left\{\min\left\{\prod\limits_{m=1}^{d}\mu_{A_m^k}(x_m)\ ,\mu_{B_k}(y)\right\}\right\}\right)dy}{\int\limits_S \left(\max\limits_{1\le k\le n}\left\{\min\left\{\prod\limits_{m=1}^{d}\mu_{A_m^k}(x_m)\ ,\mu_{B_k}(y)\right\}\right\}\right)dy} \qquad (2.24)$$

Cũng như vậy với nguyên tắc triển khai sum–PROD, phép giao theo (1.12), công thức Lukasiewicz cho phép hợp và giải mờ theo (2.22)

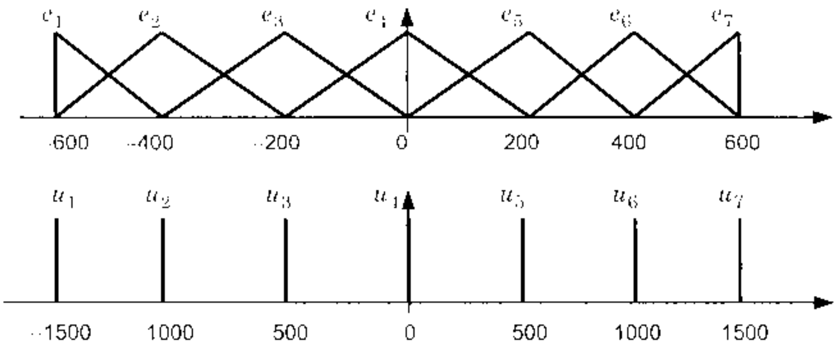
$$y' = \frac{\sum\limits_{k=1}^{n}\left(\int\limits_S y\mu_{B_k}(y)\min\limits_{1\le m\le d}\mu_{A_m^k}(x_m)dy\right)}{\sum\limits_{k=1}^{n}\left(\int\limits_S \mu_{B_k}(y)\min\limits_{1\le m\le d}\mu_{A_m^k}(x_m)dy\right)} \qquad (2.25)$$

Câu hỏi ôn tập và bài tập

1) Chọn các tập mờ cho biến vào x và biến ra y để được bộ điều khiển mờ có đặc tính vào/ ra tĩnh như hình dưới.



2) Cho các tập mờ đầu vào của biến sai lệch e và các tập mờ đầu ra của biến ra u cua bộ điều khiển mờ có dạng như sau:



Luật hợp thành được xây dựng theo quan hệ:

R_1: NẾU $e=e_1$ THÌ $u=u_1$

R_2: NẾU $e=e_2$ THÌ $u=u_2$

⋮

R_7: NẾU $e=e_7$ THÌ $u=u_7$

Hãy xác định quan hệ vào/ra tĩnh của bộ điều khiển mờ và giá trị rõ u_0 biết giá trị rõ đầu vào là $e_0=-250$. nếu

a) luật hợp thành được xây theo công thức max–MIN và giải mờ theo phương pháp điểm trọng tâm.

b) luật hợp thành được xây dựng theo công thức max–MIN và giải mờ theo phương pháp cực đại với nguyên lý cận trái.

c) luật hợp thành theo công thức max–PROD và giải mờ theo phương pháp điểm trọng tâm.

d) luật hợp thành theo công thức sum–MIN và giả mờ theo phương pháp điểm trọng tâm.

e) luật hợp thành theo công thức sum–PROD và giải mờ theo phương pháp điểm trọng tâm.

3 ĐIỀU KHIỂN MỜ

Điều khiển mờ chiếm một vị trí rất quan trọng trong điều khiển học kỹ thuật hiện đại. Ngay từ buổi đầu, điều khiển mờ đã đem lại sự ngạc nhiên đáng kể rằng hoàn toàn trái với tên gọi của nó, *kỹ thuật điều khiển này đồng nghĩa với độ chính xác và khả năng thực hiện*. Tuy là ngành kỹ thuật điều khiển non trẻ nhưng những ứng dụng trong công nghiệp của điều khiển mờ thật rộng rãi như: điều khiển nhiệt độ, điều khiển giao thông vận tải, điều khiển trong các lĩnh vực sản xuất hàng hóa dân dụng ....

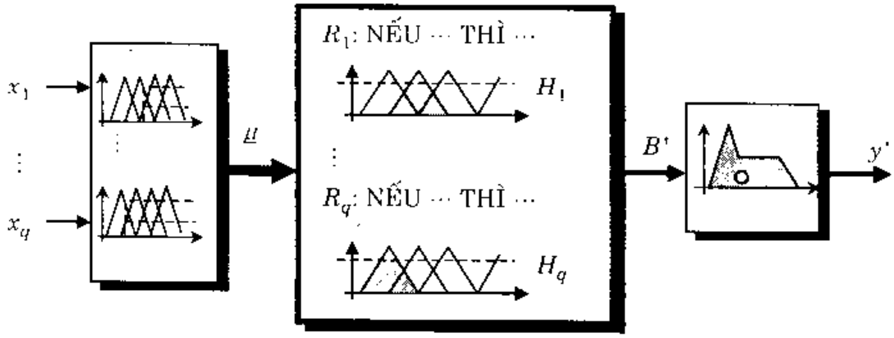
Trong thực tế, nhiều giải pháp tổng hợp, thiết kế bộ điều khiển kinh điển thường bị bế tắc khi gặp những bài toán có độ phức tạp của hệ thống cao, độ phi tuyến lớn, sự thường xuyên thay đổi trạng thái và cấu trúc của đối tượng ..., hoặc giả nếu có thể tổng hợp được trong phạm vi lý thuyết thì khi thực hiện cũng gặp không ít những khó khăn về giá thành và độ tin cậy của sản phẩm. Những khó khăn đó sẽ không còn là những vấn đề nan giải khi bộ điều khiển được thiết kế dựa trên cơ sở logic mờ và càng đơn giản hơn trong việc thực hiện giải pháp này. Các bộ điều khiển được thiết kế trên cơ sở logic mờ có tên gọi là *bộ điều khiển mờ*. Chúng có chung một đặc điểm là làm việc theo nguyên tắc sao chép lại kinh nghiệm, tri thức của con người trong điều khiển, vận hành máy móc.

So với các giải pháp kỹ thuật từ trước đến nay được áp dụng để tổng hợp các hệ thống điều khiển, phương pháp tổng hợp hệ thống bằng logic mờ chỉ ra những ưu điểm rõ rệt sau đây:

1) Khối lượng công việc thiết kế giảm đi nhiều do không cần sử dụng mô hình đối tượng, với các bài toán thiết kế có độ phức tạp cao, giải pháp dùng bộ điều khiển mờ cho phép giảm khối lượng tính toán và giá thành sản phẩm,
2) Bộ điều khiển mờ dễ hiểu hơn so với các bộ điều khiển khác (cả về kỹ thuật) và dễ dàng thay đổi,
3) Trong nhiều trường hợp bộ điều khiển mờ làm việc ổn định hơn, bền vững *(robust)* hơn và chất lượng điều khiển cao hơn.

3.1 Bộ điều khiển mờ cơ bản

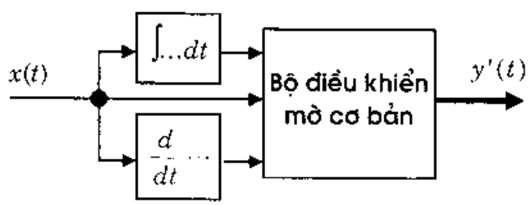
Chương 1 và 2 đã trình bày những thành phần cơ bản của một bộ điều khiển mờ bao gồm khâu Fuzzy hóa, thiết bị thực hiện luật hợp thành và khâu giải mờ (xem *hình 3.1*). Một bộ điều khiển mờ chỉ gồm ba thành phần như vậy có tên gọi là *bộ điều khiển mờ cơ bản*.



Hình 3.1: Bộ điều khiển mờ cơ bản.

Do bộ điều khiển mờ cơ bản chỉ có khả năng xử lý các giá trị tín hiệu hiện thời nên nó thuộc nhóm các *bộ điều khiển tĩnh*. Tuy vậy để mở rộng miền ứng dụng của chúng vào các bài toán điều khiển động, các khâu động học cần thiết sẽ được nối thêm vào bộ điều khiển mờ cơ bản (xem *hình 3.2*). Các khâu động đó chỉ có nhiệm vụ cung cấp thêm cho bộ điều khiển mờ cơ bản các giá trị đạo hàm hay tích phân của tín hiệu. Cùng với những khâu động bổ sung này, bộ điều khiển cơ bản sẽ được gọi là *bộ điều khiển mờ*.

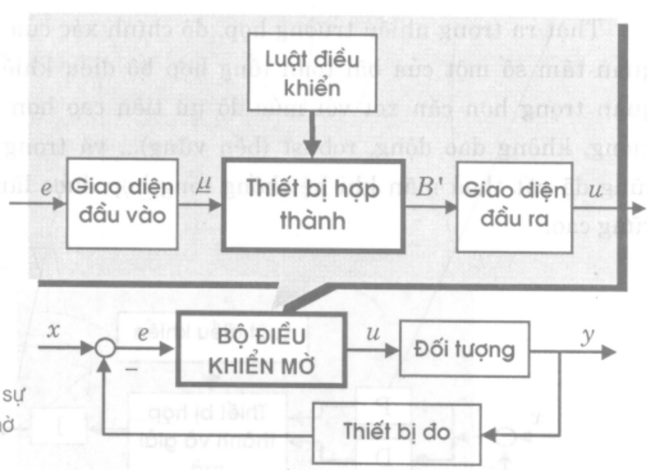
Hình 3.2: Ví dụ về một bộ điều khiển mờ động.



3.2 Nguyên lý điều khiển mờ

Trước khi đi vào những ứng dụng cụ thể của bộ điều khiển mờ, hãy nghiên cứu cấu trúc và nguyên lý làm việc của một hệ thống điều khiển mờ. Trong *hình 3.3* là một hệ thống điều khiển tự động với bộ điều khiển mờ.

Về nguyên tắc, hệ thống điều khiển mờ cũng không có gì khác với các hệ thống điều khiển tự động thông thường khác. Sự khác biệt ở đây là bộ điều khiển mờ làm việc có tư duy như "bộ não" dưới dạng trí tuệ nhân tạo. Nếu khẳng định làm việc với bộ điều khiển mờ có thể giải quyết được mọi vấn đề từ trước đến nay chưa giải quyết được theo phương pháp kinh điển thì không hoàn toàn chính xác, vì hoạt động của bộ điều khiển phụ thuộc vào kinh nghiệm và phương pháp rút ra kết luận theo tư duy của con người, sau đó được cài đặt vào máy tính trên cơ sở của logic mờ. Hệ thống điều khiển mờ do đó cũng có thể coi như là một hệ thống *neuron* (hệ thần kinh), hay đúng hơn là một hệ thống điều khiển được thiết kế mà không cần biết trước mô hình của đối tượng.



Hình 3.3: Hệ kín, phản hồi âm với sự tham gia của bộ điều khiển mờ

Hệ thống điều khiển mờ được thiết kế trên:

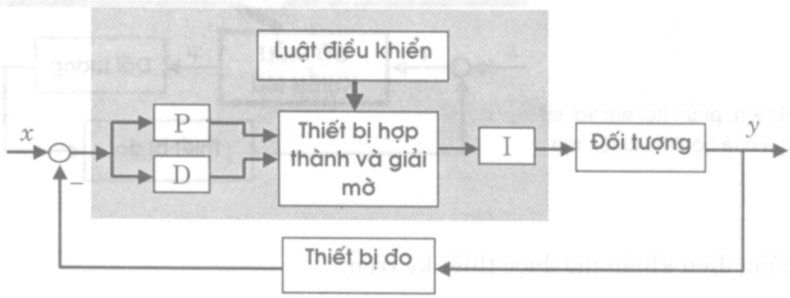
- *dao diện đầu vào* bao gồm khâu Fuzzy hóa và các khâu phụ trợ thêm để thực hiện các bài toán động như tích phân, vi phân ...,
- *thiết bị hợp thành* mà bản chất của nó sự triển khai luật hợp thành R được xây dựng trên cơ sở luật điều khiển hay như trong một số các tài liệu khác còn gọi là luật quyết định và

- *khâu giao diện đầu ra* (khâu chấp hành) gồm khâu giải mờ và các khâu giao diện trực tiếp với đối tượng.

Nguyên tắc tổng hợp một bộ điều khiển mờ hoàn toàn dựa vào những phương pháp toán học trên cơ sở định nghĩa các biến ngôn ngữ vào/ra và sự lựa chọn những luật điều khiển. Do các bộ điều khiển mờ có khả năng xử lý các giá trị vào/ra biểu diễn dưới dạng dấu phẩy động với độ chính xác cao nên chúng hoàn toàn đáp ứng được các yêu cầu của một bài toán điều khiển "rõ ràng" và "chính xác ".

Trong sơ đồ mạch điều khiển trong *hình 3.1* có khâu đối tượng. Đối tượng này được điều khiển bằng đại lượng u là tín hiệu đầu ra của bộ điều khiển mờ. Vì các tín hiệu điều khiển đối tượng là các "tín hiệu rõ", nên tín hiệu ra của bộ điều khiển mờ trước khi đưa vào điều khiển đối tượng phải qua khâu giải mờ nằm trong bộ giao diện đầu ra. Các tín hiệu ra y của đối tượng được đo bằng các bộ cảm biến và được xử lý sơ bộ trước khi đưa vào bộ điều khiển. Các tín hiệu này cũng là các "tín hiệu rõ", do vậy để bộ điều khiển mờ hiểu được chúng, tín hiệu y và ngay cả tín hiệu chủ đạo x phải được mờ hóa.

Thật ra trong nhiều trường hợp, độ chính xác của hệ thống không phải là điều quan tâm số một của bài toán tổng hợp bộ điều khiển, mà còn nhiều điểm khác quan trọng hơn cần xét với mức độ ưu tiên cao hơn như vấn đề tiết kiệm năng lượng, không dao động, robust (bền vững)... và trong nhiều trường hợp người ta cũng đã rất thoả mãn khi hệ thống tổng hợp được làm việc ổn định và có độ bền vững cao.



Hình 3.4: Bộ điều khiển mờ có khâu P, khâu D ở đầu vào và I ở đầu ra.

Trái tim của bộ điều mờ chính là luật điều khiển mờ cơ bản có dạng là tập các mệnh đề hợp thành cùng cấu trúc NẾU ··· THÌ ··· và nguyên tắc triển khai các mệnh đề hợp thành đó có tên là nguyên tắc max–MIN hay sum–MIN .... Mô hình R của luật điều khiển được xây dựng theo một nguyên tắc triển khai đã chọn trước và có tên gọi là luật hợp thành. Thiết bị thực hiện luật hợp thành trong bộ điều khiển mờ là *thiết bị hợp thành*. Hai thành phần cơ bản là luật điều khiển và nguyên tắc triển khai hình thành nên cuộc sống của bộ điều mờ mà ở đó nguyên tắc triển khai như một động cơ và luật điều khiển như là nguồn cung cấp năng lượng cho động cơ quay. Để cho thiết bị thực hiện luật điều khiển làm việc đúng chế độ phải chọn cho nó các biến ngôn ngữ hợp lý có khả năng biểu diễn các đại lượng vào/ra chuẩn và phù hợp với luật điều khiển. Dạng đúng của các luật điều khiển mờ cơ bản được hình thành nhờ quá trình luyện tập và kinh nghiệm thiết kế. Chương này sẽ trình bày các phương pháp tổng hợp những bộ điều khiển mờ cơ bản như hệ mờ P, hệ mờ PID, hệ mờ trượt. Những hệ mờ đặc biệt như hệ mờ lai, hệ mờ thích nghi sẽ được mô tả sau trong chương 4.

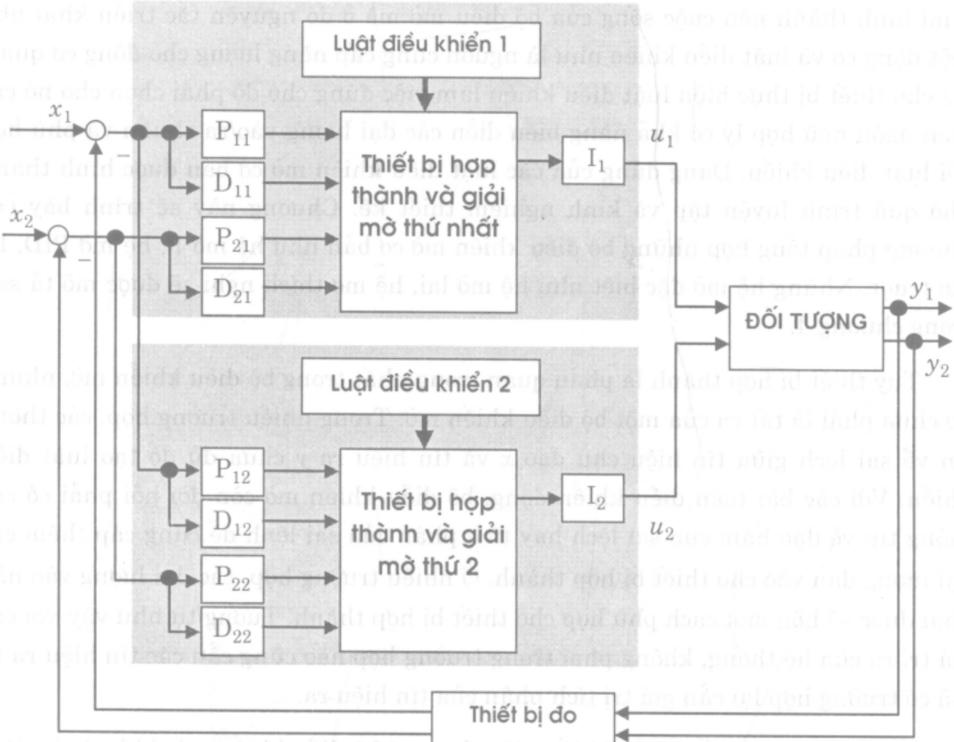
Tuy thiết bị hợp thành là phần quan trọng nhất trong bộ điều khiển mờ, nhưng nó chưa phải là tất cả của một bộ điều khiển mờ. Trong nhiều trường hợp, các thông tin về sai lệch giữa tín hiệu chủ đạo x và tín hiệu ra y chưa đủ để tạo luật điều khiển. Với các bài toán điều khiển động, bộ điều khiển mờ còn đòi hỏi phải có các thông tin về đạo hàm của sai lệch hay tích phân của sai lệch để cung cấp thêm các đại lượng đầu vào cho thiết bị hợp thành. Ở nhiều trường hợp, các đại lượng vào này phải được số hóa một cách phù hợp cho thiết bị hợp thành. Tương tự như vậy với các giá trị ra của hệ thống, không phải trong trường hợp nào cũng cần các tín hiệu ra rõ mà có trường hợp lại cần giá trị tích phân của tín hiệu ra.

Đối với các hệ thống điều khiển gián đoạn có bộ điều khiển mờ, khi nó còn làm việc trên cơ sở các tín hiệu số, có thể thiết kế các bộ điều chỉnh theo luật P (luật tỷ lệ), theo luật I (luật tích phân) và theo luật D (luật vi phân) như sau:

- luật điều khiển P: $y_k = K \cdot x_k$, trong đó K là hệ số khuếch đại,
- luật điều khiển I: $y_{k+1} = y_k + \dfrac{T_a}{T_I} x_k$ với T_I là hằng số tích phân,
- luật điều khiển D: $y_{k+1} = \dfrac{T_D}{T_a}(x_k - y_k)$ với T_D là hằng số vi phân,

trong đó T_a là chu kỳ gián đoạn (chu kỳ lấy mẫu tín hiệu).

Hình 3.4 là một ví dụ về điều khiển một đối tượng đơn giản có một tín hiệu vào và một tín hiệu ra (hệ SISO) bằng bộ điều khiển mờ. Sai lệch e giữa tín hiệu chủ đạo và tín hiệu ra được đưa vào bộ điều chỉnh theo luật PD và sau đó được đưa vào bộ điều khiển mờ, bộ điều chỉnh I được dùng như một thiết bị chấp hành, đầu vào lấy sau bộ giải mờ và đầu ra được dẫn tới đối tượng.



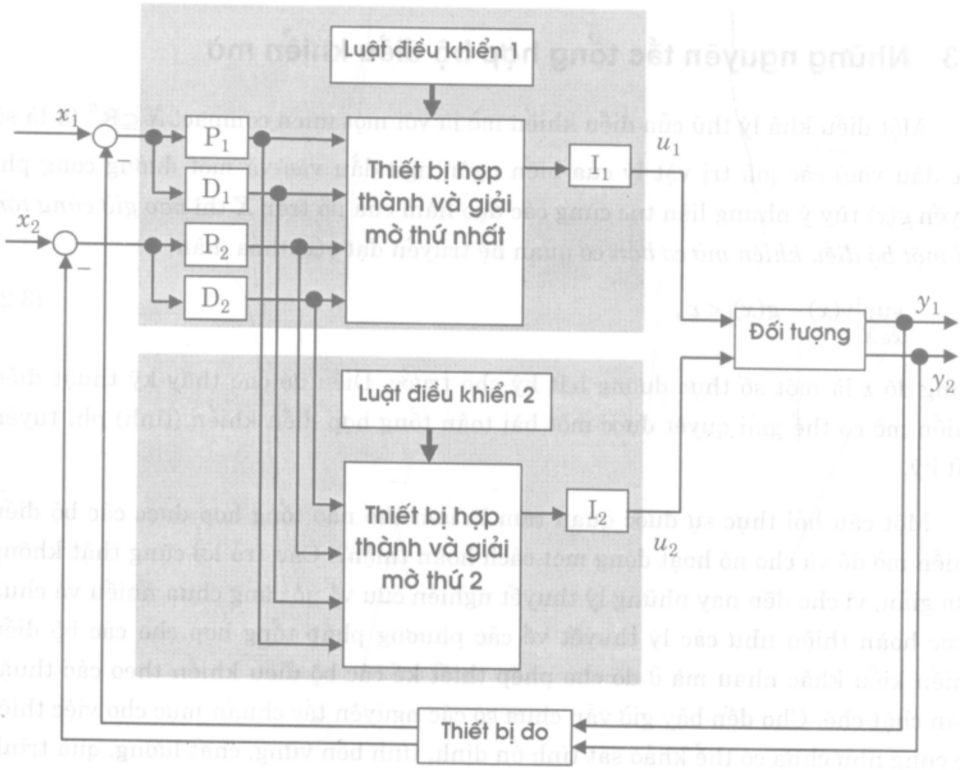
Hình 3.5: Hệ điều khiển mờ nhiều chiều với hai thiết bị hợp thành.

Xét tiếp bài toán điều khiển có thiết bị hợp thành cho phép làm việc được với ít nhất 2 vào và 2 ra. Câu hỏi được đặt ra là làm thế nào để chỉnh định một hệ thống không chỉ bao gồm một đại lượng chỉnh định mà bao gồm hai đại lượng khác nhau?. Vấn đề này có thể giải quyết không có gì khó khăn khi chịu trả giá cho một ít khối lượng tính toán phát sinh bằng cách sử dụng hai thiết bị hợp thành trong hai bộ điều khiển khác nhau. Những thiết bị này cùng xử lý tín hiệu vào như nhau, nhưng đầu ra được xác định theo các điều kiện và nguyên tắc khác nhau như trong *hình 3.5*.

Đối với các bài toán tổng thể có thể phải cần đến nhiều thiết bị thực hiện luật điều khiển theo kiểu được mắc nối tiếp hoặc mắc song song hay mắc theo kiểu mạng như trong *hình 3.6*. Các hệ thống có cấu trúc như vậy rất cần thiết cho cả các hệ thống có nhiều đầu vào, ví dụ như cho 10 đầu vào hoặc nhiều hơn nữa. Trong những trường hợp như vậy nguyên tắc điều khiển phải được biểu diễn rất chi tiết. Nếu như chỉ dùng một thiết bị hợp thành thể hiện luật điều khiển thì luật điều khiển đó sẽ phải có dạng chung:

$$\text{NẾU } \chi_1 = A_1^{\ k} \text{ VÀ } \chi_2 = A_2^{\ k} \text{ VÀ } \chi_3 = A_3^{\ k} \text{ VÀ} \cdots \text{ VÀ } \chi_9 = A_9^{\ k} \text{ VÀ } \cdots$$

$$\text{THÌ } \gamma_1 = B_1^{\ k} \text{ VÀ } \gamma_2 = B_2^{\ k} \text{ VÀ } \cdots \tag{3.1}$$



Hình 3.6: Hệ điều khiển mờ với hai thiết bị hợp thành nối song song.

Thật dễ dàng nhận biết được rằng phải là bậc thiên tài và phải có lòng kiên nhẫn mới làm được công việc cài đặt tổng quát luật điều khiển trên đây thành thiết bị hợp thành, sau đó phải thực hiện giải mờ nữa. Trong những trường hợp như vậy, cách tốt nhất là nên chia bài toán điều khiển thành nhiều bài toán khả thi đơn giản như chuyển luật (3.1) trên (mà thực chất là luật điều khiển nhiều đầu vào, nhiều đầu ra) thành tập các luật chỉ có một đầu ra. Một điều cần nên biết là số lượng mệnh đề hợp thành tăng theo hàm số mũ so với số lượng tín hiệu đầu vào của một hệ thống điều khiển mờ, chính vì thế số lượng luật điều khiển sẽ tăng lên rất nhiều cho một bộ điều khiển mờ có nhiều đầu vào.

Tổng hợp các bộ điều khiển mờ phức tạp như vậy, đương nhiên không phải là một phương án thực tế và cũng không nên phát triển công nghệ theo hướng này. Trong các chương mục tiếp theo, các phương pháp thiết kế các hệ thống đơn giản và những hướng dẫn tổng quát trong việc thiết kế hệ thống sẽ được đề cập tới.

3.3 Những nguyên tắc tổng hợp bộ điều khiển mờ

Một điều khá lý thú của điều khiển mờ là với một miền compact $X \subseteq \mathbb{R}^n$ (n là số các đầu vào) các giá trị vật lý của biến ngôn ngữ đầu vào và một đường cong phi tuyến $g(x)$ tùy ý nhưng liên tục cùng các đạo hàm của nó trên X thì *bao giờ cũng tồn tại một bộ điều khiển mờ cơ bản* có quan hệ truyền đạt $y(x)$ thỏa mãn

$$\sup_{x\in X}\left|y(x)-g(x)\right|<\varepsilon\,, \tag{3.2}$$

trong đó ε là một số thực dương bất kỳ cho trước. Điều đó cho thấy kỹ thuật điều khiển mờ có thể giải quyết được một bài toán tổng hợp điều khiển (tĩnh) phi tuyến bất kỳ.

Một câu hỏi thực sự được quan tâm là làm thế nào tổng hợp được các bộ điều khiển mờ đó và cho nó hoạt động một cách hoàn thiện?. Câu trả lời cũng thật không đơn giản, vì cho đến nay những lý thuyết nghiên cứu về nó cũng chưa nhiều và chưa được hoàn thiện như các lý thuyết về các phương pháp tổng hợp cho các bộ điều khiển kiểu khác nhau mà ở đó cho phép thiết kế các bộ điều khiển theo các thuật toán chặt chẽ. Cho đến bây giờ vẫn chưa có các nguyên tắc chuẩn mực cho việc thiết kế cũng như chưa có thể khảo sát tính ổn định, tính bền vững, chất lượng, quá trình quá độ cũng như quá trình ảnh hưởng của nhiễu... cho các bộ điều khiển mờ và nguyên lý tối ưu các bộ điều khiển này về phương diện lý thuyết. Nhưng cũng đừng vội kết luận rằng các lý thuyết điều khiển cho đến nay đã thực sự hoàn thiện, bởi vì điểm yếu trong lý thuyết hệ mờ là những vấn đề về độ phi tuyến của hệ. Các hệ thống có tính chất phi tuyến khỏe vẫn là vấn đề nan giải đối với các nhà khoa học, những người thường xuyên giải quyết các vấn đề của mình bằng công cụ toán học, vì những thành tựu lý thuyết đẹp đẽ cho hệ tuyến tính không sử dụng được cho hệ phi tuyến và những kết luận tổng quát cho các hệ thống phi tuyến thì hầu như khó đạt được. Các quá trình động trong hệ phi tuyến thường mang lại những điều bất ngờ, thí dụ như trong các hệ tự chỉnh, hệ hỗn loạn (chao).....

Cuối cùng thì rút ra được các kết luận gì?. Trước hết là những điểm sau đây:

1) *Không bao giờ lại thiết kế bộ điều khiển mờ để giải quyết một bài toán tổng hợp mà có thể dễ dàng thực hiện bằng các bộ điều khiển kinh điển* (*bộ điều khiển* P. - PI. - PD. - PID. *bộ điều khiển trạng thái*) *thoả mãn các yêu cầu đặt ra.*

Nhưng ngay cả nghĩa *thoả mãn các yêu cầu đặt ra* cũng hoàn toàn mang tính chất mờ, bởi vì *thoả mãn* ở đây cũng chỉ là đối với một vài chỉ tiêu.

Hãy thử *nghĩ* đến vấn đề phải ổn định hình ảnh cho một máy quay phim được đặt trên một vật luôn rung. Ví dụ như cần thu các hình ảnh của vệ tinh, quay một chiếc xe ô tô đang chuyển động... mà máy quay phim lại được cầm trên tay. Với vấn đề đặt ra như trên cũng có thể giải quyết được bằng các phương pháp kinh điển, ổn định hệ thống quay bằng mạch ổn định điện, bộ suy giảm thủy khí, mạch ổn dịnh ảnh chi ly,... Tóm lại, để làm được việc này phải tốn rất nhiều công sức. Ở đây, chỉ cần sử dụng một bộ logic mờ với rất ít các điều kiện để giải quyết vấn đề ổn định ảnh, và lời giải này có thể áp dụng cho nhiều các trường hợp khác nhau.

Trước tiên, *đừng* bắt đầu bằng những nguyên tắc cứng rắn, hãy nên dùng những kinh nghiệm và tìm cách thể hiện nó bằng logic mờ, nếu như nó mang lại kết quả nhanh hơn và tốt hơn. Một hướng giải quyết bài toán thiết kế hệ thống điều khiển tự động rất hiệu quả đó là kết hợp giữa các phương pháp kinh điển với logic mờ. Nếu tổng hợp hệ thống theo cách này sẽ đạt được các ưu điểm của cả hai phương pháp (kinh điển và mờ).

2) *Kết luận tiếp theo là hiện nay, việc sử dụng bộ điều khiển mờ cho các hệ thống cần độ an toàn cao vẫn còn bị hạn chế, do những yêu cầu chất lượng và mục đích của hệ thống chỉ có thể xác định và đạt được qua thực nghiệm.*

Nếu như các tính chất này có thể thể hiện bằng các luật thì chỉ còn một câu hỏi phải giải quyết đó là mô hình phải chính xác như thế nào được xem là đủ so với đối tượng thực.

3) *Kết luận thứ ba và có lẽ là kết luận quan trọng nhất là bộ điều khiển mờ phải được phát triển qua thực nghiệm.*

Trong thực tế tuy phần lớn việc xây dựng mô hình hoặc tổng hợp hệ thống kinh điển đều phải dựa vào thực nghiệm và nhiều mạch vòng nội suy nhưng không phải kết quả cuối cùng nào cũng được chấp nhận. Đáng tiếc là trong nhiều trường hợp sử dụng các bộ điều khiển thông thường như bộ điều khiển PID hay bộ điều khiển trạng thái không tận dụng được các kinh nghiệm của con người, nhất là trong bài toán điều khiển các quá trình công nghệ hoá học, tự động điều khiển ô tô.... Con người không phải lúc nào cũng đòi hỏi hệ thống điều khiển tự động phải có chất lượng cao nhất. Theo những chuyên gia có kinh nghiệm thì chỉ tiêu có mức ưu tiên cao nhất trong bài toán tổng hợp hệ thống thường là tính bền vững.

Một điểm thú vị nữa là con người không có những "cảm biến" có thể thực hiện những phép đo hoàn toàn chính xác, song lại có thể điều chỉnh được tính ổn định và bền vững cho hệ thống ngay cả khi lượng thông tin thu thập được không chính xác. Bộ điều khiển mờ, mà thực chất là mô phỏng lại nguyên lý điều khiển của con người, cũng có khả năng đó nên khi áp dụng kỹ thuật điều khiển mờ, không giống như các bộ điều khiển kinh điển, các bộ cảm biến có thể chọn loại rẻ tiền và không cần có độ chính xác cao.

Nếu như những hiểu biết như vậy đã được chuẩn bị, các bộ điều khiển mờ được thiết kế cũng có khả năng làm việc ổn định và bền vững. Và nếu được tối ưu hoá một cách khéo léo, bộ điều khiển mờ trong nhiều trường hợp có thể làm việc tốt hơn cả sự linh hoạt của con người. Hơn nữa bộ điều khiển mờ làm việc không mệt mỏi và không bị mất tập trung như con người. Trong những trường hợp chưa có đầy đủ sự hiểu biết và kinh nghiệm, con người có thể điều khiển đối tượng bằng cách làm ngay một mô hình cho đối tượng và điều khiển bằng tay. Bằng cách này con người tự thu thập những hiểu biết và kinh nghiệm cần thiết.

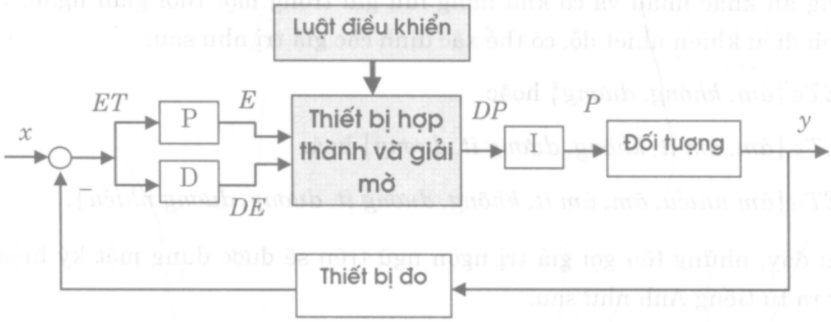
Giả thiết rằng, người thiết kế đã có đủ các kinh nghiệm và muốn chuyển nó thành thiết bị hợp thành trong một bộ điều khiển mờ thì phải tiến hành các bước sau đây:

- định nghĩa tất cả các biến ngôn ngữ vào và ra,
- định nghĩa tập mờ (giá trị ngôn ngữ) cho các biến vào/ ra,
- xây dựng các luật điều khiển (các mệnh đề hợp thành),
- chọn thiết bị hợp thành (max-MIN hay sum-MIN...),
- tối ưu hệ thống.

Dưới đây là một ví dụ về bài toán điều hòa nhiệt độ trong một căn phòng. Vấn đề đặt ra là tổng hợp một bộ điều khiển để điều hòa nhiệt độ cho căn phòng đó. Thiết bị điều khiển nhiệt độ hai chiều nóng lạnh với việc chỉnh định nhiệt độ mỗi chiều hoàn toàn tuyến tính.

3.3.1 Định nghĩa các biến vào/ra

Trong ví dụ này, việc định nghĩa các biến vào/ra cho bộ điều khiển mờ tương đối đơn giản. Đại lượng vào của bộ điều khiển mờ chính là sai lệch (sau đây sẽ được ký hiệu bằng ET) giữa nhiệt độ cần giữ ổn định (tín hiệu chủ đạo x) và nhiệt độ thực y (nhiệt độ đo được từ bộ cảm biến tín hiệu ra của đối tượng). Ngoài ra trong bộ điều khiển mờ còn sử dụng đến sự biến đổi theo thời gian của sai lệch (đạo hàm $\frac{d}{dt}ET$) giữa tín hiệu chủ đạo và tín hiệu ra của đối tượng (được biểu diễn bằng ký hiệu DET).



Hình 3.7: Bộ điều khiển nhiệt độ.

Công suất sưởi nóng hay làm mát P là biến ra của bộ điều khiển. P là một giá trị rõ. Bên trong, thuộc phần giao diện đầu ra, bên cạnh khâu giải mờ bộ điều khiển còn phải sử dụng khâu tích phân để biến đổi giá trị $\frac{d}{dt}P$, được ký hiệu là DP, tại đầu ra của bộ điều khiển mờ cơ bản, tức là đầu ra rõ của thiết bị hợp thành, thành tín hiệu P của bộ điều khiển mờ. Thiết bị hợp thành có hai biến vào là ET, DET và một biến ra DP. *Hình 3.7* mô tả mạch điều khiển trên.

Vì vậy, cấu trúc của bộ điều khiển mờ sẽ rất đơn giản chỉ bao gồm một đầu vào ET và một đầu ra P.

3.3.2 Xác định tập mờ

Bước tiếp theo là định nghĩa các biến ngôn ngữ vào/ra bao gồm số các tập mờ và dạng các hàm thuộc của chúng. Để làm được việc đó cần xác định:

1) *Miền giá trị vật lý (cơ sở) của các biến ngôn ngữ vào/ra.*

Sai lệch nhiệt độ *ET* được chọn trong miền giá trị từ -12^0C đến $+12^0C$, tốc độ biến đổi *DET* của sai lệch có giá trị biến đổi từ $-6^0C/s$ tới $+6^0C/s$ và tốc độ biến đổi công suất ra *DP* nằm trong khoảng $-120W/s$ đến $+120W/s$.

2) *Số lượng tập mờ (giá trị ngôn ngữ).*

Về nguyên tắc, số lượng các giá trị ngôn ngữ cho mỗi biến ngôn ngữ nên nằm trong khoảng từ 3 đến 10 giá trị. Nếu số lượng giá trị ít hơn 3 thì có ít ý nghĩa, vì không thực hiện được việc lấy vi phân. Nếu lớn hơn 10, con người khó có khả năng bao quát, vì con người phải nghiên cứu đầy đủ để đồng thời phân biệt khoảng 5 đến 9 phương án khác nhau và có khả năng lưu giữ trong một thời gian ngắn. Đối với quá trình điều khiển nhiệt độ, có thể xác định các giá trị như sau:

$ET \in \{$*âm, không, dương*$\}$ hoặc

$ET \in \{$*âm, âm ít, không, dương ít, dương*$\}$ hoặc

$ET \in \{$*âm nhiều, âm, âm ít, không, dương ít, dương, dương nhiều*$\}$.

Sau đây, những tên gọi giá trị ngôn ngữ trên sẽ được dùng một ký hiệu ngắn gọn suy ra từ tiếng Anh như sau:

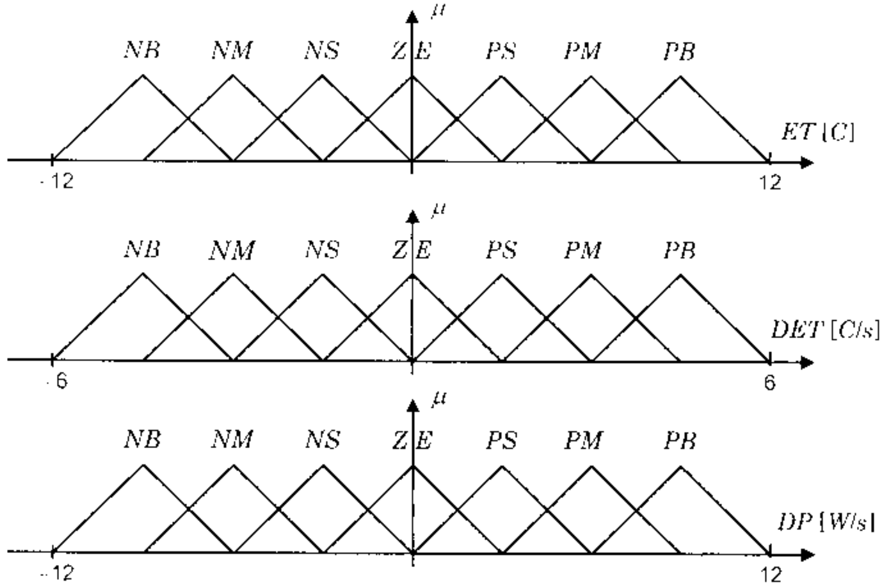
âm nhiều – *NB* (Negative Big),
âm – *NM* (Negative Medium),
âm ít – *NS* (Negative Small),
không – *ZE* (Zero),
dương ít – *PS* (Positive Small),
dương – *PM* (Positive Medium),
dương nhiều – *PB* (Positive Big).

Với những ký hiệu như vậy thì miền xác định (ngôn ngữ) của các biến vào/ra sẽ là:

$ET \in \{NB, NM, NS, ZE, PS, PM, PB\}$,
$DET \in \{NB, NM, NS, ZE, PS, PM, PB\}$,
$DP \in \{NB, NM, NS, ZE, PS, PM, PB\}$.

3) *Xác định hàm thuộc.*

Đây là một điểm cực kỳ quan trọng vì quá trình làm việc của bộ điều khiển mờ rất phụ thuộc vào dạng và kiểu hàm thuộc. Đáng tiếc là không có một lời khuyên nào khác cho việc chọn hàm thuộc là hãy chọn hàm thuộc từ những dạng hàm đã biết trước và mô hình hóa nó cho đến khi nhận được bộ điều khiển mờ làm việc như mong muốn.



Hình 3.8: Định nghĩa tập mờ cho các biến ngôn ngữ.

Cần chọn các hàm thuộc có phần chồng lên nhau và phủ kín miền giá trị vật lý để trong quá trình điều khiển không xuất hiện "lỗ hổng". Trong trường hợp với một giá trị vật lý rõ x_0 của biến đầu vào mà tập mờ B' đầu ra có độ cao bằng 0 (miền xác định là một tập rỗng) và bộ điều khiển không thể đưa ra một quyết định điều khiển nào được gọi là hiện tượng "*cháy nguyên tắc*", lý do là hoặc không định nghĩa được nguyên tắc điều khiển phù hợp hoặc là do các tập mờ của biến ngôn ngữ có những "lỗ hổng".

Cũng như vậy đối với biến ra, các hàm thuộc dạng hình thang với độ xếp chồng lên nhau rất nhỏ, nhìn chung không phù hợp đối với bộ điều khiển mờ vì những lý do đã trình bày ở trên. Nó tạo ra một vùng "chết" (*dead zone*) trong trạng thái làm việc của bộ điều khiển.

Trong một vài trường hợp đặc biệt, chọn hàm thuộc dạng hình thang hoàn toàn hợp lý, đó là trường hợp mà sự thay đổi các miền giá trị của tín hiệu vào không kéo theo sự thay đổi bắt buộc tương ứng cho miền giá trị của tín hiệu ra. Nói chung, hàm thuộc được chọn sao cho miền tin cậy của nó chỉ có một phần tử, hay nói cách khác chỉ tồn tại một điểm vật lý có độ phụ thuộc bằng độ cao của tập mờ. Trong ví dụ này, tập mờ được chọn có dạng hình tam giác cân với

$$\alpha = \beta = 1. \tag{3.3}$$

Hình 3.8 biểu diễn tập mờ được chọn.

4) *Rời rạc hóa các tập mờ.*

Độ phân dải của các giá trị phụ thuộc được chọn trước hoặc là cho các nhóm điều khiển mờ loại dấu phẩy động (các số biểu diễn dưới dạng dấu phẩy động có độ chính xác đơn) hoặc nguyên ngắn (giá trị phụ thuộc là các số nguyên có độ dài 2 byte) hoặc theo byte (các giá trị phụ thuộc là các số không dấu có độ dài 1 byte). Những khả năng để tổng hợp các hệ thống rất khác nhau. Ví dụ như loại linh kiện OMRON-Chip FP-3000 có độ phân dải 12 bit còn linh kiện Togai-Chip FC110 8 bit có độ phân dải từ 8 đến 12 bit. Các hệ thống Fuzzy Tech của hãng Inform cho phép mỗi biến ngôn ngữ có độ phân dải khác nhau và các hệ thống phát triển của hãng Togai có thể xác định trước số lượng các giá trị phụ thuộc trong tập mờ và đối với độ phân dải các giá trị này có thể chọn hoặc là loại dấu phẩy động hoặc là loại giá trị không dấu có độ dài 1 byte. Phương pháp rời rạc hóa sẽ là yếu tố quyết định giữa độ chính xác và tốc độ của bộ điều khiển.

3.3.3 Xây dựng các luật điều khiển

Trong việc xây dựng các luật điều khiển (mệnh đề hợp thành)cần lưu ý là ở vùng lân cận điểm không (ví dụ như trong trường hợp $DT = ZE$) không được tạo ra các "lỗ hổng", bởi vì khi gặp phải các "lỗ hổng" xung quanh điểm làm việc bộ điều khiển sẽ không thể làm việc đúng theo như trình tự đã định -xem thêm phần bộ relay mờ có trễ trong chương 2.

Ngoài ra cần phải để ý rằng, trong phần lớn các bộ điều khiển, tín hiệu ra sẽ bằng không khi tất cả các tín hiệu vào bằng không.

Những nguyên tắc sau đây được đặt ra cho bộ điều khiển nhiệt độ:

R_1: NẾU $ET = NB$ VÀ $DET = ZE$ THÌ $DP = NB$ HOẶC

R_2: NẾU $ET = NM$ VÀ $DET = ZE$ THÌ $DP = NM$ HOẶC

R_3: NẾU $ET = NS$ VÀ $DET = ZE$ THÌ $DP = NS$ HOẶC

R_4: NẾU $ET = ZE$ VÀ $DET = ZE$ THÌ $DP = ZE$ HOẶC

R_5: NẾU $ET = PS$ VÀ $DET = ZE$ THÌ $DP = PS$ HOẶC

R_6: NẾU $ET = PM$ VÀ $DET = ZE$ THÌ $DP = PM$ HOẶC

R_7: NẾU $ET = PB$ VÀ $DET = ZE$ THÌ $DP = PB$ HOẶC

R_8: NẾU $ET = ZE$ VÀ $DET = NB$ THÌ $DP = NB$ HOẶC

R_9: NẾU $ET = ZE$ VÀ $DET = NM$ THÌ $DP = NM$ HOẶC

R_{10}: NẾU $ET = ZE$ VÀ $DET = NS$ THÌ $DP = PS$ HOẶC

R_{11}: NẾU $ET = ZE$ VÀ $DET = PS$ THÌ $DP = PS$ HOẶC

R_{12}: NẾU $ET = ZE$ VÀ $DET = PM$ THÌ $DP = PM$ HOẶC

R_{13}: NẾU $ET = ZE$ VÀ $DET = PB$ THÌ $DP = PB$

Hình 3.9 biểu diễn các luật điều khiển này dưới dạng ma trận. Cách biểu diễn này rất tiện lợi và bao quát. Từ ma trận trong *hình 3.9* có thể thấy rõ chỉ có 13 luật trong tổng số 49 khả năng phối hợp là thích ứng với nguyên tắc điều khiển nhiệt độ. Các luật điều khiển được thiết lập dựa trên mệnh đề hợp thành với hai điều kiện và một kết luận. Thực chất, như ví dụ trên đã chỉ ra, chỉ có một phần trong toàn bộ các khả năng liên kết được lấp kín, đó là các trường hợp theo kinh nghiệm được coi là thường xảy ra trong thực tế.

Hình 3.9: Biểu diễn tập các luật điều khiển dưới dạng ma trận.

		DET						
		NB	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
	NB				*NB*			
	NM				*NM*			
	NS				*NS*			
ET	*ZE*	*NB*	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
	PS				*PS*			
	PM				*PM*			
	PB				*PB*			

Để phát triển thêm, có thể chọn hệ số an toàn cho từng luật điều khiển, tức là khi thiết lập luật hợp thành chung

$$R = R_1 \vee R_2 \vee \cdots \vee R_{13} \tag{3.4}$$

không phải tất cả các luật điều khiển R_k , k=1,2...,13 được tham gia một cách bình đẳng mà theo một hệ số an toàn định trước. Ngoài những hệ số an toàn cho từng luật điều khiển còn có hệ số an toàn cho từng mệnh đề điều kiện của một luật điều khiển khi số các mệnh đề của nó nhiều hơn 1. Song trong trường hợp xét ở đây không cần đến cả hai loại hệ số an toàn đó.

3.3.4 Chọn thiết bị hợp thành

Có thể chọn thiết bị hợp thành theo những nguyên tắc đã giới thiệu trong chương 1 và chương 2 để triển khai phép hoặc trong (3.4) bao gồm

- sử dụng công thức (1.10) có luật max–MIN, max–PROD,
- sử dụng công thức *Lukasiewics* có luật sum–MIN, sum–PROD,
- sử dụng tổng Einstein,
- sử dụng tổng trực tiếp,
- ...,

3.3.5 Chọn nguyên lý giải mờ

Các phương pháp xác định giá trị đầu ra rõ, hay con gọi là quá trình giải mờ hoặc rõ hóa đã được trình bày trong chương 1. Phương pháp giải mờ được chọn cũng gây ảnh hưởng đến độ phức tạp và trạng thái làm việc của toàn bộ hệ thống. Thường trong thiết kế hệ thống điều khiển mờ, giải mờ bằng phương pháp điểm trọng tâm có nhiều ưu điểm hơn cả, bởi vì như vậy trong kết quả đều có sự tham gia của tất cả kết luận của các luật điều khiển R_k , k=1,2,...,13 (mệnh đề hợp thành).

3.3.6 Tối ưu

Sau khi bộ điều khiển mờ đã được tổng hợp, có thể ghép nối nó với đối tượng điều khiển thực hoặc với một đối tượng mô phỏng để thử nghiệm. Trong quá trình thử nghiệm cần đặc biệt kiểm tra xem có tồn tại "lỗ hổng" nào trong quá trình làm việc hay không, tức là phải xác định xem tập các luật điều khiển được xây dựng có *đầy đủ* hay không để khắc phục. Nguyên nhân của hiện tượng "lỗ hổng" có thể do việc thiết lập các nguyên tắc điều khiển chung quanh điểm làm việc không phủ lên nhau hoàn toàn, hoặc là có một số kết quả sai trong các nguyên tắc điều khiển được thiết lập. Một nguyên nhân nữa có thể xảy ra là bộ điều khiển làm việc không ổn định, vì nó nằm quá xa điểm làm việc. Trong mọi trường hợp trước hết nên xem xét lại các luật điều khiển cơ sở.

Sau khi đã đảm bảo được bộ điều khiển làm việc ổn định và không có các "lỗ hổng", bước tiếp theo là tối ưu trạng thái làm việc của nó theo các chỉ tiêu khác nhau. Chỉnh định bộ điều khiển theo các chỉ tiêu này chủ yếu được thực hiện thông qua việc hiệu chỉnh hàm thuộc và thiết lập thêm các nguyên tắc điều khiển bổ sung hoặc sửa đổi lại các nguyên tắc điều khiển đã có. Việc chỉnh định sẽ rất có kết quả nếu như được thực hiện trên một hệ kín.

Khi xử lý các kết quả chỉnh định cần đặc biệt để ý khi các hệ thống không phụ thuộc vào thời gian hoặc các hệ thống có hằng số thời gian trễ T_t lớn. Những tính chất này của hệ sẽ làm cho các biến đổi khi chỉnh định thường khó nhận biết. Trong các trường hợp đó tốt hơn là nên thực hiện từng bước và ghi lại biên bản cho mọi trường hợp.

3.4 Các bộ điều khiển mờ

Phần này sẽ trình bày các nguyên tắc cơ bản để thiết kế các bộ điều khiển mờ và phân loại các bộ điều khiển mờ theo tính chất của nó. Đặc biệt là khi muốn nghiên cứu các phương pháp chuyển đổi các bộ điều khiển P (hay PI) kinh điển thành các bộ điều khiển mờ. Điều đó giúp cho việc xâm nhập vào kỹ thuật điều khiển mờ dễ dàng hơn.

Khi tổng hợp các bộ điều khiển mờ cần lưu ý đến tính phi tuyến khá mạnh của bộ điều khiển mờ. Phần lớn các đối tượng được điều khiển trong thực tế có tính phi tuyến, phụ thuộc vào thời gian, có hằng số thời gian trễ lớn và có tham số rải. Các hệ thống như vậy thường gặp trong công nghệ chế biến. Đối với các hệ thống như vậy thì việc điều khiển bằng kỹ thuật mờ rất thích hợp. Tuy vậy nền tảng cho sự ứng dụng thành công kỹ thuật điều khiển mờ là kiến thức, kinh nghiệm và sự hiểu biết của các chuyên gia về hệ thống đó, người đã từng trực tiếp điều khiển hệ thống bằng tay.

Phần này không đề cập đến các vấn đề đặt biệt như vậy. Các phương pháp tổng hợp bộ điều khiển sau đây được trình bày cho một số bộ điều khiển và một số đối tượng nhất định. Do tính chất phi tuyến của đối tượng cũng như của bộ điều khiển nên không thể đưa ra được các nguyên tắc tổng hợp tổng quát cho các đối tượng như trong hệ tuyến tính, vì các tính chất của hệ tuyến tính có thể nhận được qua đáp ứng của hệ thống với tín hiệu vào là hàm bậc thang đơn vị, nhưng với hệ phi tuyến thì không.

1 Phương pháp tổng hợp kinh điển

Trước khi đi vào việc phân tích và tổng hợp các bộ điều khiển mờ, cũng cần lược qua một cách ngắn gọn các phương pháp tổng hợp kinh điển, vì đứng trên một phương diện nào đó điều này cũng thật là thú vị. Phương pháp tổng hợp kinh điển - điều mà ở đây không làm cũ đi khái niệm đó - bao gồm các bước:

1) Xây dựng mô hình đối tượng đủ chính xác.
2) Đơn giản hóa mô hình.
3) Tuyến tính hóa mô hình tại điểm làm việc.
4) Chọn bộ điều khiển thích hợp, ví dụ như bộ điều khiển P, PID, bộ điều khiển trạng thái ... và xác định các tính chất mà bộ điều khiển phải có.
5) Tính toán thông số của bộ điều khiển. Để thực hiện việc xác định thông số của bộ điều khiển có rất nhiều các phương pháp như phương pháp đường đặc tính tần số với tiêu chuẩn Nyquist hay phương pháp quĩ đạo nghiệm số. Bằng các phương pháp này sẽ tổng hợp được bộ điều khiển ổn định, nhưng không tổng hợp được các bộ điều khiển có đặc tính động xác định. Các phương pháp tối ưu xác định thông số của bộ điều khiển thường phải làm với nhiều chỉ tiêu mâu thuẫn với nhau như ổn định nhưng thời gian quá trình quá độ phải ngắn và độ quá điều chỉnh cực đại phải nhỏ ...
6) Kiểm tra bộ điều khiển vừa thiết kế bằng cách ghép nối với mô hình đối tượng điều khiển, nếu kết quả không được như mong muốn, phải thiết kế lại theo các bước từ 2 đến 6 cho đến khi đạt được kết quả mong muốn.
7) Đưa bộ điều khiển vừa thiết kế vào điều khiển đối tượng thực và kiểm tra quá trình làm việc của hệ thống. Nếu chưa đạt yêu cầu thiết kế lại bộ điều khiển theo các bước từ 1 đến 6 cho đến khi đạt được các chỉ tiêu chất lượng mong muốn.

Tổng hợp một bộ điều khiển với các chức năng hoàn hảo phụ thuộc rất nhiều vào các nhà chuyên môn. Quá trình tổng hợp sẽ rút ngắn lại vì chỉ còn phải thực hiện các bước 4 và 5 nếu đã có mô hình đối tượng. Xây dựng một mô hình đối tượng hữu ích là một đòi hỏi rất khó thực hiện, vì bên cạnh những hiểu biết tốt về lý thuyết, còn đòi hỏi rất nhiều vào kinh nghiệm trong việc nhận dạng hệ thống mà chủ yếu là dựa vào kinh nghiệm cũng như sự hiểu biết đối tượng.

Nhìn chung, phương pháp tổng hợp kinh điển thường gặp những khó khăn do việc phải xây dựng được mô hình đối tượng trước khi thiết kế các bộ điều khiển. Mặt khác các bộ điều khiển phải được thiết kế dựa trên cơ sở kỹ thuật và đảm bảo tính chất phù hợp đối tượng của các bộ điều khiển này.

Một hệ thống điều khiển mờ, được minh họa trong *hình 3.3*, bao gồm một bộ điều khiển mờ, thiết bị cảm biến và đối tượng điều khiển. Cấu trúc một mạch điều khiển như vậy cũng giống như những mạch vòng điều khiển mà từ trước đến nay đã rất quen biết, sự khác nhau ở đây là ở vị trí các bộ điều khiển P, PID quen thuộc là bộ điều khiển mờ. Trường hợp đơn giản nhất là tín hiệu ra của bộ cảm biến được đưa qua một bộ biến đổi A/D và tín hiệu ra của bộ điều khiển mờ được đưa qua một bộ biến đổi D/A qua khuếch đại và đưa về điều khiển đối tượng.

Các bước cơ bản để tổng hợp một bộ điều khiển mờ đã được giới thiệu trong mục 3.3. Trong những mục tiếp theo sẽ giới thiệu và phân tích chi tiết các bộ điều khiển mờ.

3.4.2 Mô hình đối tượng điều khiển

Để thấy được toàn bộ hoạt động của hệ thống điều khiển, một mô hình đối tượng sẽ được mô phỏng ở đây. Quá trình làm mát và sưởi ấm cho một căn phòng được chọn như một mô hình khâu quán tính bậc nhất. Giả thiết công suất làm mát (hay sưởi ấm) $1kW$ thay đổi được nhiệt độ phòng cỡ 10^oC và hằng số thời gian quán tính bằng $1000s$. Do đó hệ số khuếch đại sẽ là K=10/1000 C/W = 0,01 C/W và hằng số thời gian T_1.

Mô hình đối tượng biểu diễn dưới dạng phương trình vi phân tuyến tính như sau:

$$T_1 \frac{dy}{dt} + y(t) = K \cdot x(t). \tag{3.5}$$

Thật ra mô hình (3.5) chỉ mang tính chất minh họa cho thí dụ sau, còn trong thực tế khi thiết kế hệ điều khiển mờ không nhất thiết phải biết trước mô hình mà chỉ cần thể hiện những hiểu biết về đối tượng qua các biến ngôn ngữ về động học của đối tượng, những biến này lại được phản chiếu qua các biến ngôn ngữ và các nguyên tắc điều khiển cơ sở của bộ điều khiển mờ. Trong nhiều trường hợp, khả năng nhận dạng đối tượng qua mô hình cực kỳ khó khăn và nhiều trường hợp không thể thực hiện được, nên việc tổng hợp hệ thống điều khiển bằng thiết kế bộ điều khiển mờ cho phép tiết kiệm rất nhiều công sức và giá thành lại rẻ. *Đó là điểm mạnh của điều khiển mờ trong việc thiết kế các hệ thống điều khiển các đối tượng phức tạp*, các đối tượng mà việc xây dựng mô hình cực kỳ khó khăn. Ngay cả đối với các đối tượng điều khiển đơn giản qui trình thiết kế hệ thống mờ cũng ngắn hơn so với qui trình thiết các hệ thống điều khiển kinh điển.

Những bước thiết kế nhiều công nhất trong việc tổng hợp bộ điều khiển mờ là bước 3.3.4 (xây dựng các luật điều khiển)và 3.3.5 (chọn thiết bị hợp thành), nhất là cho các bộ điều khiển mờ có nhiều đầu vào và nhiều tập mờ, vì số lượng các nguyên tắc điều khiển có thể có rất lớn và không phải lúc nào cũng dễ dàng lập ra các nguyên tắc đó vì nó đòi hỏi phải có hiểu biết kinh nghiệm để có thể tổng hợp được hệ thống.

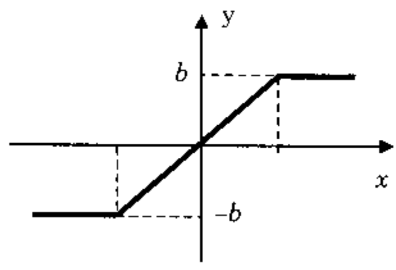
3.4.3 Bộ điều khiển mờ tĩnh

Các bộ điều khiển tĩnh là những bộ điều khiển có quan hệ vào/ra $y(x)$, trong đó x là đầu vào và y là đầu ra, theo dạng một phương trình đại số (tuyến tính hoặc phi tuyến). Các bộ điều khiển tĩnh điển hình là bộ khuếch đại P, bộ điều khiển relay hai vị trí, ba vị trí ....

Những bộ điều khiển tĩnh này rất hay gặp trong các hệ thống điều khiển tự động được thiết kế theo phương pháp kinh điển, nhất là các bộ điều khiển P, và bộ điều khiển hai vị trí. Thiết kế và chỉnh định các bộ điều khiển này đơn giản, nhưng khi sử dụng các bộ điều khiển này trong hệ thống hệ thống điều khiển thì thường không đạt được chất lượng điều khiển tốt. Mặc dù vậy trong thực tế các bộ điều khển này vẫn được dùng rất nhiều, bởi vì chúng tương đối đơn giản, robust (bền vững) và không cần phải chọn nhiều thông số tối ưu.

Bộ điều khiển mờ đơn giản theo luật tỷ lệ là một bộ điều khiển có một đầu vào, một đầu ra và tín hiệu ra của bộ điều khiển mờ luôn tỷ lệ với sự biến đổi của tín hiệu đầu vào cho tới khi nó đạt được *giá trị bão hòa.*

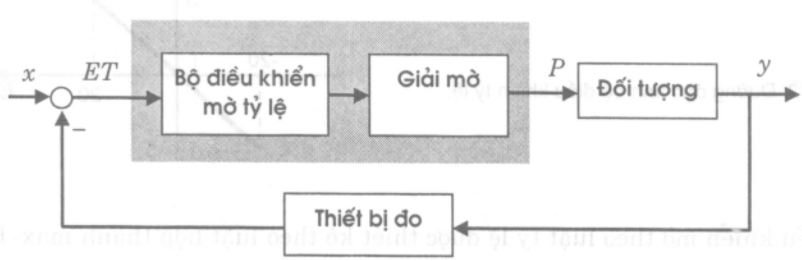
Hình 3.10: Quan hệ truyền đạt bộ điều khiển mờ theo luật tỷ lệ.



Định nghĩa như trên cho phép bộ điều khiển có đặc tính tuyến tính hoặc tuyến tính từng đoạn đều được xếp vào nhóm các bộ điều khiển mờ theo luật tỷ lệ, có nghĩa là cả bộ điều khiển mờ hai vị trí. Điều đó cũng chứng tỏ rằng một bộ điều khiển mờ theo luật tỷ lệ có độ tự do cao hơn bộ điều khiển theo luật tỷ lệ kinh điển, vì bộ điều theo luật tỷ lệ chỉ có một thông số thay đổi đó là hệ số khuếch đại K. Một bộ điều khiển mờ theo luật tỷ lệ có độ tự do rất lớn. Ví dụ như đầu vào của một bộ điều khiển mờ theo luật tỷ lệ bao gồm 5 tập mờ có hàm thuộc dạng hình tam giác và đầu ra bao gồm 5 singleton, thì số lượng tham số của bộ điều khiển mờ sẽ bằng $5 \cdot 3+5=20$, những thông số này sẽ xác định các hàm thuộc. Ngoài ra còn có các độ tự do khác trong việc chọn thuật toán, phương pháp xác định luật hợp thành và phương pháp giải mờ.

Với những độ tự do nhiều như vậy và với trạng thái truyền đạt phi tuyến thì việc phân tích hệ thống bằng các phương pháp toán học kinh điển gặp rất nhiều khó khăn (đặt biệt là việc khảo sát tính ổn định cho hệ thống). Các vấn đề gặp phải trong việc thiết kế bộ điều khiển mờ sẽ tăng theo qui luật lũy thừa khi số lượng đầu vào và đầu ra tăng lên. Điều này hoàn toàn tương phản với việc người ta có thể thiết kế một bộ điều khiển mờ có chức năng hoàn hảo trong một thời gian tương đối ngắn. Nhưng ngược lại, với độ tự do lớn như vậy, việc thiết kế bộ điều khiển mờ để điều khiển các đối tượng phức tạp sẽ đơn giản hơn nhiều so với các phương pháp thiết kế các bộ điều khiển thông thường.

Thiết kế một bộ điều khiển mờ chỉ có thể thực hiện được nếu như chuyển được những kinh nghiệm và hiểu biết về hệ thống thành các luật điều khiển. Trong trường hợp việc chuyển đổi đó không thực hiện được ngay, việc thiết kế vẫn có thể được tiến hành theo các phương pháp học như Neuro-Fuzzy-Logic hoặc mạng Neuron, nhưng những phương pháp "*tự học*" này đều đòi hỏi hoặc là bộ điều khiển đã biết trước hoặc là nó sẽ tự đi tìm và xây dựng mô hình nghịch đảo của đối tượng. Bởi vậy cũng không nên trông đợi nhiều vào những phương pháp này vì nhận dạng hệ phi tuyến rất khó khăn.



Hình 3.11: Điều khiển nhiệt độ bằng bộ điều khiển mờ tỷ lệ.

Việc tổng hợp bộ điều khiển mờ tỷ lệ cho hệ thống điều khiển nhiệt độ như *hình 3.11* mô tả được bắt đầu bằng định nghĩa các biến ngôn ngữ vào ra. Đầu vào của bộ điều khiển là sai lệch giữa tín hiệu chủ đạo và tín hiệu ra của bộ cảm biến, ký hiệu là ET. Phạm vi giá trị của biến ngôn ngữ này nằm trong khoảng từ -20^0C đến 20^0C. Đầu ra của bộ điều khiển là công suất làm mát hay sưởi ấm P với miền giá trị vật lý là từ - $5kW$ đến +5 kW.

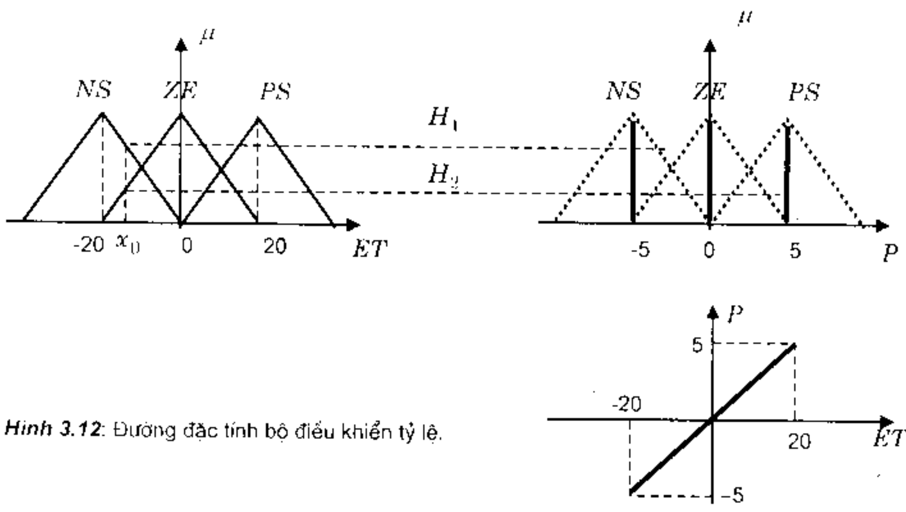
Cho biến vào và biến ra 3 giá trị mờ NS, ZE, PS. Cả 3 giá trị mờ của biến vào ET có hàm thuộc dạng hình tam giác cân và đầu ra có dạng singleton (xem *hình 3.12*). Luật hợp thành gồm 3 luật điều khiển

R_1: NẾU $ET = NS$ THÌ $P = NS$ HOẶC

R_2: NẾU $ET = ZE$ THÌ $P = ZE$ HOẶC

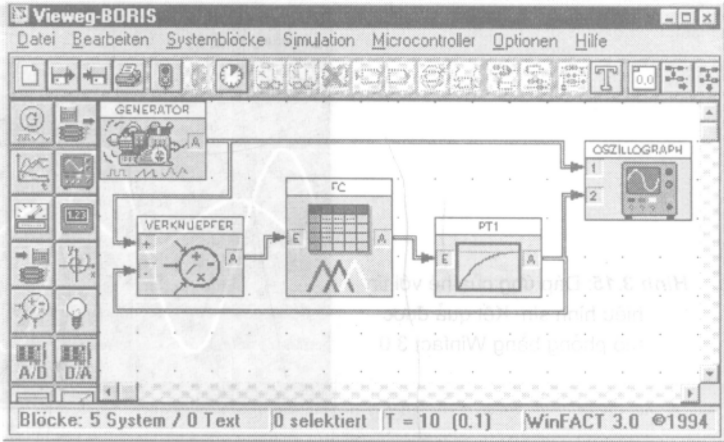
R_3: NẾU $ET = PS$ THÌ $P = PS$.

Để có thể xử lý được tín hiệu bằng bộ điều khiển mờ cần thiết phải chọn được thời gian cắt mẫu phù hợp. Đây là một quá trình biến đổi chậm nên có thể chọn thời gian cắt mẫu $T_a = 10s$.



Hình 3.12: Đường đặc tính bộ điều khiển tỷ lệ.

Bộ điều khiển mờ theo luật tỷ lệ được thiết kế theo luật hợp thành max–PROD và giải mờ theo phương pháp độ cao. Như đã phân tích ở phần 2.1.2, bộ điều khiển này có hàm truyền đạt tuyến tính trong khoảng [–20 , 20].

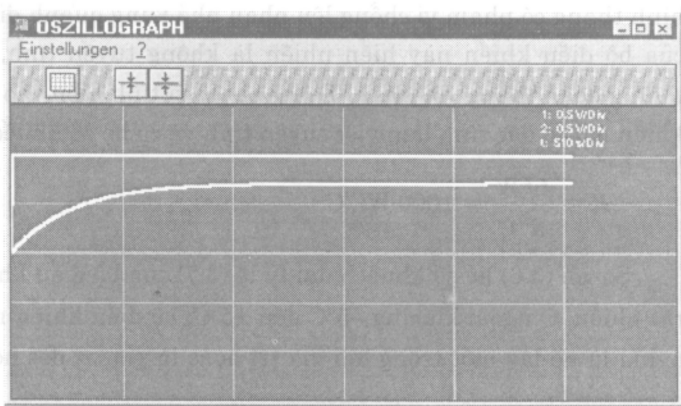


Hình 3.13: Điều khiển nhiệt độ bằng bộ điều khiển mờ tỷ lệ.

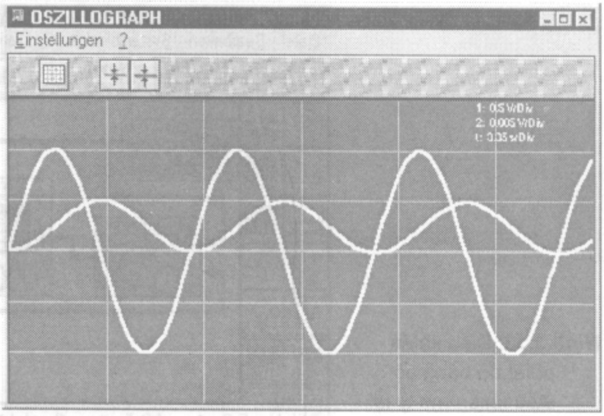
Hệ số khuếch đại của bộ điều khiển mờ là:

$$K = \frac{5kW}{20^0C} = 250\ W/{}^0C. \tag{3.6}$$

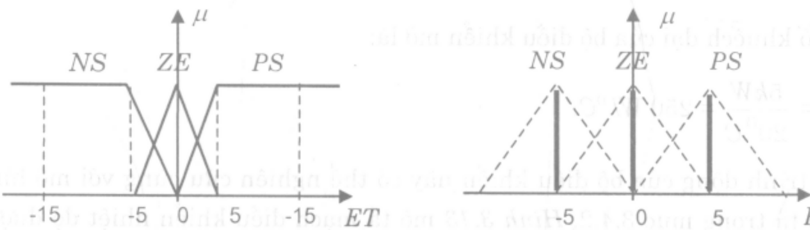
Quá trình động của bộ điều khiển này có thể nghiên cứu cùng với mô hình đối tượng mô tả trong mục 3.4.2. *Hình 3.13* mô tả mạch điều khiển nhiệt độ được tổng hợp nhờ chương trình *Winfact* 3.0 (*Windows Fuzzy Control Tools*). *Hình 3.14* biểu diễn đáp ứng của hệ thống điều khiển nhiệt độ đối với tín hiệu vào là hàm $1(t)$. Bước nhảy của hàm $1(t)$ tương ứng với 10^0C. Sai số tĩnh của toàn bộ hệ thống cỡ $2{,}5^0C$. Sai số này rất lớn do việc chọn hệ số khuếch đại của bộ điều khiển quá nhỏ. Nếu quan sát đáp ứng của hệ thống đối với tín hiệu chủ đạo là tín hiệu điều hòa (*hình 3.15*) sẽ xác định được rằng góc lệch pha giữa tín hiệu cần đạt được và tín hiệu ra của hệ thống quá lớn.



Hình 3.14: Đáp ứng của hệ với tín hiệu $1(t)$. Kết quả mô phỏng bằng Winfact 3.0



Hình 3.15: Đáp ứng của hệ với tín hiệu hình sin. Kết quả được mô phỏng bằng Winfact 3.0

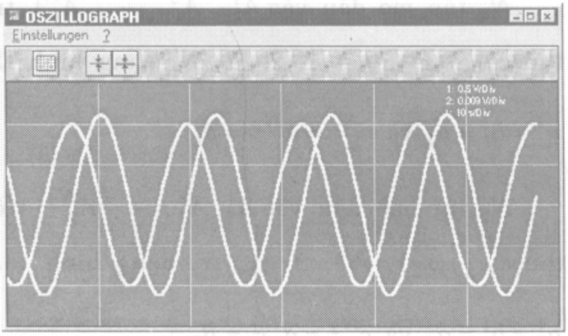


Hình 3.16: Thay đổi tập mờ đầu vào để tăng hệ số tỷ lệ K.

Để giảm sai lệch tĩnh của hệ thống, về nguyên tắc hoàn toàn giống như cho hệ tuyến tính, có thể thực hiện bằng cách tăng hệ số khuếch đại của bộ điều khiển mờ theo luật tỷ lệ. Để làm được việc đó cần thay đổi lại các tập mờ thuộc biến ngôn ngữ *ET*. *Hình 3.16* cho thấy hàm thuộc cho các giá trị *PS* và *NS* được chọn theo dạng hình thang có phạm vi chồng lên nhau nhỏ xung quanh điểm 0. Quan hệ truyền đạt của bộ điều khiển này hiển nhiên là không tuyến tính, nhưng được ghép nối từ những đoạn thẳng với nhau. Trong phạm vi tính hiệu vào từ -5^0C đến $+5^0C$ bộ điều khiển mờ có đặc tính làm việc tuyến tính và có hệ số khuếch đại tỷ lệ bằng:

$$K = \frac{5kW}{5^0C} = 1000\ W/{}^0C. \tag{3.7}$$

So với (3.6) hệ số khuếch đại tỷ lệ (3.7) của bộ điều khiển mờ lúc này đã lớn hơn rất nhiều. Ở ngoài khoảng -5^0C đến $+5^0C$, bộ điều khiển mờ có đặc tính bão hòa, có nghĩa là nó lấy một trong hai giá trị hoặc là giá trị đốt nóng cực đại hoặc là giá trị làm lạnh cực đại. *Hình 3.17* cho thấy chất lượng của hệ thống khi làm việc với bộ điều khiển này tốt hơn bộ điều khiển được thiết kế trước đó.



Hình 3.17: Đáp ứng của hệ với tín hiệu hình sin sau khi tăng hệ số khuếch đại K lên 4 lần.

3.4.4 Thuật toán tổng hợp một bộ điều khiển mờ tĩnh

Phần trên đã trình bày phương pháp tổng hợp và chỉnh định một bộ điều khiển mờ tỷ lệ. Không nhất thiết phải định nghĩa lại các tập mờ đầu vào mới có thể thay đổi được hệ số khuếch đại K như đã làm, tuy nhiên ở bài toán đơn giản trên thì phương pháp đó là thích hợp và dễ chỉnh định nhất.

Phần này sẽ trình bày một thuật toán tổng hợp một bộ điều khiển mờ có quan hệ truyền đạt cho trước.

Bài toán đặt ra

Giả sử X là một tập compact trong $\mathbb{R}^2$ có dạng: $X = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2]$. Cho trước hàm hai biến $g(\underline{x})$, $\underline{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ có miền xác định là X. Hãy tổng hợp một bộ điều khiển mờ tĩnh trên X có đường đặc tính $y(\underline{x})$ của quan hệ truyền đạt "gần giống" đường $g(\underline{x})$ đã cho.

Mặc dù bài toán chỉ xét trên phạm vi bộ điều khiển cần tổng hợp có hai tín hiệu vào là x_1, x_2 và một tín hiệu ra là y, song việc mở rộng sang trường hợp nhiều đầu vào, một đầu ra là hoàn toàn tương đương.

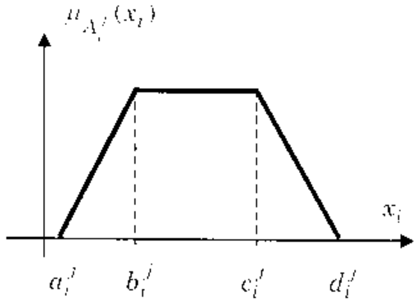
Bài toán đặt ra, trên cơ sở công thức (3.2), luôn có nghiệm.

Thuật toán tổng hợp bộ điều khiển mờ

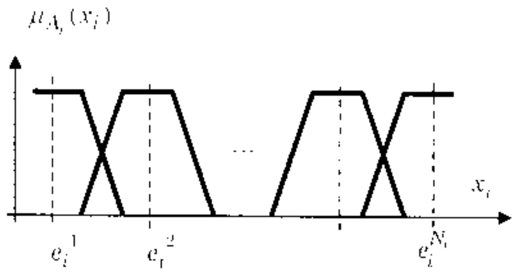
1) *Định nghĩa các tập mờ.*

N_1 tập mờ đầu vào A_1^1, A_1^2, ⋯ , $A_1^{N_1}$ trên khoảng $[\alpha_1, \beta_1]$ của x_1 có hàm thuộc $\mu_{A_1^j}(x_1)$, $j = 1, 2, \ldots, N_1$ dạng hình thang cân cho trong *hình 3.16* với $a_1^1 = b_1^1 = \alpha_1$ và $c_1^{N_1} = d_1^{N_1} = \beta_1$.

N_2 tập mờ đầu vào A_2^1, A_2^2, ⋯ , $A_2^{N_2}$ trên khoảng $[\alpha_2, \beta_2]$ của x_2 có hàm thuộc $\mu_{A_2^j}(x_2)$, j=1, 2, ..., N_2 dạng hình thang cân cho trong *hình 3.18* với $a_2^1 = b_2^1 = \alpha_2$ và $c_2^{N_2} = d_2^{N_2} = \beta_2$.



Hình 3.18: Hàm thuộc của các tập mờ đầu vào với i = 1, 2 và j = 1, 2, ... , N_i



Hình 3.19: Tập các hàm thuộc các tập mờ đầu vào (i = 1, 2).

Ký hiệu các giá trị $e_i^1 = \alpha_i$, $e_i^{N_i} = \beta_i$ và $e_i^j = \dfrac{b_i^j + c_i^j}{2}$ cho i=1,2 và j= 2, 3, ... , $N_i - 1$. *Hình 3.19* biểu diễn các tập mờ đã được định nghĩa.

Các tập mờ đầu ra B_{pq} được định nghĩa dạng singleton (hàm Kronecker) tại điểm

$$y_{pq} = g(\underline{e}_{pq}) \quad \text{với} \quad \underline{e}_{pq} = \begin{pmatrix} e_1^p \\ e_2^q \end{pmatrix}. \tag{3.8}$$

2) *Xây dựng các luật điều khiển.*

Thiết lập tất cả $N_1 \times N_2$ các luật điều khiển theo cấu trúc:

NẾU $\chi_1 = A_1^p$ VÀ $\chi_2 = A_2^q$ THÌ $\gamma = B_{pq}$ với $p = 1, 2, \dots, N_1$ và $q = 1, 2, \dots, N_2$.

3) *Chọn thiết bị hợp thành.*

Chọn nguyên tắc triển khai sum-PROD cho mệnh đề hợp thành, tích đại số cho phép giao và công thức Lukasiewicz cho phép hợp thì tập mờ đầu ra B' khi đầu vào là một giá trị rõ $\underline{x}_0 = \begin{pmatrix} x_{10} \\ x_{20} \end{pmatrix}$ sẽ là

$$\mu_{B'}(y) = \min\left\{1, \sum_{p=1}^{N_1} \sum_{q=1}^{N_2} \left(\mu_{B_{pq}}(y) \mu_{A_1^p}(x_{01}) \mu_{A_2^q}(x_{02}) \right) \right\}. \tag{3.9}$$

Để ý rằng $\mu_{B_{pq}}(y)$ là một hàm Kronecker nên

$$\mu_{B'}(y) = \min\left\{1, \sum_{p=1}^{N_1} \sum_{q=1}^{N_2} \left(\mu_{A_1^p}(x_{01}) \mu_{A_2^q}(x_{02}) \right) \right\}. \tag{3.10}$$

4) *Chọn phương pháp giải mờ.*

Chọn phương pháp độ cao để giải mờ và để ý rằng các hàm thuộc là những hình thang cân nên phép tính lấy MIN trong (3.10) có thể bỏ qua mà không ảnh hưởng tới kết quả, vậy thì từ (2.23) và (3.10) có được

$$y(\underline{x}_0) = \frac{\displaystyle\sum_{p=1}^{N_1} \sum_{q=1}^{N_2} \left(y_{pq} \mu_{A_1^p}(x_{01}) \mu_{A_2^q}(x_{02}) \right)}{\displaystyle\sum_{p=1}^{N_1} \sum_{q=1}^{N_2} \left(\mu_{A_1^p}(x_{01}) \mu_{A_2^q}(x_{02}) \right)}. \tag{3.11}$$

Đường đặc tính của quan hệ truyền đạt bộ điều khiển mờ vừa thiết kế được suy ra từ (3.11) có dạng như sau:

$$y(\underline{x}) = \frac{\displaystyle\sum_{p=1, q=1}^{p=N_1, q=N_2} \left(g(\underline{e}_{pq}) \mu_{A_1^p}(x_1) \mu_{A_2^q}(x_2) \right)}{\displaystyle\sum_{p=1, q=1}^{p=N_1, q=N_2} \left(\mu_{A_1^p}(x_1) \mu_{A_2^q}(x_2) \right)}. \tag{3.12}$$

Sai số

Sai số giữa $g(\underline{x})$ và $y(\underline{x})$ của bộ điều khiển mờ tổng hợp được có công thức cho trong (3.12) sẽ là

$$\|g-y\|_{\infty} \le \left\|\frac{\partial g}{\partial x_1}\right\|_{\infty} h_1 + \left\|\frac{\partial g}{\partial x_2}\right\|_{\infty} h_2 \ , \tag{3.13}$$

trong đó ký hiệu $\|\cdot\|_{\infty}$ được hiểu là chuẩn vô cùng, tức là

$$\|f\|_{\infty} = \sup_{\underline{x}\in X}|f(\underline{x})| \tag{3.14}$$

và $$h_1 = \max_{1\le j<N_1}\left|e_1^{j-1} - e_1^j\right| \ , \qquad h_2 = \max_{1<j<N_2}\left|e_2^{j-1} - e_2^j\right| \ . \tag{3.15}$$

Phần chứng minh cho công thức (3.13) có thể tìm thấy trong tài liệu [17]. Công thức (3.13) chỉ rằng nếu tồn tại $\left\|\frac{\partial g}{\partial x_i}\right\|_{\infty}$, i=1, 2, mà điều này sẽ xảy ra nếu đó là hàm liên tục (trong không gian compact), thì với một ε>0 bất kỳ cho trước, bao giờ cũng tìm được những giá trị e_i^j thích hợp sao cho

$$\|g-y\|_{\infty} < \varepsilon.$$

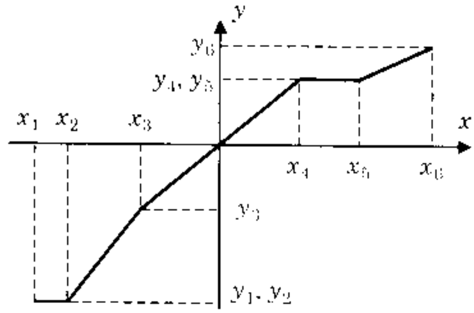
3.4.5 Tổng hợp bộ điều khiển mờ tuyến tính từng đoạn

Thuật toán tổng quát trên cho phép thiết kế một bộ điều khiển mờ có đặc tính $y(\underline{x})$ của quan hệ truyền đạt cho trước với một sai số ε tùy ý. Tuy nhiên để cho sai số càng nhỏ thì số các giá trị mờ của biến ngôn ngữ phải càng nhiều và do đó càng làm tăng thêm độ phức tạp của thiết bị hợp thành, thậm chí số các giá trị mờ phải là vô cùng mới có thể đạt được sai số bằng 0 – xem công thức (3.13) và (3.15).

Quay lại phương pháp tổng hợp bộ điều khiển mờ có đặc tính

$$y = \begin{cases} 5\cdot \mathrm{sgn}(ET) & \text{khi } |x| \ge 20 \\ -Kx & \text{khi } |x| < 20 \end{cases} \tag{3.16}$$

đã trình bày trong mục 3.4.3 thì thấy phương pháp đó hoàn toàn có thể được tổng quát hóa để thiết kế những bộ điều khiển mờ có đường đặc tính $y(x)$ tuyến tính từng khúc với sai số bằng 0 mà không cần phải tăng số các giá trị mờ đến giới hạn vô cùng. Vấn đề đặt ra là phải định nghĩa các hàm thuộc như thế nào và sử dụng nguyên tắc giải mờ gì thì sẽ nối được các đoạn thẳng với nhau một cách liên tục tại những điểm nút.



Hình 3.20: Đường đặc tính $y(x)$ cho trước.

Xét ví dụ đường $y(x)$ cho trong *hình 3.20*. Đường $y(x)$ có 6 cặp điểm nút (x_k, y_k), k=1, 2, ... ,6. Để đường đặc tính của bộ điều khiển sẽ tổng hợp cũng liên tục được tại những điểm nút này thì:

- Mỗi một giá trị rõ đầu vào phải tích cực được 2 luật điều khiển,
- Mỗi hàm thuộc đầu vào có dạng hình tam giác với đỉnh là một điểm nút thứ k và miền xác định là khoảng $[x_{k-1}, x_{k+1}]$ nằm hai phía điểm nút đó,
- Những hàm thuộc đầu ra là những hàm Kronecker định nghĩa tại các điểm nút y_k.
- Sử dụng nguyên tắc độ cao để giải mờ,
- Cài đặt luật hợp thành theo nguyên lý max–MIN.

Hình 3.21 biểu diễn các hàm thuộc vào ra của bộ điều khiển mờ có đường đặc tính $y(x)$ đã cho trong *hình 3.20*.

Thiết bị hợp thành của bộ điều khiển gồm 6 luật:

R_1 : NẾU $\chi = A_1$ THÌ $\gamma = B_1$ HOẶC

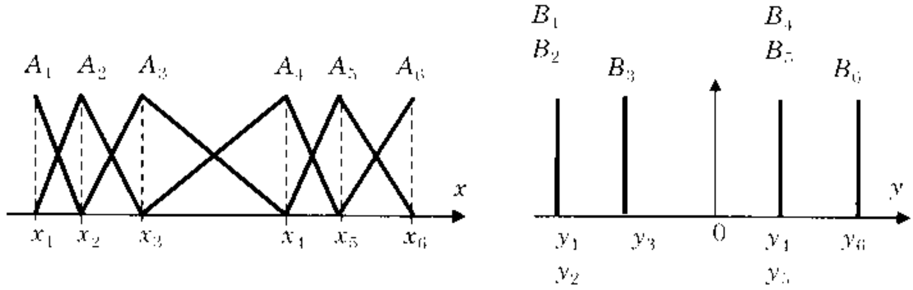
R_2 : NẾU $\chi = A_2$ THÌ $\gamma = B_2$ HOẶC

R_3 : NẾU $\chi = A_3$ THÌ $\gamma = B_3$ HOẶC

R_4 : NẾU $\chi = A_4$ THÌ $\gamma = B_1$ HOẶC

R_5 : NẾU $\chi = A_5$ THÌ $\gamma = B_5$ HOẶC

R_6 : NẾU $\chi = A_6$ THÌ $\gamma = B_6$.



Hình 3.21: Hàm thuộc của các biến ngôn ngữ vào ra.

Tổng quát lên thì thuật toán tổng hợp bộ điều khiển mờ có đường đặc tính $y(x)$ tuyến tính từng khúc cho trước sẽ như sau:

1) Xác định các điểm nút (x_k, y_k), k=1, 2, ... , n của $y(x)$.

2) Định nghĩa n tập mờ đầu vào A_k, k=1, 2, ... , n có hàm thuộc $\mu_{A_k}(x)$ dạng hình tam giác với đỉnh là điểm x_k và miền xác định là khoảng $[x_{k-1}, x_{k+1}]$, trong đó cho B_1 và B_n thì các điểm x_0, x_{n+1} là những điểm bất kỳ thỏa mãn $x_0 < x_1$ và $x_{n+1} > x_n$.

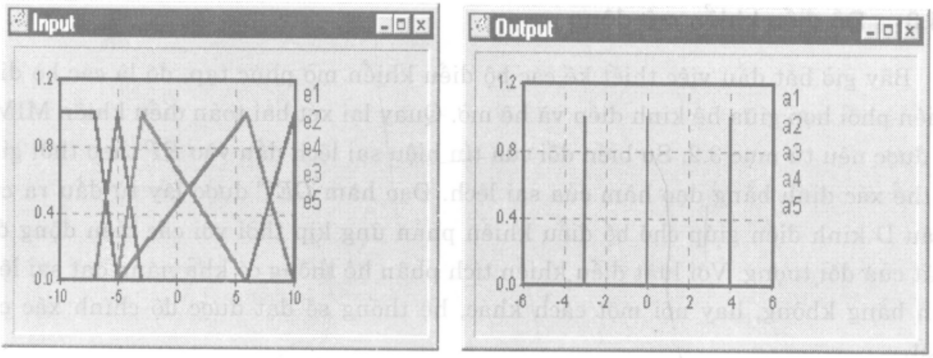
3) Xác định n tập mờ đầu ra B_k, k=1, 2, ... , n có $\mu_{B_k}(y)$ là các hàm Kronecker định nghĩa tại y_k.

4) Định nghĩa tập n luật điều khiển dạng

$$R_k : \text{NẾU } \chi = A_k \text{ THÌ } \gamma = B_k, \quad k=1, 2, \ldots, n.$$

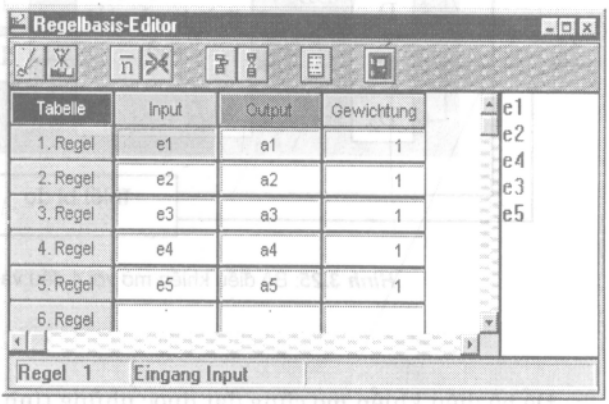
5) Sử dụng nguyên tắc độ cao để giải mờ.

Hình 3.22 trình bày một ví dụ về khai báo tập mờ vào/ra và *hình 3.24* biểu diễn đường đặc tính $y(x)$ quan hệ truyền đạt bộ điều khiển mờ với các tập mờ đó. Bộ điều khiển này được thiết kế bằng chương trình Flot của phần mềm Winfact 3.0 theo thuật toán trên.

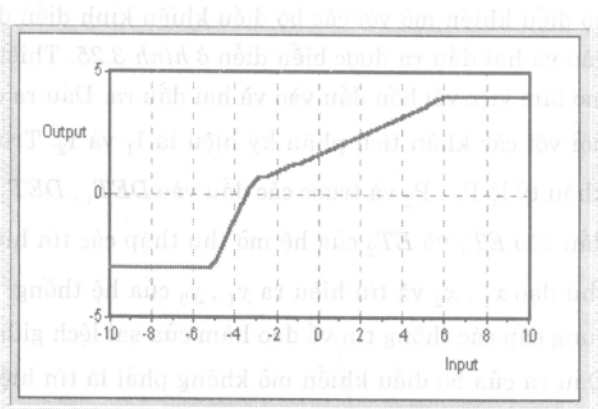


a) b)

Hình 3.22: Xây dựng các tập mờ vào/ra cho bộ điều khiển mờ tuyến tính từng đoạn bằng phần mềm Winfact 3.0.
a) Các tập mờ đầu vào b) Các tập mờ đầu ra



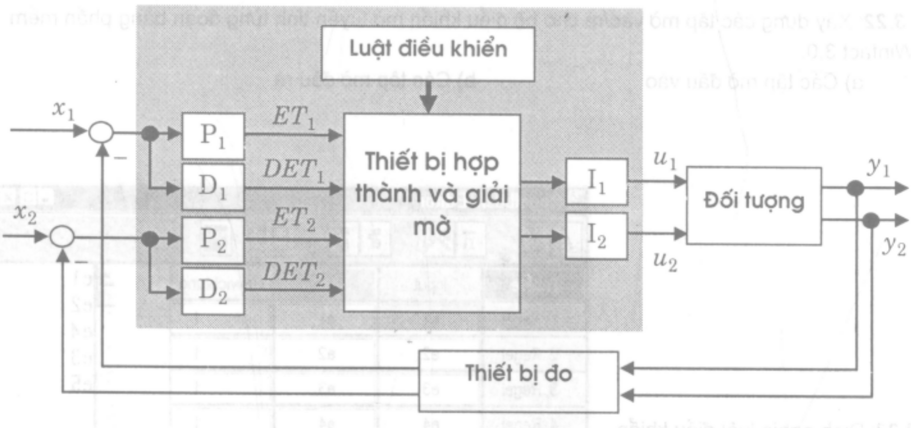
Hình 3.23: Định nghĩa luật điều khiển giữa các tập mờ vào/ra của biến ngôn ngữ có tên là Input và Output trên Winfact 3.0.



Hình 3.24: Đường đặc tính $y(x)$ của quan hệ truyền đạt bộ mờ được thiết kế nhờ phần mềm Winfact 3.0

3.4.6 Bộ điều khiển mờ động

Bây giờ bắt đầu việc thiết kế các bộ điều khiển mờ phức tạp, đó là các bộ điều khiển phối hợp giữa hệ kinh điển và hệ mờ. Quay lại xét bài toán điều khiển MIMO đã được nêu từ mục 3.2. Sự biến đổi của tín hiệu sai lệch đầu vào ET theo thời gian có thể xác định bằng đạo hàm của sai lệch. Đạo hàm DET được lấy từ đầu ra của khâu D kinh điển giúp cho bộ điều khiển phản ứng kịp thời với các biến động đột xuất của đối tượng. Với luật điều khiển tích phân hệ thống có khả năng đạt sai lệch tĩnh bằng không, hay nói một cách khác, hệ thống sẽ đạt được độ chính xác cao nhất.



Hình 3.25: Bộ điều khiển mờ với 4 đầu vào và 2 đầu ra.

Để bộ điều khiển mờ cũng đạt được những tính chất như trên cần phải phối hợp bộ điều khiển mờ với các bộ điều khiển kinh điển đó. Bộ điều khiển mờ với hai đầu vào và hai đầu ra được biểu diễn ở *hình 3.25*. Thiết bị hợp thành của bộ điều khiển mờ làm việc với bốn đầu vào và hai đầu ra. Đầu ra của thiết bị hợp thành được ghép nối với các khâu tích phân ký hiệu là I_1 và I_2. Trước các đầu vào ET_1 , ET_2 là các khâu tỷ lệ P_1 , P_2 và trước các đầu vào DET_1 , DET_2 là các khâu vi phân D_1 , D_2. Các đầu vào ET_1 và ET_2 của hệ mờ thu thập các tín hiệu sai lệch tức thời giữa tín hiệu chủ đạo x_1 , x_2 và tín hiệu ra y_1 , y_2 của hệ thống. Còn các đầu vào DET_1 và DET_2 cung cấp các thông tin về đạo hàm của sai lệch giữa tín hiệu chủ đạo và tín hiệu ra. Đầu ra của bộ điều khiển mờ không phải là tín hiệu điều khiển u_1 và u_2 mà là đạo hàm $\frac{du_1}{dt}, \frac{du_2}{dt}$ của các tín hiệu đó. Chỉ sau khi qua các khâu tích I_1 và I_2 lúc đó mới có được tín hiệu điều khiển u_1 và u_2 cho đối tượng.

Việc giới thiệu phương pháp tổng hợp một bộ điều khiển mờ động sau đây sẽ lần lượt được thực hiện với những bộ điều khiển I, PD, PI và PID.

Bộ điều khiển mờ theo luật PID

Trong kỹ thuật điều khiển kinh điển, bộ điều khiển PID được biết đến như là một giải pháp đa năng và có miền ứng dụng rộng lớn. Định nghĩa về bộ điều khiển theo luật PID kinh điển trước đây vẫn có thể sử dụng cho một bộ điều khiển mờ theo luật PID. Bộ điều khiển mờ theo luật PID được thiết kế theo hai thuật toán:

- thuật toán chỉnh định PID mờ hoặc
- thuật toán PID tốc độ.

Bộ điều khiển mờ được thiết kế theo thuật toán chỉnh định PID có 3 đầu vào gồm sai lệch ET giữa tín hiệu chủ đạo và tín hiệu ra, đạo hàm DET của sai lệch và tích phân IET của sai lệch. Đầu ra của bộ điều khiển mờ chính là tín hiệu điều khiển $u(t)$. Mô hình toán học của bộ PID theo thuật toán chỉnh định có dạng:

$$u(t) = K\left(ET + \frac{1}{T_I}\int_0^t ETdt + T_D\frac{d}{dt}ET\right). \tag{3.16}$$

Với thuật toán PID tốc độ, bộ điều khiển PID có 3 đầu vào: sai lệch ET giữa tín hiệu đầu vào và tín hiệu chủ đạo, đạo hàm bậc nhất DET_1 và đạo hàm bậc hai DET_2 của sai lệch. Đầu ra của hệ mờ là đạo hàm $\frac{du}{dt}$ của tín hiệu điều khiển $u(t)$. Bộ điều khiển PID theo thuật toán tốc độ có mô hình

$$\frac{du}{dt} = K\left(\frac{d}{dt}ET + \frac{1}{T_I}ET + \frac{d^2}{(dt)^2}ET\right). \tag{3.17}$$

Do trong thực tế thường có một hoặc hai thành phần trong (3.16), (3.17) được bỏ qua nên thay vì thiết kế một bộ điều khiển PID hoàn chỉnh người ta lại thường tổng hợp các bộ điều khiển PI với mô hình

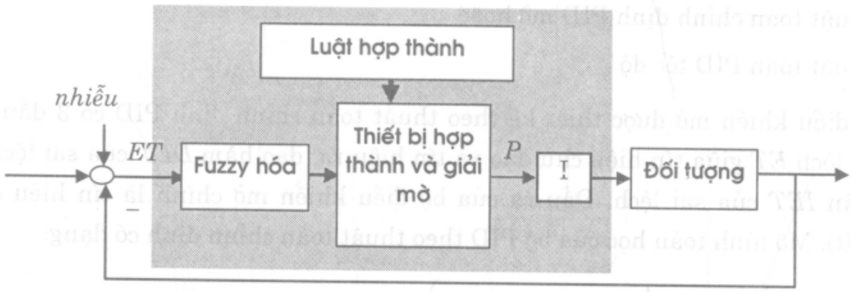
$$u(t) = K\left(ET + \frac{1}{T_I}\int_0^t ETdt\right) \quad \text{hoặc} \quad \frac{du}{dt} = K\left(\frac{d}{dt}ET + \frac{1}{T_I}ET\right) \tag{3.18}$$

hay là bộ điều khiển PD với mô hình

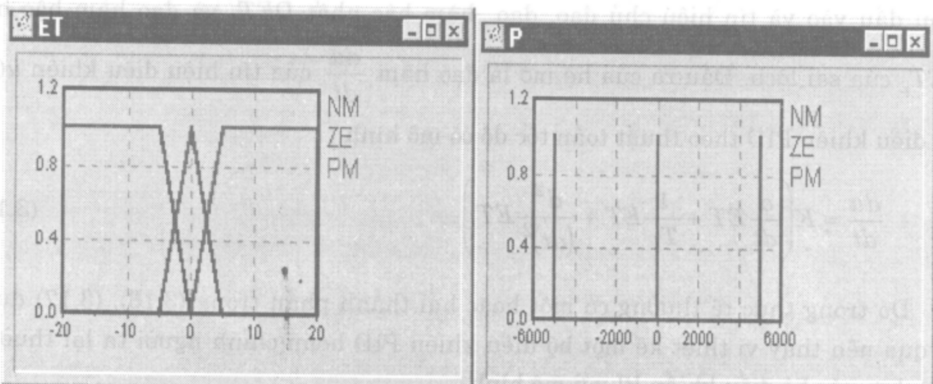
$$u(t) = K\left(ET + T_D \frac{d}{dt}ET\right) \quad \text{hoặc} \quad \frac{du}{dt} = K\left(\frac{d}{dt}ET + \frac{d^2}{(dt)^2}ET\right). \tag{3.19}$$

Bộ điều khiển mờ theo luật I

Một bộ điều khiển mờ theo luật I có thể thiết kế từ một bộ điều khiển mờ theo luật P (bộ điều khiển mờ tuyến tính) bằng cách mắc nối tiếp một khâu tích phân kinh điển vào trước hoặc sau khối mờ đó. Do tính phi tuyến của hệ mờ, nên việc mắc khâu tích phân trước hay sau hệ mờ hoàn toàn khác nhau. Ở phần này khâu tích phân sẽ được mắc ở đầu ra của hệ mờ.

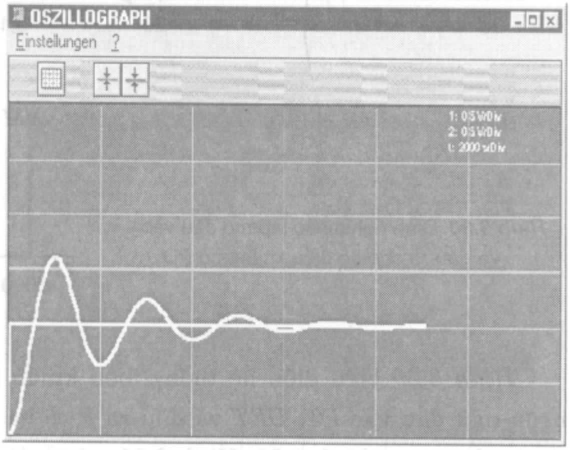


Hình 3.26: Hệ thống điều khiển nhiệt độ bằng bộ điều khiển mờ I. Đối tượng là một khâu quán tính bậc nhất



Hình 3.27: Định nghĩa các giá trị mờ vào ra cho hai biến ngôn ngữ ET và P.

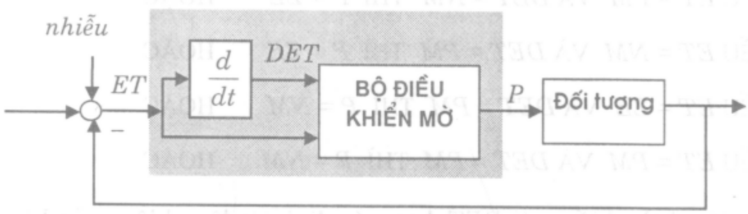
Hình 3.26 mô tả hệ thống điều khiển nhiệt độ bằng bộ mờ I. Các thông số của biến ngôn ngữ được biểu diễn trong *hình 3.27*. Đáp ứng của hệ thống với tín hiệu vào là hàm bậc thang đơn vị có lẫn nhiễu (dạng phân bố chuẩn) được biểu diễn trong *hình 3.28*. Kết quả này trong kỹ thuật điều khiển hoàn toàn bình thường, vì một bộ điều khiển theo luật I không thực sự thích hợp với đối tượng là một khâu quán tính bậc nhất.



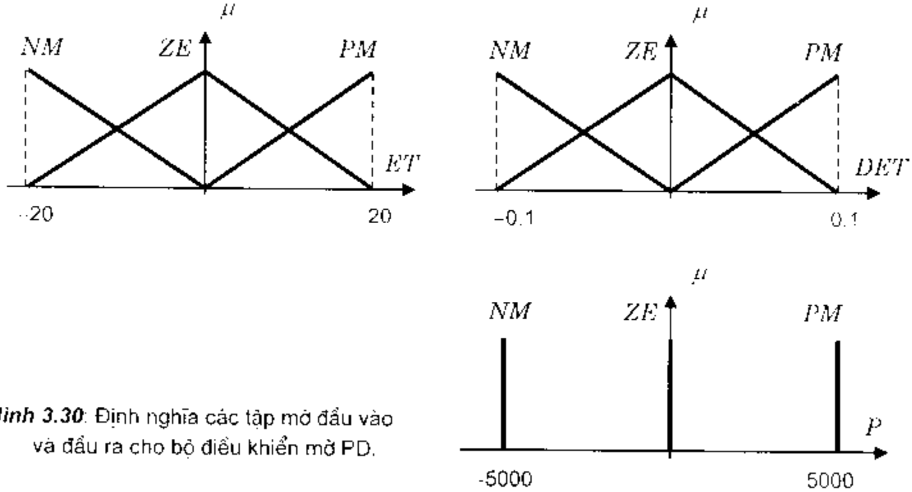
Hình 3.28: Đáp ứng của hệ thống điều khiển nhiệt độ bằng bộ mờ I khi đầu vào là hàm bậc thang $1(t)$ có lẫn nhiễu phân bố chuẩn.

Bộ điều khiển mờ theo luật PD

Theo (3.19) thì khi mắc nối tiếp ở đầu vào của một bộ điều khiển mờ theo luật tỷ lệ một khâu vi phân sẽ có được một bộ điều khiển mờ theo luật PD. Thành phần của bộ điều khiển này cũng giống như bộ điều khiển theo luật PD thông thường bao gồm sai lệch giữa tín hiệu chủ đạo và tín hiệu ra của hệ thống ET và đạo hàm của sai lệch DET. Thành phần vi phân giúp cho hệ thống phản ứng chính xác hơn với những biến đổi lớn của sai lệch theo thời gian. Phát triển tiếp từ ví dụ về bộ điều khiển mờ theo luật P thành bộ điều khiển mờ theo luật PD hoàn toàn đơn giản.



Hình 3.29: Hệ thống điều khiển nhiệt độ bằng bộ điều khiển mờ PD. Đối tượng là một khâu quán tính bậc nhất



Hình 3.30: Định nghĩa các tập mờ đầu vào và đầu ra cho bộ điều khiển mờ PD.

Hình 3.29 biểu diễn hệ thống điều khiển nhiệt độ bằng bộ mờ PD. Mỗi biến ngôn ngữ đầu vào *ET*, *DET* và đầu ra *P* có ba giá trị mờ *NM*, *ZE* và *PM* với hàm thuộc cho trong *hình 3.30*. *Hình 3.31* mô tả đáp ứng của hệ thống với tín hiệu $1(t)$ và tín hiệu dãy xung vuông tại đầu vào. Thiết bị hợp thành của bộ mờ thực hiện các luật điều khiển sau:

R_1 : NẾU $ET = NM$ VÀ $DET = ZE$ THÌ $P = PM$ HOẶC

R_2 : NẾU $ET = ZE$ VÀ $DET = ZE$ THÌ $P = ZE$ HOẶC

R_3 : NẾU $ET = PM$ VÀ $DET = ZE$ THÌ $P = NM$ HOẶC

R_4 : NẾU $ET = NM$ VÀ $DET = NM$ THÌ $P = PM$ HOẶC

R_5 : NẾU $ET = ZE$ VÀ $DET = NM$ THÌ $P = PM$ HOẶC

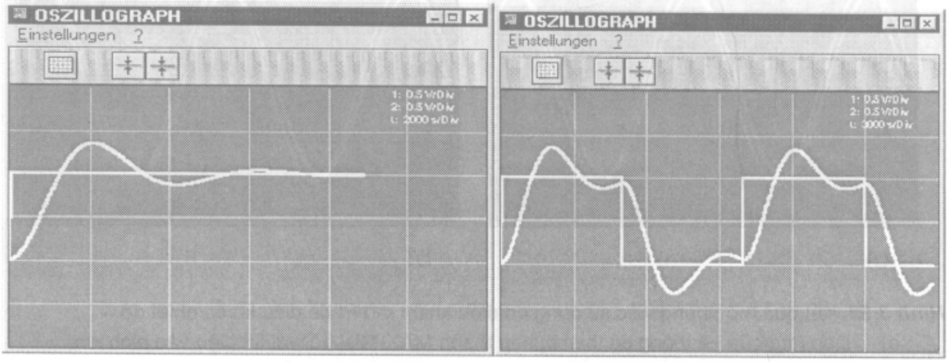
R_6 : NẾU $ET = PM$ VÀ $DET = NM$ THÌ $P = ZE$ HOẶC

R_7 : NẾU $ET = NM$ VÀ $DET = PM$ THÌ $P = ZE$ HOẶC

R_8 : NẾU $ET = ZE$ VÀ $DET = PM$ THÌ $P = NM$ HOẶC

R_9 : NẾU $ET = PM$ VÀ $DET = PM$ THÌ $P = NM$ HOẶC

Phạm vi giá trị cho biến mới *DET* được xác định từ độ nghiêng của hàm quá độ của hệ thống có bộ điều khiển mờ theo luật P. Nhận thấy từ *hình 3.31*, các đáp ứng này không khác gì so với hệ thống sử dụng bộ điều khiển mờ theo luật tỷ lệ, duy chỉ có độ lệch pha của đáp ứng khi tín hiệu vào là tín hiệu hình dãy xung là nhỏ hơn và có thể coi như bằng 0, giá trị cực đại và cực tiểu của đáp ứng gần trùng với tín hiệu chủ đạo. Đó là những ưu điểm mà bộ điều khiển mờ theo luật P không thể có được.



a) b)

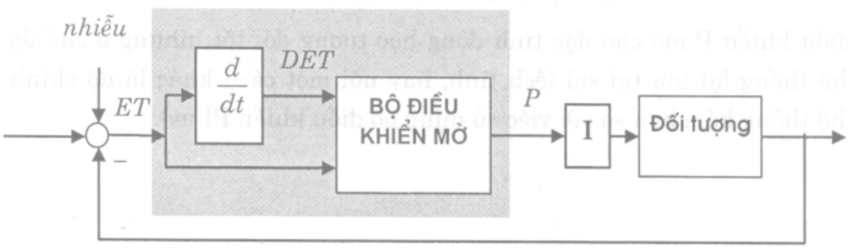
Hình 3.31: Đáp ứng của hệ thống điều khiển nhiệt độ bằng bộ điều khiển mờ PD với các tín hiệu đầu vào khác nhau

a) Đáp ứng của hệ thống với tín hiệu đầu vào $1(t)$.

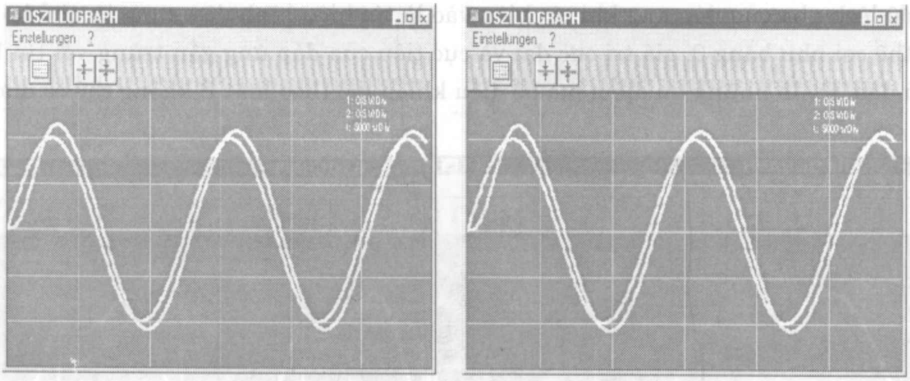
b) Đáp ứng của hệ thống với tín hiệu đầu vào là dãy xung vuông.

Bộ điều khiển mờ theo luật PI

Bộ điều khiển mờ theo luật PI thường được sử dụng để triệt tiêu sai lệch tĩnh của hệ thống. Theo (3.18) thì bộ điều khiển PI sẽ mờ được thiết kế trên cơ sở của bộ điều khiển PD mờ, bằng cách mắc nối tiếp ở đầu ra của bộ điều khiển PD mờ một khâu tích phân (*hình 3.32*). Các giá trị mờ cho biến vào/ra ET, DET và P được định nghĩa như ở *hình 3.30*.



Hình 3.32: Hệ thống điều khiển nhiệt độ bằng bộ điều khiển mờ PI. Đối tượng là một khâu quán tính bậc nhất.



a) b)

Hình 3.33: Kết quả mô phỏng việc sử dụng bộ điều khiển mờ PI để điều khiển nhiệt độ.

a) Đáp ứng của hệ thống có thiết bị hợp thành MAX-MIN đối với tín hiệu vào hình sin.

b) Đáp ứng của hệ thống có thiết bị hợp thành MAX-PROD đối với tín hiệu vào hình sin.

Hình 3.33 là đáp ứng của hệ thống đối với các thiết bị hợp thành khác nhau. Tín hiệu chủ đạo đều là tín hiệu dạng hình sin. Đặc tính động học và chất lượng điều khiển của hệ thống khi sử dụng bộ điều khiển PI mờ hiển nhiên tốt hơn khi sử dụng các bộ điều khiển mờ khác. Ngoài ra các kết quả còn chỉ ra rằng sử dụng luật hợp thành max–PROD cho chất lượng điều khiển giống như là sử dụng luật hợp thành max–MIN, ở chỗ độ quá điều chỉnh cực đại và thời gian quá độ của hệ thống.

Một số kết luận

Với những nghiên cứu trên có thể rút ra các kết luận sau:

- Bộ điều khiển PI mờ cho đặc tính động học lý tưởng. Ở chế độ tĩnh, bộ điều khiển PI mờ có khả năng triệt tiêu sai lệch tĩnh.
- Bộ điều khiển P mờ cho đặc tính động học tương đối tốt, nhưng ở chế độ xác lập hệ thống lại tồn tại sai lệch tĩnh, hay nói một cách khác là độ chính xác của hệ thống kém hơn so với việc sử dụng bộ điều khiển PI mờ.

3.5 Bộ điều khiển mờ trượt

3.5.1 Nguyên lý điều khiển trượt

Xét lớp các đối tượng kiểu SISO một đầu vào $u\in\mathbb{R}$, một đầu ra $y\in\mathbb{R}$ có n biến trạng thái $x_1, x_2, \cdots, x_n$ với mô hình

$$\begin{aligned}\dot{x}_1 &= x_2\\ &\vdots\\ \dot{x}_{n-1} &= x_n\\ \dot{x}_n &= f(x_1, x_2, \cdots, x_n) + u\ ,\end{aligned} \tag{3.20}$$

trong đó $y = x_1$.

Nếu dùng ký hiệu vector

$$\underline{y} = \left(y, \frac{dy}{dt}, \cdots, \frac{d^{n-1}y}{dt^{n-1}}\right)^T = (x_1, \cdots, x_n)^T$$

thì công thức (3.20) viết được thành

$$\frac{d^n y}{(dt)^n} = f(\underline{y}) + u \tag{3.21}$$

Khi $n = 2$ sẽ có

$$\frac{d^2 y}{(dt)^2} = f(y, \dot{y}) + u \tag{3.22}$$

Bài toán đặt ra cho trường hợp tổng quát (n>2) là tổng hợp một bộ điều khiển mờ phản hồi u=$u(\underline{y})$ thỏa mãn $|u| \le u_{\max}$ để trạng thái $\underline{y}$ của đối tượng được đưa về một điểm trạng thái $\underline{y}_0 = \begin{pmatrix} x_{10}\\ \vdots \\ x_{n0}\end{pmatrix}$ mong muốn cho trước, tức là để sai lệch

$$\underline{e}(t) = \begin{pmatrix} e_1\\ \vdots \\ e_n\end{pmatrix} = \underline{y}_0 - \underline{y} \to \underline{0}\ .$$

với $\underline{0}$ là vector 0 trong $\mathbb{R}^n$.

Xét hàm chuyển đổi, ký hiệu bằng $s(e)$, có dạng như sau:

$$s(e) = \left(1 + \lambda \frac{d}{dt}\right)^{n-1} e(t) = \sum_{k=0}^{n-1} C_{n-1}^{k} \lambda^{k} \frac{d^k e(t)}{dt^k} \tag{3.23}$$

với $\lambda > 0$, $e(t) = e_1 = x_{10} - y(t)$ và $C_{n-1}^{k} = \frac{(n-1)!}{k!(n-1-k)!}$. Chú ý rằng hàm $s(e)$ là một hàm thực.

Khi $n = 2$ thì (3.23) trở thành

$$s(e) = e + \lambda \dot{e} . \tag{3.24}$$

Công thức (3.23) chỉ rằng khi có được $\underline{y} = \underline{y}_0$ thì $s(e)=0$, hơn nữa phương trình vi phân $s(e)=0$ với điều kiện ban đầu $\underline{e}(0)=\underline{0}$ cũng chỉ có một nghiệm

$$\underline{e}(t)=\underline{0} \quad \Leftrightarrow \quad \underline{y} = \underline{y}_0$$

duy nhất. Bởi vậy bài toán điều khiển sai lệch trạng thái trên cho không gian $\mathbb{R}^n$ trở thành bài toán tìm hàm $u=u(\underline{y})$ với $|u| \le u_{\max}$ sao cho $s(e)$ tiến đến 0, tức là một bài toán sai lệch trong $\mathbb{R}$. Để có được điều này thì hàm $u=u(\underline{y})$ phải được chọn sao cho khi $s(e) > 0$ thì $\dot{s}(e)<0$ và ngược lại khi $s(e) < 0$ thì $\dot{s}(e)>0$, tức là phải thỏa mãn

$$\dot{s}(e) \cdot \mathrm{sgn}(s) < 0 . \tag{3.25}$$

Điều kiện (3.25) được gọi là điều kiện trượt (*sliding condition*), nó đảm bảo rằng sai lệch $\underline{e}$ sẽ giảm nếu như trạng thái $\underline{y}$ chưa nằm trên mặt cong $s(e)=0$. Mặt cong $s(e) = 0$ vì thế còn có tên gọi là mặt cong trượt (*sliding surface*).

Xét riêng cho đối tượng hai trạng thái (n=2) thì với (3.24) và (3.22) có được điều kiện cho bài toán tổng hợp điều khiển để hệ trượt trên mặt cong $s(e)=0$ về điểm trạng thái $\underline{y}_0$ là

$$(\dot{e} + \lambda \ddot{e})\mathrm{sgn}(e + \lambda \dot{e}) < 0 \quad \Leftrightarrow \quad u = \begin{cases} < -\left(\dfrac{\dot{e}}{\lambda} + f(y,\dot{y})\right) & \text{khi } e + \lambda \dot{e} < 0 \\ > -\left(\dfrac{\dot{e}}{\lambda} + f(y,\dot{y})\right) & \text{khi } e + \lambda \dot{e} > 0 \end{cases}$$

Từ đây suy ra có thể chọn u như sau:

$$u = \begin{cases} u_{max} & \text{khi } e+\lambda\,\dot{e}<0 \\ -u_{max} & \text{khi } e+\lambda\,\dot{e}>0 \end{cases} \qquad (3.26a)$$

Công thức (3.26) được xây dựng với giả thiết là điểm trạng thái mong muốn $\underline{y}_0$ không phụ thuộc thời gian. Trong trường hợp $y_0 = y_0(t)$ là một hàm phụ thuộc thời gian mà có thể xem như là tín hiệu chủ đạo thì từ điều kiện

$$(\dot{e}+\lambda\,\ddot{e})\,\mathrm{sgn}(e+\lambda\,\dot{e})<0$$

hàm $u(t)$ phải có dạng như sau:

$$\dot{e}+\lambda\left(\ddot{y}_0 - f(y,\dot{y})-u\right)=\begin{cases} <0 & \text{khi} \quad e+\lambda\,\dot{e}>0 \\ >0 & \text{khi} \quad e+\lambda\,\dot{e}<0 \end{cases}$$

$$\Leftrightarrow \qquad u=\begin{cases} >\dfrac{\dot{e}}{\lambda}+f(y,\dot{y})-\ddot{y}_0 & \text{khi} \quad e+\lambda\,\dot{e}>0 \\ <\dfrac{\dot{e}}{\lambda}+f(y,\dot{y})-\ddot{y}_0 & \text{khi} \quad e+\lambda\,\dot{e}<0 \end{cases}$$

$$\Leftrightarrow \qquad u=\frac{\dot{e}}{\lambda}+f(y,\dot{y})-\ddot{y}_0+K\,\mathrm{sgn}(e+\lambda\,\dot{e}) \quad \text{với } K>0. \qquad (3.26b)$$

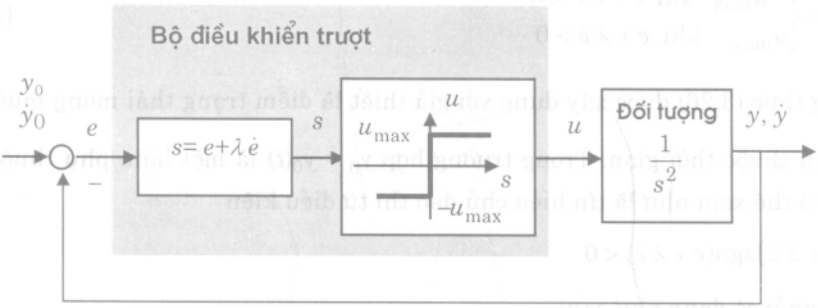
3.5.2 Hiện tượng Bang–Bang

Bộ điều khiển hai vị trí kết hợp với khâu hiệu chỉnh $s=e+\lambda\,\dot{e}$ tạo nên bộ điều khiển trượt theo công thức (3.26a) được xem như là bộ điều khiển có đặc tính tổng thể tốt nhất, nếu như xét đến cả sự tồn tại sai lệch tĩnh của hệ thống. Cộng thêm sự đơn giản trong thiết kế bộ điều khiển này thì kết quả ứng dụng nhận được khi sử dụng nó để điều khiển hệ thống thật là điều đáng kinh ngạc. Bộ điều khiển này phản ứng rất mạnh đối với sai lệch lớn còn đối với sai lệch nhỏ phản ứng nhanh và mềm dẻo. Bộ điều khiển hai vị trí có thể sử dụng tốt cho các mục đích điều khiển khác nhau, mặc dù ở chế độ xác lập vẫn tồn tại sai lệch tĩnh.

Bộ điều khiển trượt kinh điển được biết đến nhiều với những ứng dụng trong điều khiển tác động nhanh. Chúng bao gồm hai khâu PD và relay 2 vị trí mắc nối tiếp (*hình 3.34*). Các tham số λ và u_{max} được xác định từ bài toán tổng hợp cụ thể, trong đó λ quy định độ dốc đường thẳng

$$s(e)=e+\lambda\,\dot{e}=0 \qquad (3.27)$$

được gọi là *đường chuyển đổi* của tín hiệu điều khiển $u(t)$.

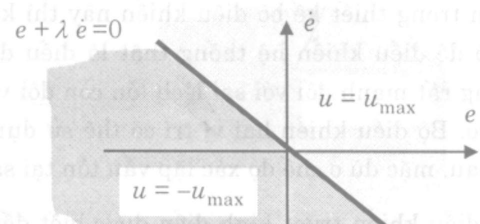


Hình 3.34: Sử dụng bộ điều khiển trượt để điều khiển tác động nhanh đối tượng tích phân kép.

Qua khâu relay hai vị trí, tín hiệu điều khiển tối ưu u sẽ nhận giá trị u_{max} nếu s nằm trên đường chuyển đổi và giá trị $-u_{max}$ khi s nằm dưới đường chuyển đổi trong mặt phẳng $(e,\ \dot{e})$ – xem *hình 3.35*.

Hãy xét bài toán điều khiển có $\underline{y}_0 = (y_0,\ \dot{y}_0)^T \equiv (0,\ 0)^T$ và đối tượng là khâu tích phân bậc hai

$$G(s) = \frac{1}{s^2} \quad \text{hay} \quad \begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u \end{cases} \qquad \text{với} \qquad y = x_1. \tag{3.28}$$



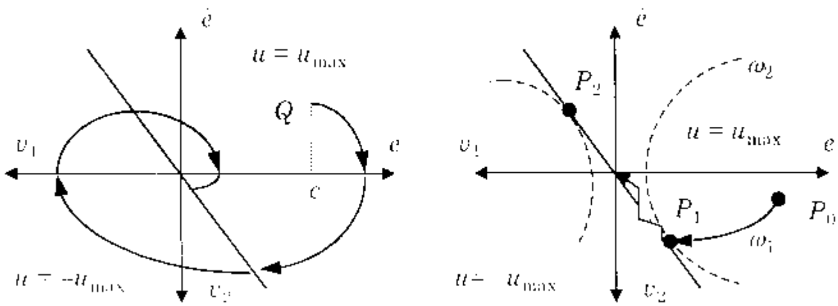
Hình 3.35: Biểu diễn đường chuyển đổi trong mặt phẳng $(e,\ \dot{e})$.

Như vậy thì với một giá trị u cố định (không phụ thuộc t), x_1 có quan hệ với x_2 như sau:

$$x_1 = \frac{x_2^2}{2u} + c\ , \tag{3.29}$$

trong đó c là hằng số phụ thuộc vào giá trị ban đầu của x_1 và x_2. Từ (3.29) có được đường quỹ đạo pha của đối tượng như trong *hình 3.36*.

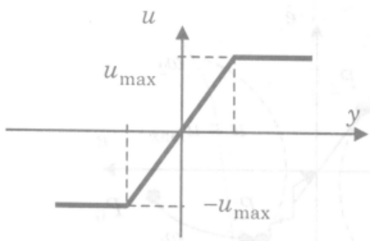
Do có giả thiết $y_0=0$, $\dot{y}_0=0$ nên $x_1=-e$ và $x_2=-\dot{e}$. Xuất phát từ một điểm trạng thái Q ban đầu, giả sử thuộc miền nằm trong nửa mặt phẳng phía trên đường chuyển đổi (3.27), quỹ đạo pha sẽ đi dọc theo đường parabol (3.29) ứng với trường hợp $u=u_{max}>0$ cho tới khi gặp đường chuyển đổi thì đổi hướng theo parabol (3.29) khác có $u=-u_{max}<0$.... Cứ như vậy quỹ đạo pha sẽ có xu hướng ngày càng tiến dần về điểm gốc tọa độ (xem *hình 3.35* bên trái), cho tới khi xảy ra trường hợp là đường parabol (3.29) tiếp theo lại nằm hoàn toàn về một phía đường chuyển đổi thì xuất hiện chế độ zick-zack về gốc tọa độ (còn gọi là *hiện tượng Bang Bang*) như *hình 3.36* bên phải mô tả.



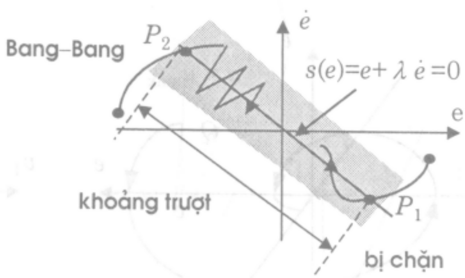
Hình 3.36: Quỹ đạo pha của hệ tác động nhanh.

Hiện tượng Bang-Bang được giải thích như sau: Giả sử từ điểm P_0 quỹ đạo pha đi theo Parabol ω_1 có $u=-u_{max}$ gặp đường chuyển đổi tại điểm P_1 là điểm tiếp xúc giữa đường chuyển đổi với đường Parabol ω_2 có $u=u_{max}$ và sẽ đi dọc theo đường Parabol ω_2 này. Nhưng vì ω_2 nằm hoàn toàn trong nửa mặt phẳng ứng với $u=u_{max}$ nên quỹ đạo chuyển ngay sang đường Parabol tiếp theo có $u=-u_{max}$ để quay trở về đường chuyển đổi $s(e)$, tại đó nó lại đổi hướng đi theo đường Parabol có $u=-u_{max}$ và quay về nửa mặt phẳng ứng với trường hợp $u=u_{max}$ ....

Về thực chất hiện tượng trượt trơn dọc theo đường chuyển đổi về gốc tọa độ chỉ xảy ra nếu như khâu relay hai vị trí có tần số chuyển đổi vô cùng lớn (∞). Trong trường hợp tần số chuyển đổi của khâu relay bị giới hạn, đường quỹ đạo pha sẽ không trượt dọc theo đường chuyển đổi mà dao động (zick–zack) quanh nó để về gốc tọa độ. Độ dốc λ của đường chuyển đổi (3.27) xác định độ dài $\overline{P_1P_2}$ đoạn xảy ra hiện tượng Bang–Bang. Độ dốc λ càng lớn, độ dài $\overline{P_1P_2}$ sẽ càng nhỏ và do đó hiện tượng Bang–Bang xảy ra càng nhanh. Để nâng cao được độ bền vững của hệ thống và tránh cho khâu relay phải làm việc với tần số quá lớn trong chế độ Bang–Bang, thông thường khâu relay hai vị trí được thay bằng một khâu khuếch đại bão hòa nhằm tạo ra sự chuyển đổi $u_{\max} \leftrightarrow -u_{\max}$ một cách liên tục (*hình 3.37*). Khi đó đường quỹ đạo pha trong chế độ Bang–Bang sẽ là một đường trơn như *hình 3.38* mô tả vì theo công thức (3.28) thì $\ddot{e} = -\dot{x}_2 = -u$ là một hàm liên tục theo t (và do đó $\dot{e}$ cũng liên tục theo t).



Hình 3.37: Thay khâu relay hai vị trí bằng khâu khuếch đại bão hòa.



Hình 3.38: Điều khiển trượt có dải băng η.

Với khâu khuếch đại bão hòa như *hình 3.37* thay cho khâu relay hai vị trí thì sai lệch $e(t)$ sẽ là $|e(t)| \le \varepsilon$, trong đó ε là sai số do khoảng chuyển đổi liên tục $u_{\max} \leftrightarrow -u_{\max}$ sinh ra.

Tương ứng với điều đó đường chuyển đổi $s(e)=0$ cũng được thay bằng miền chuyển đổi $|s(e)| \le \eta$ với η là một số thực dương thỏa mãn

$$\eta = \lambda\varepsilon. \tag{3.30}$$

Hình 3.39a trình bày kết quả mô phỏng đường quỹ đạo pha của đối tượng là khâu tích phân kép trong hệ thống điều khiển cho trong *hình 3.34* với khâu relay hai vị trí mờ thu được nhờ chương trình Winfact 3.0. *Hình 3.39b* là đường quỹ đạo pha khi khâu relay hai vị trí mờ được thay bằng khâu chuyển đổi bão hòa mờ.

Theo công thức (3.26b) thì tín hiệu điều khiển u với điều kiện trượt có dải băng η sẽ được chọn như sau:

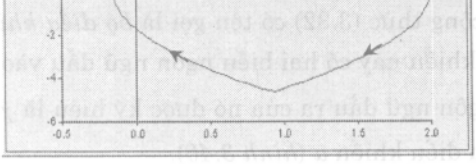
$$u=\frac{\dot{e}}{\lambda}+f(y,\dot{y})-\ddot{y}_0+K\cdot h\left(\frac{s(e)}{\eta}\right). \tag{3.31}$$

trong đó K là một hằng số dương và

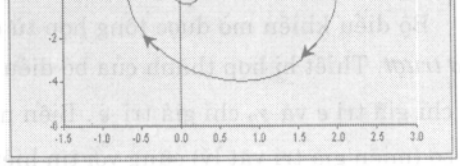
$$h\left(\frac{s(e)}{\eta}\right)=\begin{cases}\operatorname{sgn}\left(\frac{s(e)}{\eta}\right) & \text{khi } \left|\frac{s(e)}{\eta}\right|\geq 1\\ \frac{s(e)}{\eta} & \text{khi } \left|\frac{s(e)}{\eta}\right|\leq 1\end{cases}.$$

Rõ ràng là khi $\left|\frac{s(e)}{\eta}\right|\geq 1$, tức là khi đường quỹ đạo pha còn nằm ngoài dải băng η, thì $\operatorname{sgn}\left(\frac{s(e)}{\eta}\right)=\operatorname{sgn}(s)$ và do đó hai công thức (3.26b) và (3.31) là một. Riêng cho trường hợp đối tượng là khâu tích phân kép có mô hình cho trong (3.28) và tín hiệu chủ đạo $y_0\equiv 0$ thì

$$u=\frac{\dot{e}}{\lambda}+K\cdot h\left(\frac{s(e)}{\eta}\right). \tag{3.32}$$



a)



b)

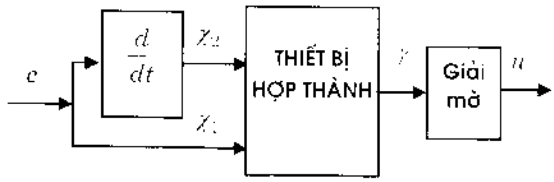
Hình 3.39: Quỹ đạo pha của đối tượng là khâu tích phân kép khi điều khiển với những khâu chuyển đổi khác nhau.

a) Quỹ đạo pha với khâu relay mở hai vị trí và hệ số $\lambda=0{,}2$.

b) Quỹ đạo pha với khâu chuyển đổi bão hòa với $\varepsilon=2$ và hệ số $\lambda=0{,}2$.

3.5.3 Tổng hợp bộ điều khiển mờ trượt

Từ bộ điều khiển mờ hai vị trí, *D. Driankov, H. Hellendorn* và *M. Reinfrank* đã phát triển thành bộ điều khiển mờ với tên *gọi Fuzzy-Sliding-Mode-Controller* (bộ điều khiển mờ hai vị trí ở chế độ trượt, hay ngắn gọn là bộ điều khiển mờ trượt). Thuật toán tổng hợp bộ điều khiển mờ trượt này sử dụng đạo hàm của sai lệch. Bộ điều khiển mờ kiểu này cho đặc tính động học rất tốt và đặc biệt không quá nhạy đối với các biến đổi của đối tượng và đối với các thiết kế với một mô hình đối tượng không chính xác.



Hình 3.40 Bộ điều khiển mờ trượt có hai đầu vào.

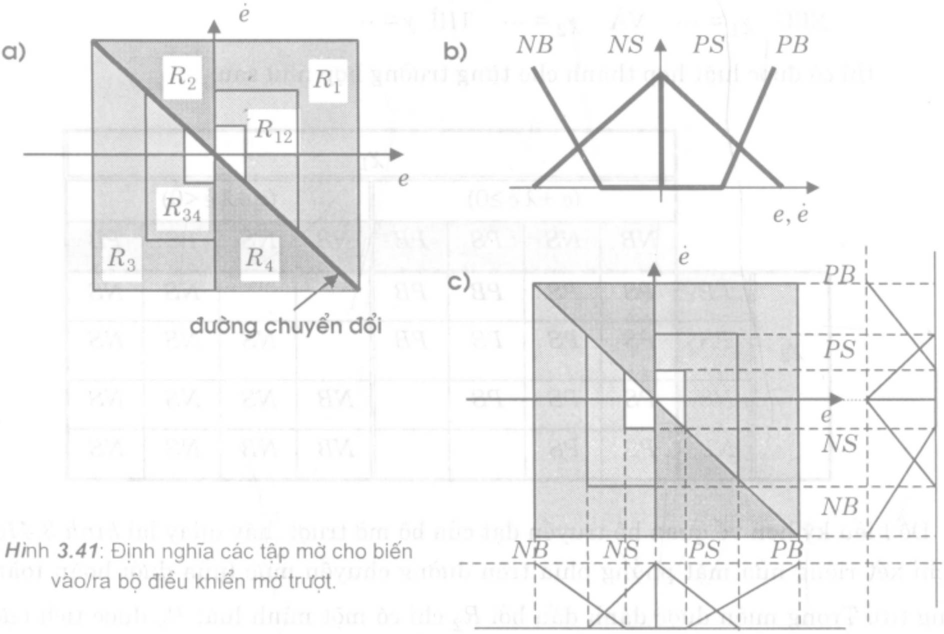
Với khoảng chuyển đổi liên tục, tín hiệu điều khiển u được xác định theo (3.31) hoặc nếu đối tượng có $f(y, \dot{y}) = 0$ như khâu tích phân kép và tín hiệu chủ đạo $\underline{y}_0$ là hằng số thì theo (3.32). Trong khoảng chuyển đổi liên tục không nhất thiết đường đặc tính $u(y)$ phải là một đường thẳng như *hình 3.37* mô tả mới có được đường quỹ đạo pha trơn trong chế độ trượt mà có thể thay bằng một đường liên tục bất kỳ. Điều chủ yếu là toàn bộ đường đặc tính $u(y)$ phải là một đường liên tục. Điều kiện để một bộ điều khiển mờ có được đường đặc tính liên tục là các giá trị mờ đầu vào phải có những miền xác định chồng nhau. Phần chồng nhau đó xác định sai lệch ε và do đó là dải băng η xung quanh đường chuyển đổi mà tại đó đường quỹ đạo pha sẽ trơn.

Bộ điều khiển mờ được tổng hợp từ công thức (3.32) có tên gọi là *bộ điều khiển mờ trượt*. Thiết bị hợp thành của bộ điều khiển này có hai biến ngôn ngữ đầu vào là χ_1 chỉ giá trị e và χ_2 chỉ giá trị $\dot{e}$. Biến ngôn ngữ đầu ra của nó được ký hiệu là γ có cơ sở (miền giá trị vật lý) cùng với tín hiệu điều khiển u (*hình 3.40*).

Thuật toán tổng hợp bộ điều khiển mờ trượt từ (3.32) bao gồm các bước:

1) Chia hai nửa mặt phẳng trên và dưới đường chuyển đổi thành các miền liên thông (có thể chồng nhau) và định nghĩa các giá trị mờ cho trên các miền đó (*hình 3.41a*). Ví dụ cả ba biến χ_1, χ_2 và γ đều có ba giá trị là *NB* (rất âm), *NS*

(âm ít), PS (dương ít) và PB (rất dương) như mô tả trong *hình 3.41b*. Ứng với việc chia miền trong *hình 3.40a*, các tập mờ NB, NS, PS, PB của từng biến có hàm thuộc được mô tả trong *hình 3.41c*.



Hình 3.41: Định nghĩa các tập mờ cho biến vào/ra bộ điều khiển mờ trượt.

2) Xây dựng luật hợp thành bao gồm các luật điều khiển R_k. Những luật điều khiển này được chia thành hai nhóm

   a) Nhóm 1 gồm các luật ứng với nửa mặt phẳng phía trên đường chuyển mức ($e + \lambda\dot{e} \geq 0$)

   R_1: NẾU ($\chi_1 = PS$ HOẶC $\chi_1 = PB$) VÀ $\chi_2 = PB$ THÌ $\gamma = PB$.

   R_2: NẾU ($\chi_1 = NS$ HOẶC $\chi_1 = NB$)

   HOẶC ($\chi_1 = PS$ VÀ ($\chi_2 = PS$ HOẶC $\chi_2 = NS$))

   THÌ $\gamma = PS$.

   b) Nhóm 2 gồm các luật ứng với nửa mặt phẳng phía dưới đường chuyển mức ($e + \lambda\dot{e} < 0$)

   R_3: NẾU ($\chi_1 = NS$ HOẶC $\chi_1 = NB$) VÀ $\chi_2 = NB$ THÌ $\gamma = NB$.

   R_4: NẾU ($\chi_1 = PS$ HOẶC $\chi_1 = PB$)

HOẶC ($\chi_1 = NS$ VÀ ($\chi_2 = NS$ HOẶC $\chi_1 = PS$))

THÌ $\gamma = NS$.

Viết lại các luật trên thành dạng chính tắc với

NẾU $\chi_1 = \cdots$ VÀ $\chi_2 = \cdots$ THÌ $\gamma = \cdots$

thì có được luật hợp thành cho từng trường hợp như sau:

		χ_1							
		$(e + \lambda \dot{e} \geq 0)$				$(e + \lambda \dot{e} < 0)$			
		NB	*NS*	*PS*	*PB*	*NB*	*NS*	*PS*	*PB*
χ_2	*PB*	*PS*	*PS*	*PB*	*PB*			*NS*	*NS*
	PS	*PS*	*PS*	*PS*	*PB*		*NS*	*NS*	*NS*
	NS	*PS*	*PS*	*PS*		*NB*	*NS*	*NS*	*NS*
	NB	*PS*	*PS*			*NB*	*NB*	*NS*	*NS*

Để hiểu kỹ hơn về quan hệ truyền đạt của bộ mờ trượt, hãy quay lại *hình 3.41a* và chỉ xét riêng nửa mặt phẳng phía trên đường chuyển mức (nửa dưới hoàn toàn tương tự). Trong miền được đánh dấu bởi R_2 chỉ có một mình luật R_2 được tích cực nên đầu ra γ sẽ luôn là PS. Cũng như vậy trong miền R_1 do chỉ có luật R_1 được tích cực nên $\gamma = PB$. Riêng trong miền R_{12} thì cả hai luật R_1 và R_2 cùng được tích cực và do đó R_{12} được xem như miền quá độ để biến γ chuyển giá trị một cách liên tục. Giá trị "mờ" cụ thể của γ trong miền này phụ thuộc vào độ thỏa mãn của R_1 và R_2.

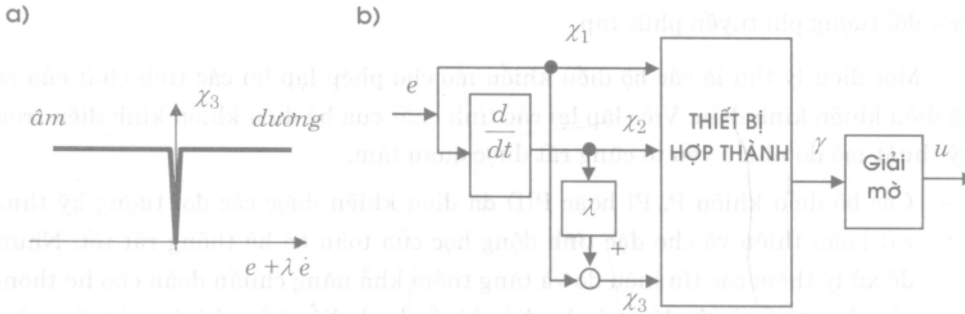
Có thể cài đặt thiết bị hợp thành theo luật trên với 3 biến vào. Ngoài hai biến cũ là χ_1, χ_2 như đã định nghĩa còn có thêm χ_3 để xác định điều kiện cho tổng $e + \lambda \dot{e}$ âm hay dương. Luật điều khiển khi đó sẽ có dạng

NẾU $\chi_1 = \cdots$ VÀ $\chi_2 = \cdots$ VÀ $\chi_3 = \cdots$ THÌ $\gamma = \cdots$,

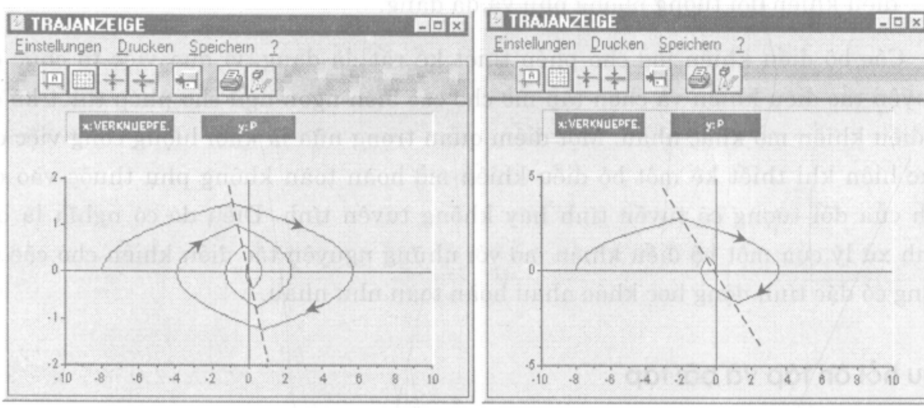
trong đó χ_3 có hai giá trị mờ cho trong *hình 3.38a)*. Ví dụ

NẾU $\chi_1 = NB$ VÀ $\chi_2 = PB$ VÀ $\chi_3 = dương$ THÌ $\gamma = PS$.

Hình 3.42b mô tả bộ mờ trượt có 3 đầu vào. *Hình 3.43* là kết quả mô phỏng bộ mờ trượt có 3 đầu vào được tổng hợp trên phần mềm Winfact 3.0.



Hình 3.42: a) Định nghĩa giá trị mờ cho biến ngôn ngữ χ_3 để xác định dấu của tổng $e+\lambda\dot{e}$.
b) Bộ điều khiển mờ trượt có ba đầu vào.



a) b)

Hình 3.43: Kết quả mô phỏng trên Winfact 3.0 việc ứng dụng bộ mờ trượt để điều khiển tác động nhanh đối tượng tích phân kép.

a) Quỹ đạo pha của đối tượng với $\lambda = 0{,}4$.

b) Quỹ đạo pha của đối tượng với $\lambda = 0{,}8$.

3.6 Kết luận

Sự ghép nối giữa các khâu tuyến tính với hệ mờ (khâu phi tuyến) đã cho ra đời các bộ điều khiển với những tính chất rất hoàn hảo và tạo ra một khả năng mới trong kỹ thuật điều khiển tự động, đó là điều khiển các đối tượng phức tạp, các đối tượng mà cho đến nay việc khống chế nó hoàn toàn khó khăn và hầu như không điều khiển được theo phương pháp kinh điển. Ở đây cũng khẳng định được một điều là trong nhiều trường hợp một bộ điều khiển mờ đơn giản cũng có thể điều khiển tốt một đối tượng phi tuyến phức tạp.

Một điều lý thú là các bộ điều khiển mờ cho phép lặp lại các tính chất của các bộ điều khiển kinh điển. Việc lặp lại các tính chất của bộ điều khiển kinh điển trong kỹ thuật mờ do nhiều yếu tố cũng rất được quan tâm:

Các bộ điều khiển P, PI hoặc PID đã điều khiển được các đối tượng kỹ thuật rất hoàn thiện và cho đặc tính động học của toàn bộ hệ thống rất tốt. Nhưng để xử lý thêm các tín hiệu đo và tăng thêm khả năng chuẩn đoán cho hệ thống, cần thay thế ở bước đầu tiên bộ điều khiển kinh điển bằng bộ điều khiển mờ và phát triển thêm hệ điều khiển dựa trên cơ sở của bộ điều khiển mờ này để có được các tính chất điều khiển mong muốn.

Cùng với kỹ thuật mờ, các bộ điều khiển chung cho phép tạo ra một khả năng điều khiển đối tượng phong phú và đa dạng.

Các bộ điều khiển mờ cho phép thiết kế rất đa dạng, vì qua việc tổ chức các nguyên tắc điều khiển và chọn tập mờ cho các biến ngôn ngữ cho phép thiết kế các bộ điều khiển mờ khác nhau. Một điểm quan trọng nữa là khối lượng công việc cần thực hiện khi thiết kế một bộ điều khiển mờ hoàn toàn không phụ thuộc vào đặc tính của đối tượng có tuyến tính hay không tuyến tính. Điều đó có nghĩa là quá trình xử lý của một bộ điều khiển mờ với những nguyên tắc điều khiển cho các đối tượng có đặc tính động học khác nhau hoàn toàn như nhau.

Câu hỏi ôn tập và bài tập

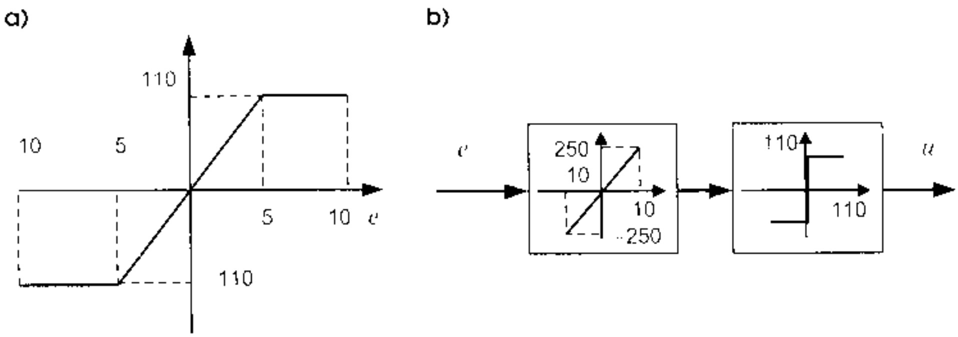
1) Cho một đối tượng nhiệt có mô hình toán học được biểu diễn dưới dạng hàm truyền đạt:

$$G(s) = \frac{250}{1 + 1200s}$$

a) Hãy thiết kế một bộ điều khiển mờ có quan hệ vào/ra tuyến tính $u(e)=50e$, trong đó e là tín hiệu sai lệch giữa giá trị đặt trước và đầu ra y của đối tượng. Vùng cần điều khiển trong khoảng giới hạn của $e \in [-200, 200]$, u là tín hiệu ra của bộ điều khiển mờ có vùng giá trị giới hạn trong khoảng $u \in [-1000, 1000]$. Kiểm tra chất lượng của hệ bằng phần mềm WinFACT hoặc phần mềm MATLAB.

b) Để khắc phục sai lệch tĩnh của hệ thống, người ta đã thiết kế thêm sau bộ điều khiển mờ một khâu điều khiển tích phân. Hãy xác định vùng làm việc cho phép của hằng số thời gian tích phân T_I để hệ đảm bảo tính ổn định. Kiểm tra kết quả tính toán.

2) Cho một động cơ điện một chiều được mô tả bằng phương trình vi phân bậc hai $\frac{d^2y}{dt^2} = 5u(t)$ được điều khiển bằng một bộ điều khiển mờ có đặc tính vào ra tĩnh như hình 3.44a)



Hình 3.44: Minh họa bài tập 2.

a) Hãy phân tích chất lượng của hệ thống được thiết kế.

b) Chất lượng của hệ thống sẽ thay đổi như thế nào nếu thay thế bộ điều khiển trên trên bằng hai bộ điều khiển mờ mắc nối tiếp nhau và có đặc tính vào ra tĩnh như hình trên bên phải:

c) Chọn các tập mờ cho biến vào, ra và định nghĩa luật hợp thành để có được những bộ điều khiển mờ có quan hệ vào/ra như trong hình đã cho ở hai câu trên.

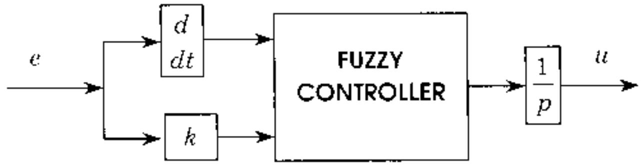
3) Cho một đối tượng điều khiển có mô hình toán học biểu diễn dưới dạng hàm truyền đạt như sau

$$G(s) = \frac{25}{(3s^2+15s+1)}$$

được điều khiển bằng một bộ mờ lai có cấu trúc như hình 3.45:

a) Hãy thiết kế bộ điều khiển mờ sao cho mạch vòng ổn định tốc độ động cơ làm việc ổn định và có sai lệch tĩnh bằng 0.

b) Kiểm tra kết quả tính toán bằng phần mềm WinFACT hoặc phần mềm MATLAB.



Hình 3.45: Cho bài tập 3.

4 HỆ MỜ LAI VÀ HỆ MỜ THÍCH NGHI

4.1 Khái niệm chung

Hệ mờ lai (Fuzzy–hybrid) là một hệ thống diều khiển tự dộng trong đó thiết bị điều khiển bao gồm hai thành phần:

- phần thiết bị điều khiển kinh điển,
- phần hệ mờ.

Bộ điều khiển mà trong quá trình làm việc có khả năng tự chỉnh định thông số của nó cho phù hợp với sự thay đổi của đối tượng được gọi là *bộ điều khiển thích nghi*. Một hệ thống điều khiển thích nghi, cho dù có hay không sự tham gia của các hệ mờ, là hệ thống điều khiển phát triển cao và có tiềm năng đặt biệt, song gắn liền với những ưu điểm dó là khối lượng tính toán thiết kế rất lớn.

Phần lớn các hệ thống điều khiển mờ lai là hệ thích nghi, nhưng không phải mọi hệ lai là hệ thích nghi. Khái niệm "thích nghi" định nghĩa ở đây không bao gồm các giải pháp thay đổi cấu trúc hệ thống cho dù sự thay đổi đó có thể phần nào phục vụ mục đích thích nghi. Ví dụ một hệ thống điều khiển có khâu tiền xử lý để tự chỉnh định tham số bộ điều khiển *một lần* khi bắt đầu khởi tạo hệ thống, sau đó trong suốt quá trình làm việc các thông số đó không được thay đổi nữa, thì không thuộc nhóm các hệ thích nghi theo nghĩa trên. Hoặc một trường hợp khác, hệ thống mà tính "tự thích nghi" của thiết bị điều khiển được thực hiện bằng cách dựa vào sự thay đổi của đối tượng mà chọn khâu điều khiển có tham số thích hợp trong số các khâu cùng cấu trúc nhưng với những tham số khác nhau đã được cài đặt từ trước, cũng không được gọi là hệ điều khiển thích nghi. Tính "thích nghi" của các loại hệ thống này được thực hiện bằng cách chuyển công tắc đến bộ điều khiển có tham số phù hợp chứ không phải tự chỉnh định lại tham số của bộ điều khiển đó theo đúng nghĩa của một bộ điều khiển thích nghi đã dịnh nghĩa.

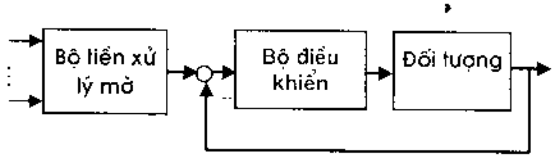
Thực tế ứng dụng kỹ thuật mờ cho thấy rằng không phải là cứ thay một bộ điều khiển mờ vào chỗ bộ điều khiển kinh điển thì sẽ có một hệ thống tốt hơn. Trong nhiều trường hợp, để hệ thống có đặc tính động học tốt và bền vững (robust) cần phải thiết kế thiết bị điều khiển lai giữa bộ điều khiển mờ và bộ điều khiển kinh điển. Ngoài ra về mặt tâm lý, các nhà thiết kế hệ thống nhiều khi cũng cảm thấy yên tâm hơn khi chọn bộ điều khiển đã được quen biết và thông dụng từ lâu, ví dụ bộ điều khiển PID kinh điển, hơn là chọn bộ điều khiển mờ cho phương án thiết kế của mình.

Một số dạng cấu trúc cơ bản của hệ mờ lai và hệ mờ thích nghi cùng với khả năng thực hiện và một vài hướng dẫn thiết kế sẽ được giới thiệu trong chương này.

4.2 Hệ mờ lai

4.2.1 Hệ lai không thích nghi có bộ điều khiển kinh điển

Hãy quan sát cấu trúc của một hệ lai trong *hình 4.1* có bộ tiền xử lý mờ. Nhiệm vụ điều khiển được giải quyết bằng bộ điều khiển kinh điển (ví dụ như bộ điều khiển PID kinh điển) và các thông số của bộ điều khiển không được chỉnh định thích nghi. Hệ mờ được sử dụng để điều chế tín hiệu chủ đạo cho phù hợp với hệ thống điều khiển. Về nguyên tắc, tín hiệu chủ đạo là một hàm thời gian bất kỳ và chỉ phụ thuộc vào những ứng dụng cụ thể. Một cấu trúc cụ thể của hệ mờ lai có bộ tiền xử lý mờ như vậy được biểu diễn trong *hình 4.2*.

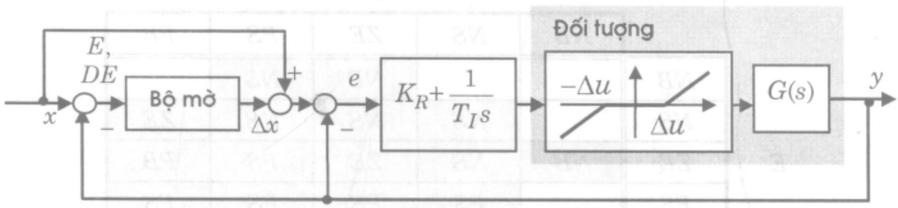


Hình 4.1: Bộ điều khiển mờ lai có khâu tiền xử lý mờ.

Tín hiệu chủ đạo x đưa vào hệ thống được điều chế qua bộ mờ. Tín hiệu vào x được so sánh với tín hiệu ra y của hệ thống và sai lệch E cùng đạo hàm DE của nó được đưa vào đầu vào của bộ lọc mờ tạo ra một lượng hiệu chỉnh Δx, tín hiệu chủ đạo đã được lọc có giá trị bằng $x+\Delta x$. Tác dụng của bộ lọc mờ trong toàn bộ hệ thống là làm cho hệ có đặc tính động tốt hơn và nâng cao khả năng bền vững của hệ khi các thông số trong hệ biến đổi. Nguyên tắc điều khiển này sẽ được minh họa bằng ví dụ dưới đây với đối tượng gồm khâu tuyến tính có mô hình toán học biểu diễn dưới dạng hàm truyền đạt.

$$G(s) = \frac{1}{s(1 + 0{,}2s)} \tag{4.1}$$

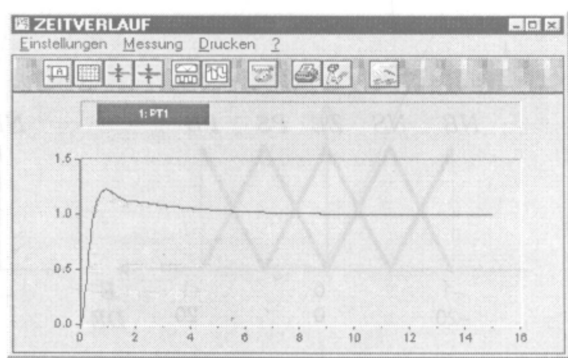
nối tiếp với một khâu khuếch đại có vùng kém nhạy có độ rộng bằng $2\Delta u$ cùng với hệ số khuếch đại K. Bộ điều khiển được chọn là bộ điều khiển PI với các thông số: $K_R = 10$ và $T_I = 0.3s$.



Hình 4.2: Hệ mờ lai với bộ lọc mờ cho tín hiệu chủ đạo x.

Trước tiên cho hệ thống làm việc không có bộ lọc mờ và các thông số của bộ khuếch đại có vùng kém nhạy như sau:

$\Delta u = 1 K = 0{,}5$.



Hình 4.3: Đáp ứng hệ thống không có bộ lọc mờ với tín hiệu $1(t)$.

Đáp ứng của hệ thống đối với tín hiệu vào là hàm bậc thang đơn vị được biểu diễn trong *hình 4.3*. Kết quả với những giá trị Δu, K khác nhau còn cho thấy rằng đặc tính động của hệ thống sẽ xấu đi khi vùng kém nhạy rộng và hệ số khuếch đại lớn. Để hiệu chỉnh đặc tính động và nâng cao độ bền vững của hệ thống, một bộ lọc mờ được đưa vào hệ thống như trong *hình 4.2*. Bộ lọc mờ điều chế tín hiệu hiệu chỉnh Δx dựa trên việc phân tích tín hiệu sai lệch E và đạo hàm DE của nó. Hàm thuộc của các đại lượng vào E và DE được chọn dưới dạng hình tam giác tiêu chuẩn và cho đại lượng ra Δx là dạng singleton (*hình 4.4*).

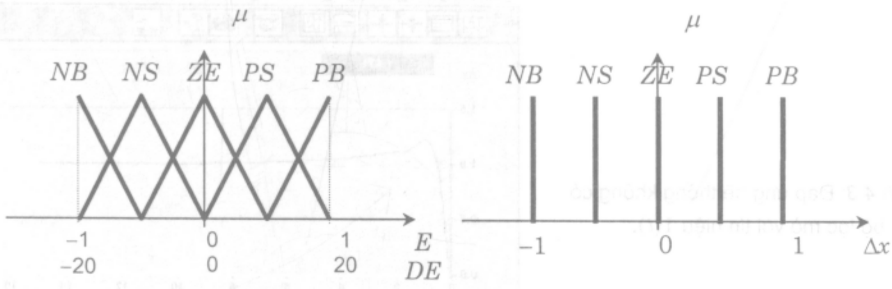
Luật điều khiển cơ sở không phải sử dụng tất cả các nguyên tắc điều khiển vì tín hiệu vào dạng hàm bậc thang đơn vị có độ biến đổi lớn, do vậy chỉ cần sử dụng một phần các nguyên tắc điều khiển. Luật điều khiển sẽ như sau:

	E \ DE	*NB*	*NS*	*ZE*	*PS*	*PB*
	NB			*NB*	*NS*	
	NS		*NS*	*NS*	*NS*	*ZE*
E	*ZE*	*NB*	*NS*	*ZE*	*PS*	*PB*
	PS		*PS*	*PS*	*PS*	*PS*
	PB		*PB*	*PS*	*PB*	

Tất cả 18 luật điều khiển từ bảng trên có cấu trúc chuẩn dưới dạng mệnh đề nguyên nhân và mệnh đề kết quả như:

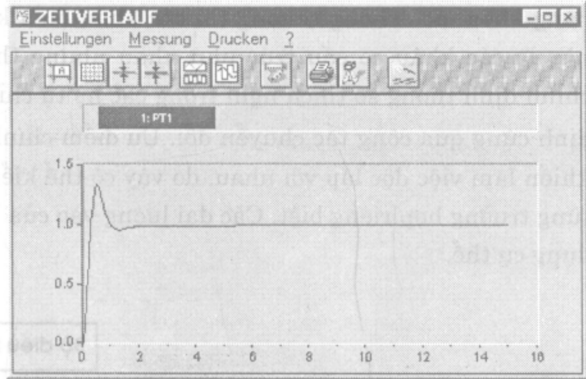
NẾU $E = ZE$ VÀ $DE = NS$ THÌ $\Delta x = NS$

⋮



Hình 4.4: Định nghĩa các giá trị mờ cho biến ngôn ngữ E, DE và Δx.

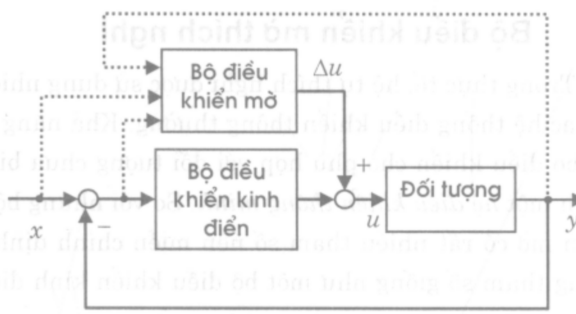
Tác dụng của bộ lọc mờ vào chất lượng động của hệ thống được chỉ ra ở *hình 4.5*. Kết quả khẳng định bộ lọc mờ cải thiện rõ rệt đặc tính động ở chỗ quá trình quá độ kết thúc rất nhanh và hầu như không phụ thuộc vào Δu .



Hình 4.5: Đáp ứng của hệ thống có bộ lọc mờ cho tín hiệu chủ đạo.

4.2.2 Hệ mờ lai cascade

Một cấu trúc mờ lai khác được biểu diễn trong *hình 4.6*, ở đó phần bù tín hiệu điều chỉnh Δu được lấy ra từ bộ điều khiển mờ.

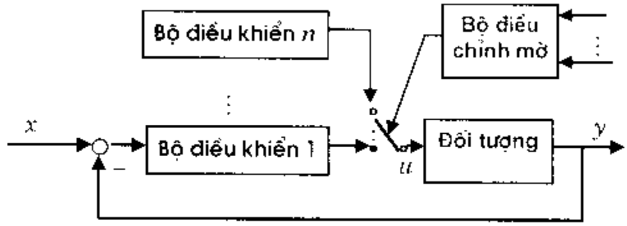


Hình 4.6: Cấu trúc hệ mờ lai Cascade.

Trong trường hợp hệ thống có cấu trúc như trên thì việc chọn các đại lượng đầu vào của hệ mờ phụ thuộc vào từng ứng dụng cụ thể. Tất nhiên các đại lượng thường được sử dụng làm tín hiệu vào của hệ mờ là tín hiệu chủ đạo x, sai lệch e, tín hiệu ra y cùng với đạo hàm hoặc tích phân của các đại lượng này. Về nguyên tắc có thể sử dụng các đại lượng khác của đối tượng cũng như sử dụng các nhiễu xác định được.

4.2.3 Điều khiển công tắc chuyển đổi "thích nghi" bằng khóa mờ

Điều khiển hệ thống theo kiểu chuyển đổi khâu điều khiển có tham số và cấu trúc phù hợp với điểm làm việc của đối tượng đòi hỏi thiết bị điều khiển phải chứa đựng tất cả các khâu có cấu trúc và tham số khác nhau cho từng trường hợp (*hình 4.7*). Hệ thống sẽ tự chọn khâu điều khiển có tham số phù hợp với đối tượng. Điều khiển công tắc chuyển đổi vị trí để chọn khâu điều khiển phù hợp được thực hiện bằng khóa mờ. Thông thường thì các khâu điều khiển được dùng trong trường hợp này là các khâu có cấu trúc như nhau nhưng tham số khác nhau. Khác với việc chỉnh định thông số thích nghi trong các hệ tự chỉnh, các thông số ở đây được chỉnh định cứng qua công tắc chuyển đổi. Ưu điểm chính của hệ thống này là các bộ điều khiển làm việc độc lập với nhau, do vậy có thể kiểm tra tính ổn định của hệ ứng với từng trường hợp riêng biệt. Các đại lượng vào của hệ mờ được xác định cho từng ứng dụng cụ thể.



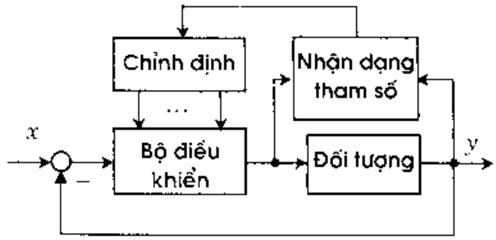
Hình 4.7: Chọn bộ điều khiển "thích nghi" bằng khóa mờ.

4.3 Bộ điều khiển mờ thích nghi

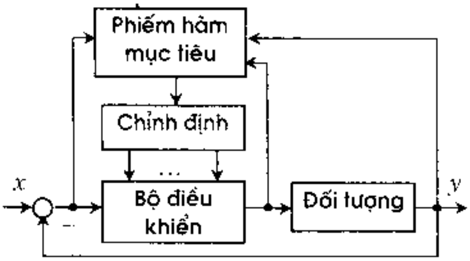
Trong thực tế, hệ tự thích nghi được sử dụng nhiều vì những ưu điểm của nó so với các hệ thống điều khiển thông thường. Khả năng tự chỉnh định lại các thông số của bộ điều khiển cho phù hợp với đối tượng chưa biết rõ đã đưa hệ thích nghi trở thành một *hệ điều khiển thông minh*. So với những bộ điều khiển kinh điển, bộ điều khiển mờ có rất nhiều tham số nên miền chỉnh định cho hệ mờ rất lớn. Bên cạnh những tham số giống như một bộ điều khiển kinh điển, ví dụ bộ PID mờ cũng có 3 tham số gồm độ khuếch đại K_R, hằng số tích phân T_I, hằng số vi phân T_D ..., một bộ điều khiển mờ còn có thêm những hàm thuộc cho các giá trị mờ, luật điều khiển, các phép toán HOẶC, VÀ, thiết bị hợp thành và nguyên lý giải mờ cũng là những tham số chỉnh định được.

4.3.1 Các phương pháp điều khiển mờ thích nghi

Các bộ điều khiển mờ thích nghi có khả năng chỉnh định các tham số của tập mờ (các hàm thuộc) gọi là *bộ điều khiển mờ tự chỉnh* (Self-Tuning-Controller). Bộ điều khiển mờ có khả năng chỉnh định lại các luật điều khiển, ví dụ chuyển từ ··· THÌ $y = NS$ ··· thành ··· THÌ $y = ZE$ ···, được gọi là *bộ điều khiển mờ tự thay đổi cấu trúc*. Trong trường hợp này, hệ thống có thể bắt đầu làm việc với các luật đã được chỉnh định hoặc với bộ điều khiển còn chưa đủ các luật điều khiển. Các luật điều khiển cần được bổ sung thêm sẽ được thiết lập trong quá trình "học".



Hình 4.8: Phương pháp điều khiển thích nghi trực tiếp



Hình 4.9: Phương pháp điều khiển thích nghi gián tiếp

Hệ thống điều khiển cơ bản của hệ thích nghi hoàn toàn giống như các hệ thống điều khiển một mạch vòng thông thường. Các tính chất của đối tượng dưới tác dụng của điều khiển, thường được tiến hành nhận dạng qua hệ kín hoặc thông qua các đại lượng đặc trưng của hệ như độ quá điều chỉnh cực đại, thời gian quá điều chỉnh cực đại, bình phương sai lệch, tích phân sai số tuyệt đối.... Mạch vòng thích nghi cho hệ điều khiển mờ hoặc không mờ đều được xây dựng dựa trên hai phương pháp:

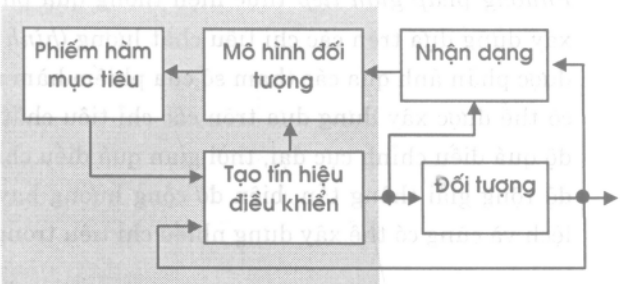
- *Phương pháp trực tiếp* thực hiện qua việc nhận dạng thường xuyên các tham số của đối tượng trong hệ kín (*hình 4.8*). Quá trình nhận dạng thông số của đối tượng có thể thực hiện bằng cách thường xuyên đo trạng thái của tín hiệu vào/ra của đối tượng và chọn một thuật toán nhận dạng hợp lý. Tất nhiên là phải đi kèm với giả thiết là mô hình của đối tượng đã biết trước (ví dụ như đối tượng có mô hình $\dfrac{K_p}{1+sT_p}$ của một khâu quán tính bậc một có trễ và các tham số K_p, T_p cần phải được nhận dạng). Mô hình của đối tượng cũng có thể là mô hình mờ. Mô hình mờ là mô hình biểu diễn dưới dạng câu điều kiện: NẾU ... THÌ ··· hoặc dưới dạng ma trận quan hệ R (ma trận biểu diễn luật hợp thành).
- *Phương pháp gián tiếp* thực hiện thông qua phiếm hàm mục tiêu của hệ kín xây dựng dựa trên các chỉ tiêu chất lượng (*hình 4.9*). Chất lượng của hệ thống được phản ánh qua các tham số của phiếm hàm mục tiêu. Phiếm hàm mục tiêu có thể được xây dựng dựa trên các chỉ tiêu chất lượng động của hệ thống như độ quá điều chỉnh cực đại, thời gian quá điều chỉnh, các chỉ tiêu ở miền tần số, độ rộng giải thông tần, biên độ cộng hưởng hay các tiêu chuẩn tích phân sai lệch và cũng có thể xây dựng nhiều chỉ tiêu trong cùng một phiếm hàm.

4.3.2 Bộ điều khiển mờ tự chỉnh cấu trúc

Bộ điều khiển mờ tự chỉnh định các luật điều khiển được gọi là bộ điều khiển mờ tự chỉnh cấu trúc. Bộ chỉnh định được thiết kế đảm bảo đầu ra là giá trị hiệu chỉnh của tín hiệu điều khiển $u(t)$ (tín hiệu ra của bộ điều khiển). Để thay đổi luật điều khiển trước tiên là phải xác định được quan hệ giữa giá trị được hiệu chỉnh ở đầu ra của bộ điều khiển với giá trị biến đổi ở đầu vào. Do vậy cần có mô hình thô của đối tượng, mô hình này dùng để tính toán giá trị đầu vào tương ứng với một giá trị đầu ra cần đạt được của bộ điều khiển. Dựa trên tín hiệu ra mong muốn và tín hiệu vào tương ứng của bộ điều khiển có thể xác định và hiệu chỉnh các nguyên tắc điều khiển, các nguyên tắc này đảm bảo chất lượng điều khiển của hệ thống. Một câu hỏi được đặt ra là những giá trị nào của tín hiệu điều khiển $u(t)$ (tín hiệu ra của bộ điều khiển) sẽ làm cho chất lượng của hệ thống xấu đi?. Để trả lời được câu hỏi này phải xác định được đặc tính động học của hệ thống. Đối với những đối tượng bậc cao có thời gian trễ lớn có thể có thời gian chỉnh định chậm, còn đối với các hệ thống bậc thấp có thời gian trễ nhỏ yêu cầu thời gian chỉnh định nhanh. Tóm lại, việc chỉnh định chỉ có ý nghĩa khi quá trình chỉnh định kết thúc trước khi hệ thống kết thúc quá trình quá độ.

4.3.3 Bộ điều khiển mờ tự chỉnh có mô hình theo dõi

Một hệ tự chỉnh không những chỉnh định trực tiếp tham số của bộ điều khiển mà còn chỉnh định cả tham số của mô hình đối tượng được gọi là bộ tự chỉnh có mô hình theo dõi (Model–Based Controller *MBC*). Với bộ điều khiển như vậy hệ mờ không chỉ sử dụng cho quá trình điều khiển đối tượng mà còn phục vụ cho quá trình nhận dạng đối tượng, được gọi là "mô hình đối tượng mờ". Hệ tự chỉnh mờ có cấu trúc như *hình 4.10* đã được áp dụng trong hệ thống điều khiển đường tầu điện ngầm ở Sendai/Nhật bản và trong các hệ thống điều khiển mức, các hệ thống mà mức độ khó thực hiện do hằng số thời gian trễ của đối tượng gây ra.



Hình 4.10: Điều khiển thích nghi có mô hình theo dõi.

Bộ điều khiển mờ có mô hình theo dõi *MBC* bao gồm ba thành phần chính:

1) Mô hình đối tượng mờ (thường có dạng ma trận quan hệ), được xác định trong khi hệ thống đang làm việc bằng cách đo và phân tích các tín hiệu đầu vào/ra của đối tượng. Vì mô hình của đối tượng gián tiếp xác định các luật hợp thành của bộ điều khiển do vậy bộ điều khiển *MBC* cũng chính là bộ điều khiển mờ tự chỉnh cấu trúc.

2) Các chỉ tiêu chất lượng được sử dụng trong phiếm hàm mục đích thường được đưa dưới dạng hàm thuộc. Thí dụ như trong hệ thống điều khiển mức, độ chênh so với mức mong muốn được biểu diễn bằng hàm thuộc dạng hình tam giác, trong đó đỉnh của tam giác chính là giá trị mức mong muốn. Nếu cần tối ưu đồng thời nhiều phiếm hàm mục đích, có thể tổ hợp các chỉ tiêu tương ứng theo toán tử liên kết min.

3) *Lựa chọn tín hiệu điều khiển u* từ tập hợp của các tín hiệu điều khiển xác định từ mô hình đối tượng và đảm bảo chỉ tiêu chất lượng nào đó của hệ thống tốt nhất.

Những bài toán thiết kế theo cấu trúc này thường có những giả thiết như sau:

1) Những thông tin về mô hình đối tượng còn rất ít khi bắt đầu quá trình điều khiển. Bởi vậy thông thường quá trình nhận dạng phải bắt đầu với ma trận quan hệ "rỗng". Theo kinh nghiệm của các phương pháp cũ thì nên bắt đầu với mô hình của đối tượng được nhận dạng ở hệ hở được gọi là *mô hình ban đầu*.

2) Trong những trường hợp đặc biệt, ở giai đoạn đầu do thiếu thông tin về đối tượng nên các quyết định điều khiển không thoả mãn được phiếm hàm mục tiêu, hay nói một cách khác là không thỏa mãn được các chỉ tiêu chất lượng đặt ra. Trong những trường hợp như vậy nên thiết kế thêm một bộ điều khiển phụ với chức năng ít nhất là giữ cho hệ thống làm việc ổn định cho đến khi mô hình đối tượng mờ được xác định hoàn toàn. Đơn giản nhất là nên giữ lại giá trị tín hiệu điều khiển *u(t)* của bước trước đó.

Thực hiện từng phần của bộ điều khiển mờ tự chỉnh có mô hình theo dõi phụ thuộc rất nhiều vào đối tượng điều khiển. Các phương pháp thiết kế và các cấu trúc khác nhau của hệ thống này có thể tìm thấy trong tài liệu tham khảo "*LIU. M.H.* Fuzzy–Modellbildung und ihre Anwendung, 1994".

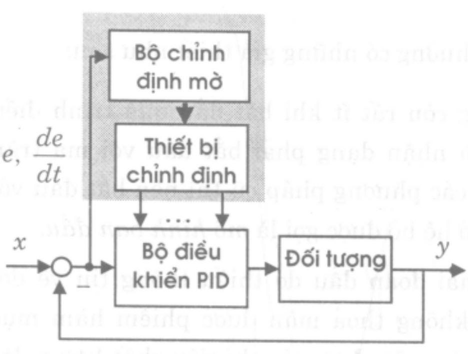
4.4 Chỉnh định mờ tham số bộ điều khiển PID

Bộ điều khiển PID kinh điển được thiết kế dựa trên các phương pháp đã biết như phương pháp tổng hợp hệ thống của *Ziegler* và *Nichols*, phương pháp của *Offerein*, phương pháp của *Reinisch* .... Bộ điều khiển này là cơ sở cho việc tổng hợp hệ thích nghi sau này. Khác với phương pháp dùng công tắc chọn bộ điều khiển phù hợp trong hệ lai, các thông số của bộ điều khiển thích nghi được hiệu chỉnh trơn. Một bộ điều khiển PID với đầu vào $e(t)$, đầu ra $u(t)$ có mô hình toán học như sau

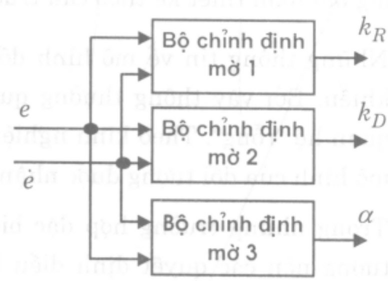
$$u(t) = K_R\left[e(t) + \frac{1}{T_I}\int_0^t e(\tau)d\tau + T_D\frac{de(t)}{dt}\right] \tag{4.2}$$

hoặc $$G_{\text{PID}}(s) = K_R + \frac{K_I}{s} + K_D s, \tag{4.3}$$

trong đó $T_I = \dfrac{K_R}{K_I}$ và $T_D = \dfrac{K_D}{K_R}$.



Hình 4.11: Phương pháp chỉnh định mờ tham số bộ điều khiển PID.



Hình 4.12: Bên trong bộ chỉnh định mờ.

Các tham số K_R, T_I, T_D hay K_R, K_I, K_D của bộ điều khiển PID được chỉnh định trên cơ sở phân tích tín hiệu chủ đạo và tín hiệu ra của hệ thống, chính xác hơn là sai lệch $e(t)$ và đạo hàm $\frac{de}{dt}$ của sai lệch. Có nhiều phương pháp chỉnh định các tham số cho bộ điều khiển PID như chỉnh định qua phiếm hàm mục tiêu, chỉnh định trực tiếp, song phương án đơn giản nhưng dễ áp dụng hơn cả là phương pháp chỉnh định mờ của Zhao, Tomizuka và Isaka (*hình 4.11*). Với giả thiết các tham số K_R, K_D bị chặn, tức là

$$K_R \in \left[K_R^{\min}, K_R^{\max}\right] \quad \text{và} \qquad K_D \in \left[K_D^{\min}, K_D^{\max}\right].$$

Zhao, Tomizuka và Isaka đã chuẩn hóa các tham số đó như sau

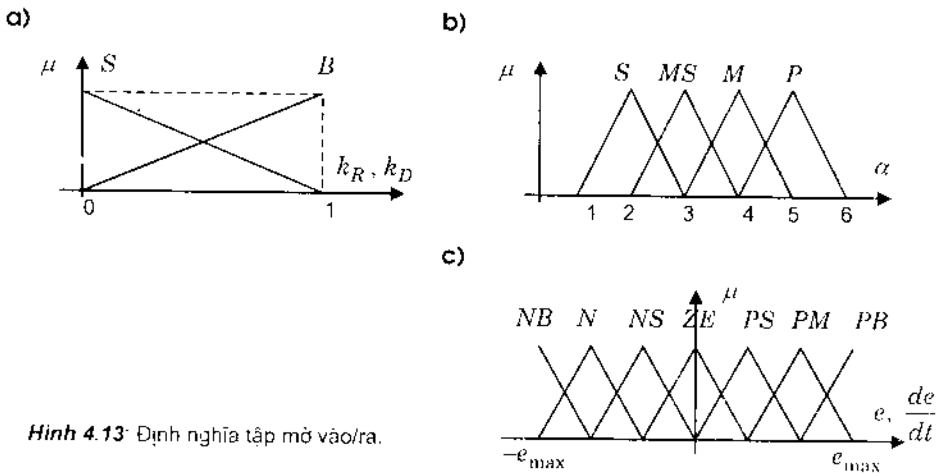
$$k_R = \frac{K_R - K_R^{\min}}{K_R^{\max} - K_R^{\min}}, \; k_D = \frac{K_D - K_D^{\min}}{K_D^{\max} - K_D^{\min}}. \tag{4.4}$$

để có $0 \le k_R, k_D \le 1$.

Như vậy bộ chỉnh định mờ sẽ có hai đầu vào là $e(t)$, $\frac{de(t)}{dt}$ và ba đầu ra là k_R, k_D, α, trong đó

$$\alpha = \frac{T_I}{T_D}. \qquad \Rightarrow \qquad K_I = \frac{K_R^2}{\alpha K_D} \tag{4.5}$$

do đó nó có thể xem nó như ba bộ chỉnh định mờ nhỏ, mỗi bộ có hai đầu vào và một đầu ra (*hình 4.12*).



Hình 4.13: Định nghĩa tập mờ vào/ra.

Biến ngôn ngữ k_R, k_D có hai gia trị mờ

a) B (big) và

b) S (small)

được định nghĩa trong *hình 4.13a)*.

Biến α có bốn giá trị

a) S (small),
b) MS (medium small),
c) M (medium) và
d) B (big)

với những hàm thuộc tương ứng cho trong *hình 4.13b)*.

Sáu giá trị mờ

a) NB (negative small),
b) NM (negative medium),
c) NS (negative small),
d) ZE (zero), PS (positive small),
e) PM (positive medium) và
f) PB (positive big)

của e và $\dot{e}$ cho trong *hình 4.13c)*, trong đó e và $\dot{e}$ được giả thiết là bị chặn

$$-e_{\max} \le e,\ \frac{de}{dt} \le e_{\max}. \tag{4.6}$$

Cả ba khâu chỉnh định mờ trong *hình 4.11* đều sử dụng nguyên tắc độ cao để giải mờ.

Luật điều khiển để chỉnh định được xây dựng theo nguyên tắc: "Tín hiệu điều khiển càng mạnh nếu k_R càng lớn, k_D và α càng nhỏ". Khi giá trị tuyệt đối của sai lệch lớn cần có tín hiệu điều khiển mạnh để đưa nhanh sai lệch về 0. Dựa theo nguyên tắc này mà có được các ma trận quan hệ sau cho từng khâu chỉnh định. Cả ba ma trận quan hệ này đều có dạng gần đối xứng qua đường chéo chính hoặc phụ.

1) Luật chỉnh định k_R

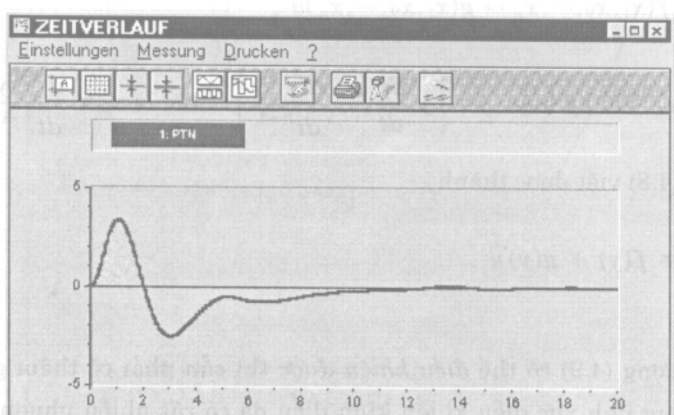
$e(t)$ \ $\dot{e}(t)$	*NB*	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
NB	*B*	*B*	*B*	*B*	*B*	*B*	*B*
NM	*S*	*B*	*B*	*B*	*B*	*B*	*S*
NS	*S*	*S*	*B*	*B*	*B*	*S*	*S*
ZE	*S*	*S*	*S*	*B*	*S*	*S*	*S*
PS	*S*	*S*	*B*	*B*	*B*	*S*	*S*
PM	*S*	*B*	*B*	*B*	*B*	*B*	*S*
PB	*B*	*B*	*B*	*B*	*B*	*B*	*B*

2) Luật chỉnh định k_D

$e(t)$ \ $\dot{e}(t)$	NB	NM	NS	ZE	PS	PM	PB
NB	S	S	S	S	S	S	S
NM	B	B	S	S	S	B	S
NS	B	B	B	B	B	B	B
ZE	B	B	B	B	B	B	B
PS	B	B	B	S	B	B	B
PM	B	B	S	S	S	B	B
PB	S	S	S	S	S	S	S

3) Luật chỉnh định α

$e(t)$ \ $\dot{e}(t)$	NB	NM	NS	ZE	PS	PM	PB
NB	S	S	S	S	S	S	S
NM	MS	MS	S	S	S	MS	MS
NS	M	MS	MS	S	MS	MS	M
ZE	B	M	MS	MS	MS	M	B
PS	M	MS	MS	S	MS	MS	M
PM	MS	MS	S	S	S	MS	MS
PB	S	S	S	S	S	S	S



Hình 4.14: Sai lệch của hệ kín có nhiễu đầu vào với bộ điều khiển PID được chỉnh định tham số theo thuật toán của Zhao, Tomizuka và Isaka. Các tham số ban đầu của bộ PID là K_R=0,03; T_I=1s và T_D=0,2s.

Hình 4.14 là kết quả mô phỏng việc sử dụng các khâu mờ tự chỉnh định tham số cho bộ điều khiển PID theo thuật toán trên. Đối tượng được điều khiển được giả thiết là có mô hình

$$G_S(s) = \frac{27}{(s+1)(s+3)^3} . \tag{4.7}$$

Giá trị vật lý của K_R và K_D là số thực trong khoảng [0, 1]. *Hình 4.14* mô tả sai lệch $e(t)$ của hệ thống có nhiễu $\eta(t)$ phân bố chuẩn và $E\{\eta\}=0$ tại đầu vào. Kết quả mô phỏng cho thấy nhờ có bộ chỉnh định mờ mà tính bền vững của hệ được nâng cao.

4.5 Tổng hợp bộ điều khiển mờ thích nghi

4.5.1 Giới hạn của bài toán

Quay lại xét lớp các đối tượng kiểu SISO đã đề cập đến trong chương 3 với một đầu vào $u \in \mathbb{R}$, một đầu ra $y \in \mathbb{R}$ có n biến trạng thái $x_1, x_2, \cdots, x_n$ biểu diễn được bằng mô hình trạng thái

$$\begin{aligned} \dot{x}_1 &= x_2 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= f(x_1, x_2, \cdots, x_n) + g(x_1, x_2, \cdots, x_n)u \ . \end{aligned} \tag{4.8}$$

trong đó $y = x_1$. Với ký hiệu $\underline{y} = \left(y, \frac{dy}{dt}, \cdots, \frac{d^{n-1}y}{dt^{n-1}} \right)^T$, tức là $x_k = \frac{d^{k-1}y}{dt^{k-1}}$, k=1, 2, $\cdots$, n, công thức (4.8) viết được thành

$$\frac{d^n y}{(dt)^n} = f(\underline{y}) + g(\underline{y})u. \tag{4.9}$$

Để đối tượng (4.9) có thể ***điều khiển được*** thì cần phải có thêm giả thiết rằng $g(\underline{y}) \neq 0$. Trong lĩnh vực điều khiển kinh điển đã có rất nhiều phương pháp tuyến tính đơn giản cho phép tổng hợp bộ điều khiển R để có sai lệch $e(t) \to 0$ (*xem hình 4.15*) như phương pháp cho trước điểm cực của Rosenbrock. Bộ điều khiển R tổng hợp được có

$$u(t) = \frac{1}{g(\underline{y})}\left[-f(\underline{y}) + \frac{d^n x}{dt^n} + \underline{k}^T \underline{e} \right], \tag{4.10}$$

trong đó

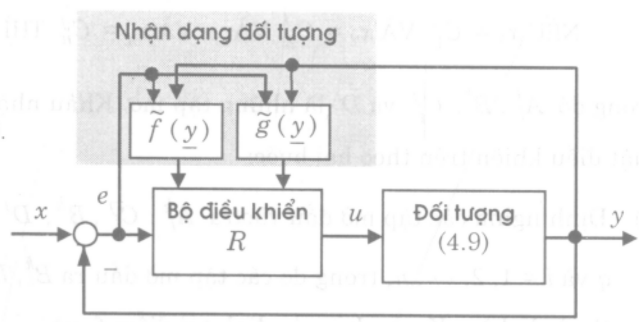
$$\underline{k}^T = \begin{pmatrix} k_1 \\ \vdots \\ k_n \end{pmatrix}, \qquad \underline{e} = \begin{pmatrix} e \\ \dot{e} \\ \vdots \\ \dfrac{d^{n-1}e}{dt^{n-1}} \end{pmatrix}$$

và các hệ số $k_1, k_2, \cdots, k_n \in \mathbb{R}$ được chọn sao cho tất cả các nghiệm của

$$s^n + k_1 s^{n-1} + \cdots + k_n = 0 \tag{4.11a}$$

nằm bên nửa hở mặt phẳng phức bên trái, tức là

$$\mathrm{Re}(s) < 0 \qquad \text{(phần thực của } s \text{ âm).} \tag{4.11b}$$



Hình 4.15: Điều khiển thích nghi.

Thay (4.10) vào (4.9) thì được

$$\frac{d^n e}{dt^n} + k_n \frac{d^{n-1}e}{dt^{n-1}} + \cdots + k_1 e = 0. \tag{4.12}$$

Do có điều kiện (4.11) nên nghiệm $e(t)$ của (4.12) chắc chắn sẽ thỏa mãn

$$\lim_{t\to\infty} e(t) = 0 .$$

Song bài toán tổng hợp điều khiển này chỉ có ý nghĩa nếu (4.9) đã cho, hay nói cách khác hàm $f(\underline{y})$ và $g(\underline{y})$ đã được biết trước mà điều này trong thực tế thường không được thỏa mãn. Với những trường hợp như vậy bộ điều khiển (4.10) chỉ có tác dụng khi $f(\underline{y})$ và $g(\underline{y})$ của đối tượng đã được nhận dạng.

4.5.2 Tổng hợp khâu nhận dạng mờ

Các hàm $f(\underline{y})$ và $g(\underline{y})$ nhận được bằng phương pháp nhận dạng sau đây sẽ được ký hiệu bởi

$$\widetilde{f}(\underline{y}) \text{ và } \widetilde{g}(\underline{y}). \tag{4.13}$$

Về thực chất, trái tim của bộ điều khiển vẫn là khâu được thiết kế theo công thức (4.10), kỹ thuật mờ ở đây chỉ được áp dụng để xác định $\widetilde{f}(\underline{y})$ và $\widetilde{g}(\underline{y})$, do đó bộ điều khiển này thuộc nhóm bộ mờ lai.

Hai hàm $f(\underline{y})$ và $g(\underline{y})$ sẽ được nhận dạng bằng các luật

NẾU $x_1 = A_1^k$ VÀ $x_2 = A_2^k$ VÀ $\cdots$ VÀ $x_n = A_n^k$ THÌ $\widetilde{f} = B^k$, $k = 1, 2, \cdots, p$

NẾU $x_1 = C_1^j$ VÀ $x_2 = C_2^j$ VÀ $\cdots$ VÀ $x_n = C_n^j$ THÌ $\widetilde{g} = D^j$, $j = 1, 2, \cdots, q$

trong đó A_i^k, B^k, C_i^j và D^j là những tập mờ. Khâu nhận dạng được thiết kế từ $p+q$ luật điều khiển trên theo hai bước:

1) Định nghĩa các tập mờ đầu vào/ra A_i^k, C_i^j, B^k, D^j với $k=1,2,\cdots,p$; $j=1, 2, \cdots, q$ và $i = 1, 2, \cdots, n$, trong đó các tập mờ đầu ra B^k, D^j có dạng singleton với hàm thuộc là hàm Kronecker xác định tại điểm f_k và g_j.

2) Xây dựng luật hợp thành R để tính $\widetilde{f}$ và S để tính $\widetilde{g}$ theo nguyên tắc sum-PROD, trong đó phép tính VÀ được thực hiện bằng công thức "tích đại số". Giải mờ được thực hiện bằng phương pháp độ cao.

Đường đặc tính của hai quan hệ truyền đạt $\underline{y} \mapsto \widetilde{f}$ và $\underline{y} \mapsto \widetilde{g}$ được xác định từ (2.11) và (2.23) có dạng như sau (xem phần 2.2)

$$\widetilde{f} = \frac{\sum_{k=1}^{p}\left(f_k \prod_{l=1}^{n} \mu_{A_l^k}(x_l)\right)}{\sum_{k=1}^{p}\prod_{l=1}^{n} \mu_{A_l^k}(x_l)}; \qquad \widetilde{g} = \frac{\sum_{j=1}^{q}\left(g_j \prod_{l=1}^{n} \mu_{B_l^j}(x_l)\right)}{\sum_{j=1}^{q}\prod_{l=1}^{n} \mu_{B_l^j}(x_l)}. \tag{4.14}$$

Để khâu nhận dạng mờ có được tính thích nghi với đối tượng, các điểm f_k , g_j trong (4.14) được xem như là những tham số tự do có thể chỉnh định được. Nếu viết gộp các điểm này lại dưới dạng vector tham số

$$\underline{f} = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} \in \mathbb{R}^n \qquad \text{và} \qquad \underline{g} = \begin{pmatrix} g_1 \\ \vdots \\ g_n \end{pmatrix} \in \mathbb{R}^n$$

thì (4.14) viết được thành

$$\widetilde{f} = \underline{f}^T \cdot \underline{\psi} \quad \text{và} \quad \widetilde{g} = \underline{g}^T \cdot \underline{\xi} \tag{4.15a}$$

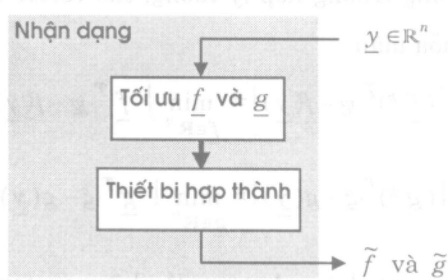
với

$$\underline{\psi} = \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_p \end{pmatrix} \quad , \psi_m = \frac{\prod_{l=1}^{n} \mu_{A_l^m}(x_l)}{\sum_{k=1}^{p}\prod_{l=1}^{n} \mu_{A_l^k}(x_l)}, \qquad m = 1, 2, \cdots, p \tag{4.15b}$$

và

$$\underline{\xi} = \begin{pmatrix} \xi_1 \\ \vdots \\ \xi_q \end{pmatrix} \quad , \xi_m = \frac{\prod_{l=1}^{n} \mu_{B_l^m}(x_l)}{\sum_{k=1}^{p}\prod_{l=1}^{n} \mu_{B_l^j}(x_l)}, \qquad m = 1, 2, \cdots, q \tag{4.15c}$$

Hình 4.16: Các bước nhận dạng $f(\underline{y})$ và $g(\underline{y})$.



Hình 4.16 mô tả quá trình xác định $\widetilde{f}$ và $\widetilde{g}$ theo (4.15), trong đó các vector tham số $\underline{f}$ và $\underline{g}$ được tối ưu từ $\underline{y}$ để bộ điều khiển có tính thích nghi được với đối tượng.

Từ ánh xạ $\underline{y} \mapsto \widetilde{f}$ và $\underline{y} \mapsto \widetilde{g}$ của khâu nhận dạng mờ (4.15a), công thức (4.10) của khâu điều khiển trở thành

$$u(t) = \frac{1}{\widetilde{g}(\underline{y})}\left[-\widetilde{f}(\underline{y}) + \frac{d^n x}{dt^n} + \underline{k}^T\underline{e}\right]. \tag{4.16}$$

Thay (4.16) vào (4.9) sẽ có được phương trình sai lệch

$$\frac{d^n e}{dt^n} = -\underline{k}^T\underline{e} + \left[\widetilde{f}(\underline{y}) - f(\underline{y})\right] + \left[\widetilde{g}(\underline{y}) - g(\underline{y})\right]u\,. \tag{4.17}$$

Nếu sử dụng các ký hiệu

$$\Phi = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & & & & & \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_{n-1} & -k_n \end{pmatrix}, \quad \underline{b} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tag{4.18}$$

thì (4.17) viết được thành dạng phương trình trạng thái của một hệ tuyến tính như sau

$$\underline{\dot{e}} = \Phi\cdot\underline{e} + \underline{b}\cdot\{[\,\underline{f}^T\cdot\underline{\psi} - f(\underline{y})] + [\,\underline{g}^T\cdot\underline{\xi} - g(\underline{y})]u\}\,. \tag{4.19}$$

4.5.3 Xác định thích nghi các vector tham số

Trong trường hợp lý tưởng, các vector tham số nhận dạng được $\underline{f}^*$ và $\underline{g}^*$ sẽ phải thỏa mãn

$$|(\underline{f}^*)^T\cdot\underline{\psi} - f(\underline{y})| = \min_{\underline{f}\in\mathbb{R}^n}|\,\underline{f}^T\cdot\underline{\psi} - f(\underline{y})| \tag{4.20a}$$

$$|(\underline{g}^*)^T\cdot\underline{\xi} - g(\underline{y})| = \min_{\underline{g}\in\mathbb{R}^n}|\,\underline{g}^T\cdot\underline{\xi} - g(\underline{y})| \tag{4.20b}$$

với mọi $\underline{y}$ và chúng được gọi là những vector tham số tối ưu.

Viết lại (4.19) với các vector tham số tối ưu $\underline{f}^*$ và $\underline{g}^*$ cho trong (4.20) thì

$$\underline{\dot{e}} = \Phi\cdot\underline{e} + \underline{b}\cdot[(\underline{f} - \underline{f}^*)^T\cdot\underline{\psi} + (\underline{g} - \underline{g}^*)^T\cdot\underline{\xi}\cdot u + w], \tag{4.21}$$

trong đó

$$w = [(\underline{f}^*)^T\cdot\underline{\psi} - f(y)] + [(\underline{g}^*)^T\cdot\underline{\xi} - g(y)]u. \tag{4.22}$$

Vì $f(y)$, $g(y)$ không biết trước, nên việc xác định $\underline{f}^*$ và $\underline{g}^*$ theo (4.20) không thể thực hiện được. Bài toán xác định $\underline{f}^*$ và $\underline{g}^*$ do đó phải chuyển sang bài toán tối ưu là tìm $\underline{f}$, $\underline{g}$ để có được giá trị cực tiểu của sai lệch $\underline{e}$ và các sai số

$$|\underline{f} - \underline{f}^*|, \quad |\underline{g} - \underline{g}^*|.$$

Điều đó đồng nghĩa với việc xác định điều kiện cho $\underline{f}$ và $\underline{g}$ để hàm Lyapunov (xem thêm chương 5 về hàm Lyapunov)

$$V = \underline{e}^T P\underline{e} + \frac{1}{\alpha}(\underline{f} - \underline{f}^*)^T(\underline{f} - \underline{f}^*) + \frac{1}{\beta}(\underline{g} - \underline{g}^*)^T(\underline{g} - \underline{g}^*) \underset{\underline{f},\underline{g}}{\rightarrow} \min!. \tag{4.23}$$

tức là V dọc theo (4.22) sẽ hướng về điểm cực tiểu, trong đó α, β là hai số thực dương có vai trò như những hệ số trọng lượng và P là một ma trận vuông $n\times n$, xác định dương thoả mãn phương trình Lyapunov

$$\Phi^T P + P\Phi = -Q. \tag{4.24}$$

trong đó Q là ma trận xác định dương bất kỳ.

Để có được (4.23), các vector tham số $\underline{f}$ và $\underline{g}$ phải được xác định sao cho đạo hàm $\dot{V}$ luôn có giá trị âm. Từ (4.23) và (4.23) suy ra

$$\dot{V} = -\underline{e}^T Q\underline{e} + \underline{e}^T P\underline{b}\cdot\underline{w} + 2(\underline{f} - \underline{f}^*)^T\cdot\left[\dot{\underline{f}} - \alpha\,\underline{e}^T P\underline{b}\,\underline{\psi}\right] +$$
$$+ 2(\underline{g} - \underline{g}^*)^T\cdot\left[\dot{\underline{g}} + \beta\,\underline{e}^T P\underline{b}\,\underline{\xi}\,u\right].$$

Số hạng thứ nhất là một số âm. Số hạng thứ hai phụ thuộc vào vector $\underline{w}$, mà theo (4.22) thì độ lớn của $|\underline{w}|$ phụ thuộc vào hệ nhận dạng mờ (4.14), do đó với công thức ước lượng sai số (3.13) bao giờ cũng xác định được các tập mờ đầu vào A_i^k, C_i^j để cho $|\underline{w}| < \varepsilon$, với ε là một số dương bé tuỳ ý.

Bởi vậy để có $\dot{V} < 0$, chỉ cần xác định vector tham số $\underline{f}$, $\underline{g}$ để hai số hạng sau bằng 0 là đủ, tức là

$$\dot{\underline{f}} = -\alpha\,\underline{e}^T P\underline{b}\,\underline{\psi} \qquad\qquad \dot{\underline{g}} = -\beta\,\underline{e}^T P\underline{b}\,\underline{\xi}\cdot u. \tag{4.25}$$

Hai công thức trên cho phép xác định vector tham số $\underline{f}$ và $\underline{g}$ thích nghi với đối tượng. Giá trị đầu $\underline{f}_0$ và $\underline{g}_0$ phải được lấy từ đối tượng (từ kinh nghiệm và hiểu biết về đối tượng). Nhận dạng hệ thống theo thuật toán thích nghi được thực hiện với các bước sau đây:

1) Xác định hai vector khởi đầu $\underline{f}_0$ và $\underline{g}_0$. Chọn ma trận P và các hệ số trọng lượng α, β thích hợp.

2) Thực hiện xoay vòng cho $k = 0, 1, 2 \cdots$ các bước

   a) Lấy $\underline{y} = \begin{pmatrix} y \\ \dot{y} \\ \vdots \\ \dfrac{d^{n-1}y}{dt^{n-1}} \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ từ đầu ra của đối tượng và tính các vector ψ, ξ theo (4.15b) và (4.15c).

   b) Tính $\widetilde{f}$ và $\widetilde{g}$ từ $\underline{f}_k$ và $\underline{g}_k$ theo công thức (4.15a).

   c) Tính $u(t)$ từ $\widetilde{f}$ và $\widetilde{g}$ theo (4.16) và đưa vào điều khiển đối tượng.

   d) Tính $\underline{f}_{k+1}$ và $\underline{g}_{k+1}$ theo (4.25) với giá trị đầu $\underline{f}_k$ và $\underline{g}_k$.

   e) Gán $k := k+1$ và quay lại bước a).

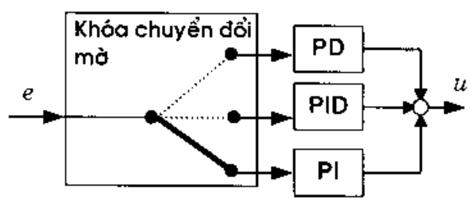
Câu hỏi ôn tập và bài tập

1) Hãy thiết kế khóa chuyển đổi mờ cho bộ điều khiển liên hợp gồm một khoá chuyển đổi mờ và 3 bộ điều khiển theo nguyên tắc:

   a) Nếu sai lệch e nằm trong khoảng $50 \le |e| \le 100$ đối tượng được điều khiển bằng bộ điều khiển PD.

   b) Nếu sai lệch nằm trong khoảng $10 \le |e| \le 50$ đối tượng được điều khiển bằng bộ điều khiển PID

   c) Nếu sai lệch nằm trong khoảng $|e| \le 10$ đối tượng được điều khiển theo luật điều khiển PI.



Hình 4.17: Cho bài tập 1.

2) Cho đối tượng có mô hình toán học dưới dạng hàm truyền đạt:

$$G(s) = \frac{120}{1 + 1000s}$$

được điều khiển bằng bộ điều khiển trên, hãy đánh giá chất lượng của hệ thống.

3) Thiết kế thuật toán chỉnh định thích nghi tham số của một bộ điều khiển PI theo nguyên lý mờ và kiểm tra kết quả tính toán qua đối tượng trong bài 1 với một nhiễu hằng số đưa vào đầu vào đối tượng. Thay đổi giá trị của nhiễu và kiểm tra khả năng hội tụ của thuật toán.

5 TÍNH ỔN ĐỊNH CỦA HỆ ĐIỀU KHIỂN MỜ

5.1 Những khái niệm cơ bản

Điều kiện quan trọng nhất khi thiết kế hệ thống là phải đảm bảo tính ổn định của hệ thống. Một hệ thống điều khiển không thể không ổn định, vì khi hệ thống đã không ổn định thì không thể khảo sát được các chất lượng khác của hệ thống như chất lượng động, chất lượng tĩnh....

5.1.1 Định nghĩa

Cho một hệ thống có m đầu vào $\underline{u}(t) \in L^m$, r đầu ra $\underline{y}(t) \in L^r$ và n biến trạng thái $x_1, x_2, \cdots, x_n$, viết chung lại thành $\underline{x} \in L^n$, được mô tả bằng phương trình trạng thái

$$\frac{d\underline{x}}{dt} = \underline{f}(\underline{x}, \underline{u}, t), \tag{5.1a}$$

$$\underline{y} = \underline{g}(\underline{x}, \underline{u}, t), \tag{5.1b}$$

trong đó ký hiệu L^k dùng để chỉ không gian các vector k chiều có phần tử là những hàm thời gian. Nếu trên L^n có chuẩn $\|x\|_p$, tức là tồn tại

$$\|x\|_p = \sum_{i=1}^{n} \left(\int_0^{\infty} |x_i(t)|^p dt \right)^{\frac{1}{p}} < \infty ,$$

với p là một số thực dương thì không gian L^n sẽ được gọi là không gian chuẩn p và viết thành L_p^n.

Trường hợp $p = \infty$ thì chuẩn $\|x\|_\infty$ của $x \in L_\infty^n$ có dạng

$$\|x\|_\infty = \sum_{i=1}^{n} \sup_t |x_i(t)| .$$

Nghiệm của (5.1a) với giá trị ban đầu $\underline{x}_0$ được gọi là *quỹ đạo trạng thái* của hệ tại thời điểm $t = t_0$ đi qua $\underline{x}_0$ và ký hiệu bởi $\underline{x}(\underline{x}_0, t)$.

Định nghĩa 5.1: (Ổn định Lyapunov)

Hệ thống (5.1) với $\underline{u}(t)\equiv\underline{0}$ trở thành

$$\frac{dx}{dt} = f(x,u,t)_{u=0} = \tilde{f}(\underline{x},t)$$

được gọi là ổn định Lyapunov, hay đơn giản là *ổn định tại điểm cân bằng* $\underline{x}_e$ (*equilibium point*), tức là điểm trạng thái thỏa mãn

$$\tilde{\underline{f}}(x_e, t) = \underline{0} \quad , \forall t$$

nếu với $\varepsilon, t_0 > 0$ bất kỳ bao giờ cũng tồn tại $\delta(\varepsilon, t_0)$ phụ thuộc ε và t_0 sao cho

$$\|\underline{x}_0 - \underline{x}_e\| < \delta \Rightarrow \|\underline{x}(\underline{x}_0, t) - \underline{x}_e\| < \varepsilon \text{ , } \forall t \geq t_0. \tag{5.3}$$

Công thức (5.3) trên nói rằng nếu cho trước một lân cận ε của $\underline{x}_e$ thì phải tồn tại một lân cận δ cũng của $\underline{x}_e$ sao cho mọi đường quỹ đạo trạng thái tại thời điểm t_0 đi qua một điểm $\underline{x}_0$ thuộc lân cận δ thì kể từ thời điểm đó sẽ nằm hoàn toàn trong lân cận ε (*hình 5.1*).

Vì $\underline{x}_0 = \underline{x}(\underline{x}_0, t_0)$ nên để có được

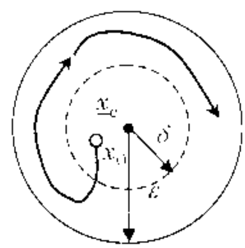
$$\|\underline{x}(\underline{x}_0, t_0) - \underline{x}_e\| < \varepsilon$$

thì lân cận δ phải nằm bên trong lân cận ε.

Nếu mệnh đề kết luận trong (5.3) được thay bằng

$$\lim_{t\to\infty} x(x_0, t) = \underline{x}_e$$

thì hệ (5.1) được gọi là *ổn định tiệm cận* tại $\underline{x}_e$.



Hình 5.1: Ổn định Lyapunov của hệ (5.1) tại điểm cân bằng $\underline{x}_e$.

5.1.2 Những điểm cần lưu ý

Mục đích của toàn bộ chương 5 này là khảo sát tính ổn định của hệ thống mờ dựa trên các tiêu chuẩn xét ổn định kinh điển như tiêu chuẩn Lyapunov, Popov, phương pháp cân bằng điều hòa, ... Khi đối tượng và bộ điều khiển vẫn được coi là tuyến tính hay nói một cách khác là có thể bỏ qua các đặc tính phi tuyến của hệ thống thì có thể kiểm tra tính ổn định của hệ dựa trên các tiêu chuẩn đại số và tần số tuyến tính quen biết. Nhưng khi đã sử dụng bộ điều khiển mờ để điều khiển đối tượng, các tiêu chuẩn cho hệ tuyến tính không còn sử dụng được nữa. Như đã biết trong các chương trước, bộ điều khiển mờ luôn luôn là một bộ điều khiển phi tuyến, do vậy khi sử dụng nó để điều khiển đối tượng thì chính bộ điều khiển này sẽ làm tăng thêm tính phi tuyến của hệ thống. Muốn kiểm tra tính ổn định của hệ điều khiển mờ phải sử dụng các tiêu chuẩn ổn định dành cho hệ phi tuyến và điều này không dễ chịu chút nào, vì

- Với hệ phi tuyến không tồn tại khái niệm ổn định hoàn toàn. Thực chất, các hệ phi tuyến có nhiều điểm cân bằng với các tính chất ổn định khác nhau. Các kết luận về tính ổn định của hệ phi tuyến thường chỉ đúng cho lân cận một điểm cân bằng nhất định của hệ thống.
- Tính ổn định của hệ tuyến tính là tính ổn định tuyệt đối, có nghĩa là mọi chuyển động trong hệ thống đều ổn. Còn trong lý thuyết về hệ phi tuyến các khái niệm về ổn định hoàn toàn mang ý nghĩa khác hẳn và chỉ có thể được định nghĩa cho riêng từng điểm trạng thái.
- Tính phi tuyến mang lại cho hệ thống những đặc tính động học phong phú, cho nên các tiêu chuẩn xét ổn định cho hệ phi tuyến thường chỉ cho các loại đặc tính phi tuyến nhất định và trước đó đã xác định được đặc tính phi tuyến. Trong thực tế thì các đặc tính phi tuyến lại thường không xác định trước được một cách đầy đủ.
- Các tiêu chuẩn xét ổn định cho hệ thống phi tuyến rất phức tạp và khó áp dụng. Thường chúng chỉ được ứng dụng cho các hệ bậc thấp.

Việc xét tính ổn định một hệ mờ phụ thuộc rất nhiều vào phương thức tích hợp giữa đối tượng và bộ điều khiển. Khảo sát tính ổn định của hệ thống với bộ điều khiển mờ và đối tượng mô tả được theo phương pháp kinh điển tương đối đơn giản, vì trong lý thuyết hệ thống điều khiển phi tuyến đã tồn tại rất nhiều tiêu chuẩn trong miền thời gian cũng như trong miền tần số để kiểm tra tính ổn định của các hệ thống này. Còn các trường hợp hệ có bộ điều khiển mờ lai hoặc mô hình đối tượng cũng mờ thì chỉ có thể dự đoán được tính ổn định của hệ thống. Đặc biệt là cho đến nay vẫn chưa tồn tại tiêu chuẩn nào có thể khảo sát được tính ổn định cho các hệ có mô hình của đối tượng là mô hình mờ.

5.2 Khảo sát tính ổn định của hệ mờ

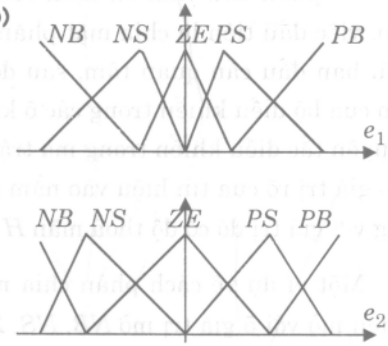
5.2.1 Phương pháp mặt phẳng pha

Phân tích hệ thống điều khiển tự động phi tuyến bằng phương pháp không gian trạng thái là một trong những phương pháp thông dụng từ trước đến nay. Quỹ đạo trạng thái được xây dựng bằng phương pháp đồ thị, đường biểu diễn chuyển động của hệ thống trong không gian trạng thái được gọi là quỹ đạo trạng thái.

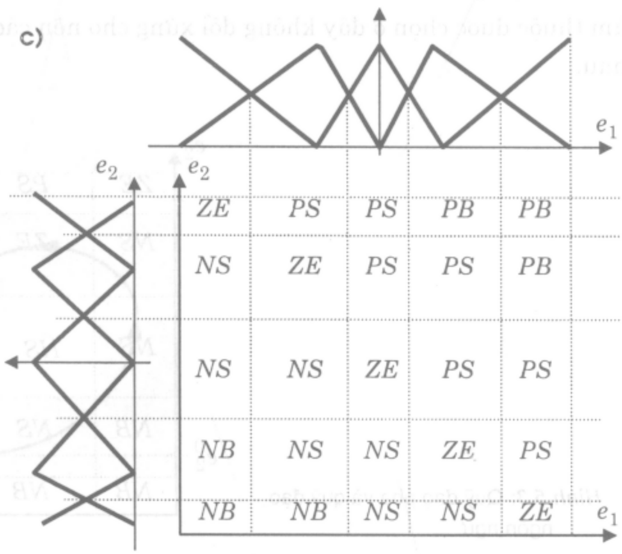
a)

e_2 \ e_1	*NB*	*NS*	*ZE*	*PS*	*PB*
PB	*ZE*	*PS*	*PS*	*PB*	*PB*
PS	*NS*	*ZE*	*PS*	*PS*	*PB*
ZE	*NS*	*NS*	*ZE*	*PS*	*PS*
NS	*NB*	*NS*	*NS*	*ZE*	*PS*
NB	*NB*	*NB*	*NS*	*NS*	*ZE*

b)



c)



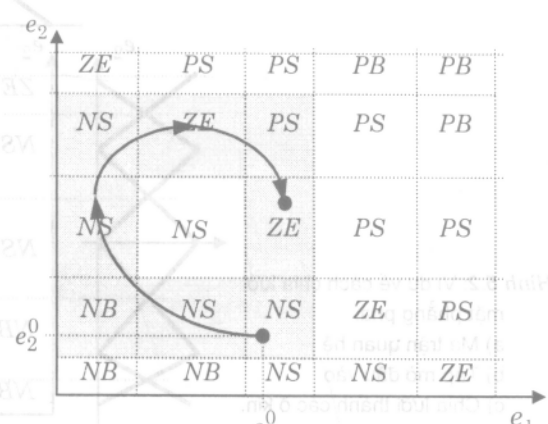
Hình 5.2: Ví dụ về cách chia lưới mặt phẳng pha.
a) Ma trận quan hệ
b) Tập mờ đầu vào
c) Chia lưới thành các ô kín.

Đặc tính động và các tính chất ổn định của hệ thống được xác định từ quỹ đạo trạng thái của hệ thống. Quỹ đạo của hệ thống có hai biến trạng thái biểu diễn trong không gian hai chiều $\mathbb{R}^2$ không có biến thời gian được gọi là *quỹ đạo pha* và không gian hai chiều $\mathbb{R}^2$ lúc này được gọi là *mặt phẳng pha*.

Tính chất phi tuyến của bộ điều khiển cũng như của đối tượng có thể bất kỳ, nhưng mô hình toán học của đối tượng phải biết trước. Đặc tính động học được xác định từ quỹ đạo pha được xây dựng xung quanh điểm làm việc của hệ thống. Sự phân tích này không khẳng định được chính xác tính ổn định của hệ thống, nhưng cũng có thể quan sát được điểm cân bằng của hệ thống và trạng thái ổn định của hệ thống tại điểm cân bằng đó.

Để phân tích tính ổn định của một hệ thống có bộ điều khiển mờ với hai đầu vào, việc đầu tiên là chia mặt phẳng pha bằng một lưới đủ chứa hết các điểm trạng thái ban đầu cần quan tâm, sau đó xây dựng quỹ đạo của hai biến ngôn ngữ đầu vào của bộ điều khiển trong các ô kín này của lưới. Mỗi một ô kín tương ứng với một nguyên tắc điều khiển trong ma trận quan hệ. Giới hạn giữa các ô kín được chọn sao cho giá trị rõ của tín hiệu vào nằm trọn trong một ô và nguyên tắc điều khiển tương ứng với giá trị đó có độ thoả mãn H cao nhất.

Một ví dụ về cách phân chia mặt phẳng pha thành các ô kín cho một bộ điều khiển mờ với 5 giá trị mờ *NB*, *NS*, *ZE*, *PS* và *PB* cho mỗi biến ngôn ngữ của hai tín hiệu vào e_1, e_2 được biểu diễn dưới dạng ma trận quan hệ cho trong *hình 5.2*. Vì các hàm thuộc được chọn ở đây không đối xứng cho nên các ô kín này có kích thước khác nhau.



Hình 5.3: Quỹ đạo pha và quỹ đạo ngôn ngữ.

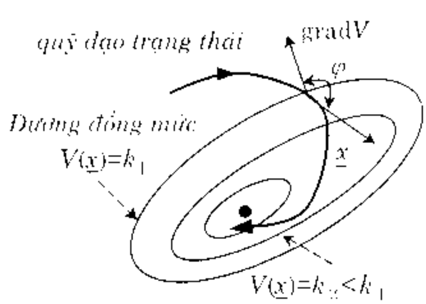
Sau khi đã chia lưới mặt phẳng pha, quỹ đạo pha cho hệ thống trong mặt phẳng pha từ một điểm đầu $\underline{x}_0 = \begin{pmatrix} e_1^0 \\ e_2^0 \end{pmatrix}$ sẽ được xây dựng dần dần từ ô này sang ô khác. Nếu hệ ổn định tại một điểm cân bằng thì kể từ một điểm thời gian nào đó quỹ đạo pha sẽ không ra khỏi ô có chứa điểm cân bằng đó và ô này có tên gọi là *ô cân bằng*. Ngược lại khi mọi quỹ đạo pha có phần cuối nằm gọn trong một ô cân bằng thì hệ ổn định (tuyệt đối). Cách biểu diễn quỹ đạo pha trên lưới mặt phẳng pha như vậy còn cho biết thêm những luật điều khiển R_k nào đã tham gia vào điều khiển quỹ đạo pha đó. Do vậy ứng với mỗi một quỹ đạo pha của hệ thống là một *quỹ đạo ngôn ngữ* gồm các luật điều khiển của các ô kín mà quỹ đạo pha lần lượt đã đi qua. *Hình 5.3* biểu diễn quỹ đạo ngôn ngữ là các ô có mầu tối cho một trường hợp cụ thể. Kết quả mô phỏng của quỹ đạo pha không những cho biết trạng thái ổn định của hệ thống mà còn cho những kinh nghiệm về cách chọn độ phân giải cho biến đầu vào. Nên chọn độ phân giải sao cho lưới mặt phẳng pha đủ mịn để các quỹ đạo ngôn ngữ đi qua nhiều giá trị mờ đầu vào khác nhau, ví dụ đi qua ít nhất hai giá trị mờ *ZE* và *PS* của e_2 hoặc ít nhất hai giá trị *NS* và *ZE* của e_1.

5.2.2 Phương pháp Lyapunov trực tiếp

Phương pháp Lyapunov trực tiếp là một công cụ quan trọng của toàn bộ lý thuyết về ổn định của Lyapunốp. Xét lớp các hệ động học autonom không bị kích thích có phương trình trạng thái

$$\frac{d\underline{x}}{dt} = \underline{f}(\underline{x},\underline{u})\Big|_{\underline{u}=\underline{0}} = \underline{\tilde{f}}(\underline{x}), \qquad (5.4)$$

trong đó $\underline{x}$ là vector trạng thái và $\underline{f}, \underline{\tilde{f}}$ là vector các hàm liên tục mô tả đặc tính động của hệ thống. Vector hàm có thể tuyến tính hoặc phi tuyến.



Hình 5.4: Mô tả tiêu chuẩn Lyapunov.

Tư tưởng của phương pháp Lyapunov được xây dựng trên cơ sở bảo tồn năng lượng của một hệ vật lý. Hệ vật lý này có năng lượng toàn bộ ở trạng thái cân bằng bằng 0, ở xung quanh vị trí cân bằng, năng lượng của hệ thống lớn hơn không và có xu thế tiến đến không. Trạng thái cân bằng được gọi là ổn định nếu ở vùng xung quanh điểm cân bằng của hệ thống giá trị của hàm giảm dần hoặc không thay đổi.

Để kiểm tra được tính ổn định của hệ thống tại vị trí cân bằng $\underline{x}_e$, cần phải xác định được hàm năng lượng $V(\underline{x})$, gọi là hàm Lyapunov, phụ thuộc vào trạng thái của hệ thống. Không mất tính tổng quát nếu giả sử rằng $\underline{x}_e$ là điểm gốc $\underline{0}$ của không gian trạng thái và trong lân cận $\underline{0}$ hàm $V(\underline{x})$ xác định dương. Vậy thì vector

$$\mathrm{grad}V = \left(\frac{\partial V}{\partial x_1},\dots,\frac{\partial V}{\partial x_n}\right)^T$$

luôn hướng ra xa điểm gốc $\underline{0}$, do đó nếu vector grad$V$ và vector $\dot{\underline{x}}$, mà nó chính là tiếp tuyến của quỹ đạo pha của hệ (5.4), lập với nhau một góc φ không nhỏ hơn 90^0 thì quỹ đạo pha $\underline{x}(\underline{x}_0, t)$ luôn có hướng về gốc tọa độ (*hình 5.4*). Điều này tương đương với

$$\frac{dV}{dt} = (\mathrm{grad}V)^T\frac{d\underline{x}}{dt} = |\mathrm{grad}V|\cdot\left|\frac{d\underline{x}}{dt}\right|\cdot\cos\varphi < 0. \qquad (5.5)$$

Suy ra:

Định lý 5.1

Nếu tồn tại hàm Lyapunov $V(\underline{x})$, thoả mãn các điều kiện:

a) xác định dương, tức là $V(\underline{x}) > 0$ với $\underline{x} \neq 0$ và $V(\underline{x}) = 0 \Leftrightarrow \underline{x} = \underline{0}$,

b) $\dfrac{dV(\underline{x})}{dt} \le 0$.

thì hệ (5.4) sẽ ổn định tại điểm $\underline{0}$.

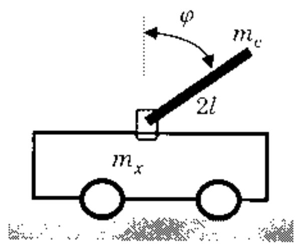
Định lý 5.1 là một điều kiện đủ để hệ (5.4) ổn định tại $\underline{0}$. Việc không tìm thấy một hàm $V(\underline{x})$ thỏa mãn hai điều kiện trên không khẳng định được là hệ (5.4) không ổn định tại $\underline{0}$.

Vì tính phi tuyến của một hệ là một tính chất hoàn toàn tự nhiên và bất kỳ, do vậy tiêu chuẩn Lyapunov là một công cụ mạnh và toàn năng để phân tích tính ổn định của hệ thống phi tuyến. Nhược điểm chính hạn chế sự ứng dụng của tiêu chuẩn này ở chỗ tìm ra được một hàm Lyapunov thích hợp. Thường năng lượng của hệ thống tỷ lệ với bình phương biên độ tín hiệu, nên hàm Lyapunov hay được sử dụng có dạng toàn phương:

$$V(\underline{x}) = \underline{x}^T R\, \underline{x}. \tag{5.6}$$

trong đó ma trận R phải là ma trận xác định dương. Trong nhiều trường hợp đơn giản ma trận R được chọn là ma trận đường chéo.

Hãy xét một ví dụ có đối tượng điều khiển là một con lắc được đặt trên một chiếc xe chuyển động được như *hình 5.5* mô tả. Trọng lượng của xe là m_x. Con lắc có trọng lượng m_c phân đều trên một độ dài $2l$. Bài toán được đặt ra ở đây là điều khiển chiếc xe chuyển động đến một vị trí A dưới tác dụng của một lực u, sao cho con lắc luôn giữ được trạng thái sai lệch góc $\varphi=0$.



Hình 5.5: Đối tượng điều khiển là xe con lắc.

Ở đây, trước tiên chưa quan tâm đến việc thiết kế một bộ điều khiển mờ như thế nào cho phù hợp với nhiệm vụ điều khiển đặt ra, điều quan tâm là làm thế nào bằng phương pháp Lyapunov trược tiếp có thể rút ra kết luận về tính ổn định của hệ thống. Trước tiên là phải xây dựng mô hình toán học cho hệ xe con lắc. Dựa vào các phương trình cân bằng lực moment ta có được mô hình phi tuyến

$$\begin{cases} \dfrac{d}{dt}x_1 = x_2 \\ \dfrac{d}{dt}x_2 = \dfrac{g\sin x_1 - \dfrac{m_x l}{m_x + m_c}\sin x_1 \cos(x_1 x_2^2) - \dfrac{1}{m_x + m_c} u\cos x_1}{\dfrac{4}{3}l - \dfrac{m_x l}{m_x + m_c}\cos^2 x_1} \end{cases} \tag{5.7}$$

trong đó $x_1 = \varphi$. Hàm Lyapunov được chọn có dạng sau:

$$V(\underline{x}) = \frac{1}{2}(x_1^2 + x_2^2). \tag{5.8}$$

Với hàm Lyapunov như vậy dương nhiên điều kiện 1 được thỏa mãn, chỉ còn phải kiểm tra điều kiện 2. Từ (5.5), (5.6) và (5.7) có được

$$\frac{d}{dt}V(\underline{x}) = (\mathrm{grad}V)^T\frac{d\underline{x}}{dt} = (x_1, x_2)\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} =$$

$$x_2\left[\frac{x_1\frac{4}{3}l - x_1\frac{m_x l}{m_x+m_c}\cos^2 x_1 + g\sin x_1 - \frac{m_x l}{m_x+m_c}\sin x_1\cos x_1 x_2^2 + \frac{1}{m_x+m_c}u\cos x_1}{\frac{4}{3}l - \frac{m_x l}{m_x+m_c}\cos^2 x_1}\right]$$

Mẫu của biểu thức trong phương trình trên luôn dương, nên việc xét dấu của biểu thức trên chỉ còn phụ thuộc vào tử số của biểu thức. Nếu x_2 dương, thì biểu thức trong ngoặc phải âm, còn nếu x_2 âm thì biểu thức trong ngoặc phải dương.

Dựa vào điều đó, tín hiệu điều khiển u phải thỏa mãn

$$u > \frac{m_x+m_c}{\cos x_1}\left[x_1\frac{4}{3}l - x_1\frac{m_x l}{m_x+m_c}\cos^2 x_1 + g\sin x_1 - \frac{m_x l}{m_x+m_c}\sin x_1\cos(x_1 x_2^2)\right]$$

khi $x_2 > 0$ và

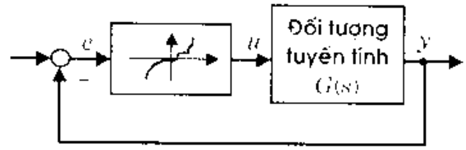
$$u < \frac{m_x+m_c}{\cos x_1}\left[x_1\frac{4}{3}l - x_1\frac{m_x l}{m_x+m_c}\cos^2 x_1 + g\sin x_1 - \frac{m_x l}{m_x+m_c}\sin x_1\cos(x_1 x_2^2)\right]$$

khi $x_2 < 0$.

Những điều kiện trên đây cho phép tổng hợp bộ điều khiển mờ để xác định u theo x_1, x_2 của đối tượng sao cho hệ ổn định.

5.2.3 Tiêu chuẩn ổn định tần số của Popov

Tiêu chuẩn ổn định Popov là phương pháp xét ổn định cho hệ thống ở miền tần số. Tiêu chuẩn này thuận tiện trong việc xét ổn định của một hệ thống điều khiển cơ bản bao gồm một hàm truyền đạt tuyến tính $G(s)$ và một đặc tính phi tuyến ở chế độ tĩnh $u(e)$ là thành phần sinh ra tính phi tuyến của bộ điều khiển (*hình 5.6*).



Hình 5.6 Hệ phi tuyến có đối tượng tuyến tính

Điều kiện để có thể áp dụng được tiêu chuẩn Popov là đối tượng $G(s)$ phải tuyến tính và có khả năng lọc các tần số cao. Tiêu chuẩn Popov sử dụng khái niệm *hàm hai cực*, tức là những hàm phức dạng $G(s) = \frac{Z(s)}{N(s)}$, trong đó $Z(s)$, $N(s)$ là những đa thức của biến p không có chung nghiệm với các hệ số là những số thực và từ $\mathrm{Re}(s)>0$ suy ra được $\mathrm{Re}\big(G(s)\big)>0$.

Định lý 5.2

Để hàm truyền đạt $G(s)$ của một hệ tuyến tính, tham số tập trung là hàm hai cực thì cần và đủ là

a) $G(j\omega)$ có phần thực dương khi $\omega>0$.

b) Mọi điểm cực của $G(s)$ nằm bên trái trục ảo. Nếu có điểm cực s_k nằm trên trục ảo thì $\lim_{s\to s_k}(s-s_k)G(s)$ là số dương hữu hạn.

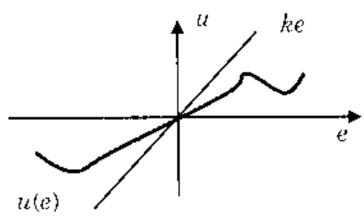
Dựa vào tính chất trên, năm 1959 Popov đã đưa ra tiêu chuẩn xét ổn định một hệ kín có đối tượng tuyến tính mô tả trong *hình 5.6* được phát biểu như sau:

Định lý 5.3: (Điều kiện đủ)

Hệ kín trong *hình 5.6* với đối tượng tuyến tính $G(s)$ và khâu điều khiển tĩnh $u(e)$ liên tục từng đoạn có $u(0)=0$ sẽ ổn định tiệm cận tuyệt đối nếu

a) tồn tại một số $k>0$ sao cho $0<\frac{u(e)}{e}<k$ với $e\neq 0$ và

b) tồn tại một số dương α sao cho hàm phức $F(s)=(1+\alpha s)G(s)+\frac{1}{k}$ là hàm hai cực.



Hình 5.7: Miền ổn định tiệm cận tuyệt đối.

Điểm đặc biệt của tiêu chuẩn Popov là có thể khảo sát tính ổn định tiệm cận tuyệt đối của hệ trong *hình 5.6* cho một lớp các khâu điều khiển phi tuyến tĩnh có đường đặc tính $u(e)$ giới hạn trong hai đoạn thẳng có độ nghiêng là 0 và k gọi là góc $(0, k)$ (*hình 5.7*). Góc "ổn định" $(0, k)$ hoàn toàn có thể được mở rộng cho trường hợp tổng quát (k_1, k_2) bằng cách có thể đưa (k_1, k_2) về góc $(0, k)$ quen biết theo phương pháp chuyển vị.

Điều kiện ổn định của Popov, cũng giống như tiêu chuẩn Lyapunov trực tiếp, chỉ là điều kiện đủ để xét ổn định cho hệ thống điều khiển phi tuyến. Do đó nếu đặc tính khâu phi tuyến nằm ngoài góc $(0, k)$ thì hệ vẫn có khả năng ổn định.

Nếu có thêm điều kiện là hàm truyền của đối tượng $G(s)$ ổn định thì theo định lý 5.2, để hàm $F(s)$ là hàm hai cực chỉ còn cần kiểm tra xem

$$\mathrm{Re}\left[(1+\alpha j\omega)G(j\omega)+\frac{1}{k}\right]\geq 0 \tag{5.9}$$

có được thỏa mãn hay không. Để làm được việc đó trước hết khai triển bất phương trình Popov (5.9) về dạng sau:

$$\alpha X(\omega)\ -\ R(\omega)\ \leq\frac{1}{k}, \tag{5.10a}$$

với

$$X(\omega)=\omega\cdot\mathrm{Im}\big(G(j\omega)\big),\qquad R(\omega)=\mathrm{Re}\big(G(j\omega)\big), \tag{5.10b}$$

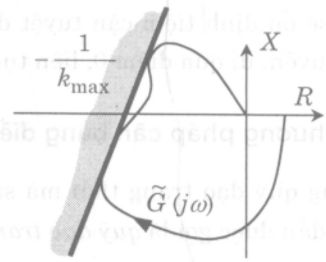
trong đó ký hiệu $\mathrm{Re}(\cdot)$ chỉ phần thực và $\mathrm{Im}(\cdot)$ chỉ phần ảo của một số phức. Biểu diễn (5.10) trên mặt phẳng có hai trục tọa độ X, R thì tất cả các điểm của mặt phẳng thỏa mãn bất phương trình Popov (5.9) là các điểm nằm bên phải của đường thẳng

$$\alpha\cdot X\ -\ R\ =\ \frac{1}{k}, \tag{5.11}$$

đường thẳng này, có tên là *đường thẳng Popov*, cắt trục R tại điểm $-\frac{1}{k}$ và có độ nghiêng bằng $\frac{1}{\alpha}$. Đặc tính tần số

$$\widetilde{G}(j\omega)=R+jX=\mathrm{Re}\big(G(j\omega)\big)+j\omega\,\mathrm{Im}\big(G(j\omega)\big) \tag{5.12}$$

được gọi là *đặc tính tần số biến dạng* hay *đặc tính tần số Popov*. Đường đặc tính tần số biến dạng có phần thực trùng với đặc tính tần số của hệ thống $G(j\omega)$, phần ảo bằng phần ảo của đặc tính tần số nhân với tần số ω. Dựa vào (5.10), (5.11) và (5.12) mà có được hệ quả sau: Hệ kín cho trong *hình 5.6* với đối tượng $G(s)$ là một khâu *tuyến tính ổn định* sẽ ổn định tiệm cận tuyệt đối với những bộ điều khiển phi tuyến tĩnh có đường đặc tính $u(e)$ thỏa mãn $u(0)=0$ và $0 < \frac{u(e)}{e} < k_{max}$, $e \neq 0$, trong đó $-1/k_{max}$ là giao điểm giữa trục R và đường thẳng tiếp xúc với đường đặc tính biến dạng $\widetilde{G}(j\omega)$ của đối tượng.



Hình 5.8: Xác định tính ổn định theo tiêu chuẩn Popov.

Phương pháp dùng đồ thị để kiểm tra tính ổn định của hệ thống theo tiêu chuẩn Popov được thực hiện theo trình tự như sau:

- Xây dựng biểu đồ vector đặc tính tần số Popov $\widetilde{G}(j\omega)$ của đối tượng.
- Nếu muốn xác định trị số k_{max} tới hạn, dựng một đường thẳng có độ nghiêng bất kỳ tiếp xúc với biểu đồ vector đặc tính tần số Popov ít nhất tại một điểm còn các điểm còn lại của đặc tính tần số Popov nằm hoàn toàn bên phải của đường thẳng này. Giao điểm của đường tiếp tuyến với trục hoành chính là giá trị $1/k_{max}$ tới hạn. Hệ thống sẽ ổn định tuyệt đối với họ đặc tính phi tuyến nằm trong góc $(0, k_{max})$. *Hình 5.8* minh họa cho việc xác định k_{max}.

Khi sử dụng tiêu chuẩn Popov không nhất thiết phải xây dựng một cách chính xác đặc tính tần số của hệ thống mà chỉ cần xây dựng đặc tính tần số thực nghiệm là đủ.

Dưới đây là một ví dụ về khảo sát tính ổn định của hệ thống bằng tiêu chuẩn Popov. Hệ thống được khảo sát ở đây là một hệ phi tuyến có phần tuyến tính là một khâu bậc 3 với hàm truyền đạt

$$G(s) = \frac{1}{(s+1)(s^2+s+1)}.$$

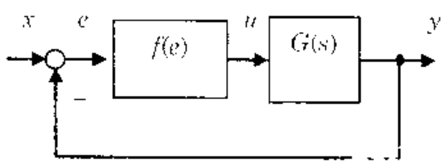
Để xác định được góc $(0, k)$ cần xây dựng đặc tính tần số cho phần tuyến tính bằng cách thay $s=j\omega$ vào hàm truyền đạt của hệ thống. Tách phần thực và phần ảo của $G(j\omega)$, nhân phần ảo với tần số ω để thu được đặc tính tần số Popov $\tilde{G}(j\omega)$. Sau đó từ bên trái của biểu đồ đặc tính tần số Popov $\tilde{G}(j\omega)$ kẻ một đường thẳng có độ nghiêng thích hợp sao cho đường thẳng này tiếp xúc với biểu đồ đặc tính tần số Popov $\tilde{G}(j\omega)$ và cắt trục hoành tại một điểm có khoảng cách nhỏ nhất so với gốc tọa độ. Chẳng hạn điểm cắt với trục hoành có giá trị ≈–0.33 thì góc Popov cần tìm sẽ là $k=3$. Hệ sẽ ổn định tiệm cận tuyệt đối ứng với bộ điều khiển có đường đặc tính là hàm phi tuyến, đi qua điểm 0, liên tục từng đoạn và nằm trong góc (0, 3).

5.2.4 Phương pháp cân bằng điều hòa

Những quỹ đạo trạng thái mà sau một khoảng thời gian hữu hạn lại quay về điểm ban đầu được gọi là *quỹ đạo trạng thái khép kín*.

Một quỹ đạo khép kín được gọi là ổn định, nếu tất cả các quỹ đạo trạng thái lân cận đều tiệm cận về nó. Ngược lại khi có một quỹ đạo trạng thái xuất phát từ nó ra lân cận xung quanh thì quỹ đạo giới hạn đó được gọi là không ổn định. Hệ có quỹ đạo khép kín được gọi là *hệ có khả năng tự dao động*.

Phương pháp hiện nay thường hay được dùng để khảo sát tính ổn định của một quỹ đạo khép kín là phương pháp cân bằng điều hòa trên cơ sở xấp xỉ hệ đó bằng những hàm điều hòa. Phương pháp này thuận tiện cho việc khảo sát các hệ thống phi tuyến giống như mô tả trong *hình 5.9* gồm một bộ điều khiển phi tuyến tĩnh có đường đặc tính vào/ra $f(e)$ và đối tượng tuyến tính với hàm truyền đạt $G(s)$ có khả năng lọc tần số cao.



Hình 5.9 Hệ phi tuyến có bộ điều khiển phi tuyến tĩnh $f(e)$ và đối tượng tuyến tính $G(s)$.

Giả sử rằng hệ cho trong *hình 5.9* tự dao động. Tín hiệu chủ đạo x là một tín hiệu hằng. Trong chế độ tự dao động, tín hiệu đầu vào bộ điều khiển $e(t)$ được giả thiết rằng có thể xấp xỉ bằng hàm điều hòa

$$e(t) \approx E_0 + E_1 \cos \omega t \qquad \text{với } E_0, E_1 \in \mathbb{R}. \qquad (5.13)$$

Tín hiệu đầu ra $u(t) = f(E_0 + E_1\cos\omega t)$ cũng là một hàm tuần hoàn với chu kỳ $2\pi/\omega$. Khai triển $u(t)$ thành chuỗi Fourier và chỉ lấy xấp xỉ $u(t)$ là hai thành phần đầu tiên sẽ có

$$u(t) \approx U_0 + \mathrm{Re}\left(U_1 e^{j\omega t}\right). \tag{5.14}$$

với

$$U_0 = \frac{1}{2\pi}\int_0^{2\pi} f(E_0 + E_1\cos\tau)\,d\tau \quad , \quad U_1 = \frac{1}{\pi}\int_0^{2\pi} f(E_0 + E_1\cos\tau)\,e^{-j\tau}\,d\tau\,.$$

Lập hai hàm

$$N_0 = \frac{U_0}{E_0} = \frac{1}{2\pi E_0}\int_0^{2\pi} f\left(E_0 + E_1\cos\tau\right)d\tau \tag{5.15a}$$

$$N_1 = \frac{U_1}{E_1} = \frac{1}{\pi E_1}\int_0^{2\pi} f\left(E_0 + E_1\cos\tau\right)e^{-j\tau}\,d\tau \tag{5.15b}$$

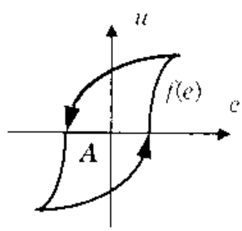
thì nhận thấy N_0 và N_1 đều là những hàm của E_0, E_1 và không phụ thuộc ω. Hai hàm N_0 và N_1 còn có tên gọi là *hệ số khuếch đại phức* của khâu phi tuyến $f(e)$ và có những tính chất sau:

1) N_0 là một số thực. N_1 cũng là một số thực nếu $f(e)$ đơn trị.

2) Nếu $f(e)$ là hàm lẻ và E_0=0 thì U_0=N_0=0 và $u(t) = |U_1|\cos\omega t$.

3) Nếu $f(e)$ là đường đặc tính đa trị, ví dụ như khâu phi tuyến kiểu từ trễ (*hình 5.10*) thì $\mathrm{Im}(N_1) = \pm\dfrac{A}{\pi X_1^2}$, trong đó A là diện tích phần trễ và dấu + cho trường hợp chiều trễ theo chiều kim đồng hồ, dấu − khi chiều trễ ngược với chiều kim đồng hồ.

Hình 5.10: Khâu phi tuyến kiểu từ trễ



Với hệ số khuếch đại N_0, N_1 công thức (5.14) sẽ viết được thành

$$u(t) \approx N_0 E_0 + \mathrm{Re}\left(N_1 E_1 e^{j\omega t}\right).$$

Suy ra

$$x - e = y(t) \approx N_0 E_0 G(0) + \mathrm{Re}\left(N_1 E_1 e^{j\omega t} G(j\omega)\right). \tag{5.16}$$

Thay (5.13) vào (5.16) được và so sánh các hệ số của hai vế sẽ có

$$G(0) = \frac{1}{N_0}\left(\frac{x}{E_0} - 1\right) \quad \text{và} \quad 1 + N_1 G(j\omega) = 0. \tag{5.17}$$

Từ một trong hai phương trình của (5.17) rút ra được quan hệ $E_0 = E(E_1)$ và khi thay vào phương trình còn lại được

$$G(j\omega) = -\frac{1}{N_1(E_1)}. \tag{5.18}$$

Nghiệm ω, E_1 sẽ là tọa độ của giao điểm hai đường đặc tính $G(j\omega)$, $-\dfrac{1}{N_1(E_1)}$ (*hình 5.11*) và đó chính là tần số và biên độ của dao động.

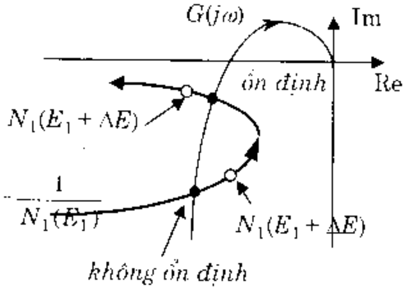
Tính ổn định của chế độ tự dao động được xác định nhờ vào cấu trúc (5.17) có dạng giống tiêu chuẩn Nyquist như sau: "Dao động sẽ không ổn định/ổn định nếu đường đặc tính tần số $G(j\omega)$ bao/không bao điểm $N_1(E_1 + \Delta E)$, trong đó E_1 là nghiệm của (5.18), tức là giao điểm của $G(j\omega)$ với $-\dfrac{1}{N_1(E_1)}$, và $\Delta E > 0$ là một sai lệch so với giao điểm đó, vì như vậy sai lệch ΔE sẽ có chiều hướng được giảm (hoặc tăng) theo thời gian".

Để minh họa phương pháp ta xét một ví dụ với hệ có cấu trúc như *hình 5.9*, trong đó đối tượng có hàm truyền

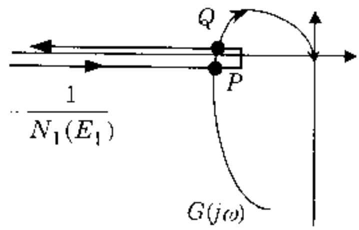
$$G(s) = 20\frac{1089}{s(s^2 + 20s + 1089)}$$

Bộ điều khiển là bộ mờ relay ba trạng thái với a=0,5 và b=1. Giả thiết đầu vào bộ điều khiển là $e(t) = E_1 \cos\omega t$. Do E_0=0 và đường đặc tính $f(e)$ là hàm lẻ nên N_0=0. Hệ số khuếch đại phức N_1 còn lại là [14]

$$N_1(E_1) = \begin{cases} 0 & \text{khi} \quad E_1 \le 0{,}5 \\ \dfrac{4}{\pi \; E_1}\sqrt{1 - \dfrac{1}{4E_1^2}} & \text{khi} \quad E_1 \ge 0{,}5 \end{cases}$$



Hình 5.11: Xác định chế độ tự dao động và tính ổn định của nó theo phương pháp cân bằng điều hòa.



Hình 5.12: Đồ thị xác định dao động ổn định theo phương pháp cân bằng điều hòa (ví dụ).

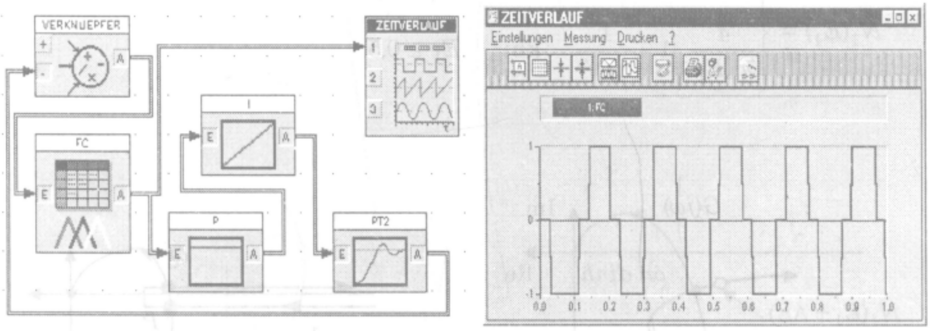
Điểm giao nhau giữa $G(j\omega)$ và $-\dfrac{1}{N_1(E_1)}$ phải thỏa mãn $\mathrm{Im}\big(G(j\omega)\big)=0$. Suy ra

$$1089\omega - \omega^3 = 0$$

$$\Rightarrow \qquad \omega = 33 \qquad \text{và} \qquad E_1 = \begin{cases} 0{,}55 \\ 1.14 \end{cases}.$$

Vậy hệ có hai dao động cùng tần số ω=33 nhưng với hai biên độ khác nhau. Để xét tính ổn định của các dao động đó có thể sử dụng đến phương pháp đồ thị như *hình 5.12* trình bày. Đường $-\dfrac{1}{N_1(E_1)}$ cắt đường đặc tính tần số $G(j\omega)$ tại hai điểm P và Q tương ứng với hai dao động trên, trong đó chỉ có dao động Q là ổn định (với E_1 = 1.14).

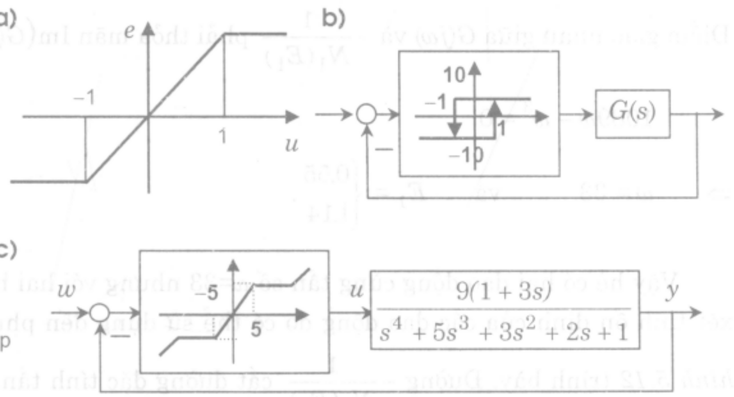
Hình 5.13 là kết quả mô phỏng ví dụ trên được xây dựng với phần mềm Winfact 3.0, trong đó bộ điều khiển mờ ba trạng thái được thiết kế giống như đã giới thiệu trong mục 2.1.1 (trường hợp 3).



Hình 5.13: Mô phỏng chế độ tự dao động của ví dụ đã cho với $E_1 = 1{,}14$.

Câu hỏi ôn tập và bài tập

1) Đối tượng có mô hình vào/ra $5\dfrac{dx}{dt} + x = 2{,}5u(t)$ được điều khiển bằng một bộ điều khiển mờ có quan hệ vào/ra tĩnh biểu diễn dưới dạng như ở hình 5.14a). Hãy kiểm tra tính ổn định của hệ theo tiêu chuẩn Lyapunov.



Hình 5.14: Cho các bài tập 1, 2 và 3.

2) Trong hệ mờ ở hình 5.14b) với $G(s) = \dfrac{25}{s^3 + 3s^2 + 2s + 1}$ có tồn tại chế độ tự giao động không? Khảo tính chất của các chế độ tự giao động trong hệ nếu có.

3) Hãy kiểm tính ổn định của hệ mờ có cấu trúc ở hình 5.14c) bằng tiêu chuẩn ổn định tuyệt đối của Popov.

6 PHẦN MỀM WINFACT

Phần mềm WinFACT, sản phẩm của văn phòng kỹ sư "Dr. J.Kahler" hiện có tại Bộ môn Điều khiển tự động Khoa Điện, ĐHBK Hà Nội, là một bộ các chương trình trợ giúp việc tổng hợp và phân tích các hệ điều khiển mờ [8]. Phần mềm bao gồm hai modul chính:

- modul Fuzzy-Shell, có tên là FLOP, sử dụng để thiết kế và phân tích một bộ điều khiển mờ.
- modul mô phỏng BORIS có cấu trúc khối, mỗi khối đặc trưng cho một khâu động học trong hệ thống điều khiển tự động. Bằng phần mềm BORIS có thể mô phỏng các hệ thống điều khiển tự động liên tục tuyến tính, hệ gián đoạn, hệ phi tuyến và hệ mờ.

Cả hai modul này cài đặt được trên các loại máy PC có bộ vi xử lý từ Intel 80386 trở lên và chạy với hệ điều hành Windows.

6.1 Cài đặt (Installation)

Toàn bộ nội dung của phần mềm WinFACT được để dưới dạng nén trong tệp **wfvieweg.exe**. Những dữ liệu này sẽ được bung ra một cách tự động. Để cài đặt WinFACT này cần thực hiện các bước sau đây:

1) Tạo trong ổ cứng một thư mục **Temporary** ví dụ

```
c:> md temp ↵
```

2) Sao chép tệp **wfvieweg.exe** từ đĩa gốc sang thư mục temp vừa khởi tạo, ví dụ

```
c:> copy a:wfvieweg c:\temp ↵
```

3) Chạy chương trình wfvieweg.exe để bung dữ liệu đã được nén, ví dụ

```
c:> cd c:\temp ↵

c:\temp> wfvieweg ↵
```

4) Chạy `install.exe` trong windows bằng lệnh Run và thực hiện các chỉ dẫn trên màn hình về nơi tìm đĩa gốc (ở đây sẽ là `c:\temp`) và nơi WinFACT được cài đặt. Địa chỉ mặc định nơi WinFACT sẽ được cài đặt là `c:\wf30`.

6.2 Tổng hợp bộ điều khiển mờ với FLOP

6.2.1 Giới thiệu chung

Các bộ điều khiển mờ có thể được tổng hợp và phân tích dễ dàng khi sử dụng chương trình FLOP (*Fuzzy Logic Operating Program*). Chương trình này cho phép thực hiện các công việc sau đây:

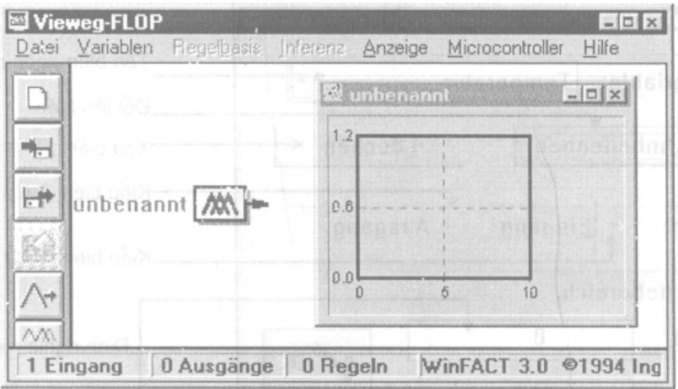
a) định nghĩa các biến ngôn ngữ và các giá trị mờ của biến,
b) chọn phép hợp và giao giữa các tập mờ,
c) soạn thảo luật hợp thành,
d) thực hiện quá trình hợp thành,
e) xác định đặc tính quan hệ truyền đạt của bộ điều khiển mờ,
f) mô phỏng theo các dữ liệu đã được chọn,
g) tạo ra bộ điều khiển mờ dưới dạng khối điều khiển mờ cho phần mềm mô phỏng hệ thống BORIS.

Có ba dạng hàm liên thuộc có thể được chọn cho một tập mờ là: dạng hình tam giác, hình thang và singleton. Có thể lựa chọn các nguyên tắc cài đặt luật hợp thành và phương pháp giải mờ khác nhau cho thiết bị hợp thành. Phần mềm có những hạn chế sau:

- số lượng biến ngôn ngữ cực đại cho mỗi bộ điều khiển mờ, kể cả các biến vào và biến ra, là 4.
- số lượng cực đại các tập mờ cho mỗi biến ngôn ngữ là 10.
- số lượng cực đại các luật điều khiển cho phép trong một luật hợp thành là 50.

6.2.2 Định nghĩa biến ngôn ngữ và các giá trị mờ

Trong cửa sổ chính của chương trình Flop, người sử dụng sẽ nhìn được tổng quan về các lệnh khai báo, sửa đổi biến ngôn ngữ, những giá trị mờ của biến ngôn ngữ và các luật điều khiển. Cửa sổ chính của Flop có dạng như sau

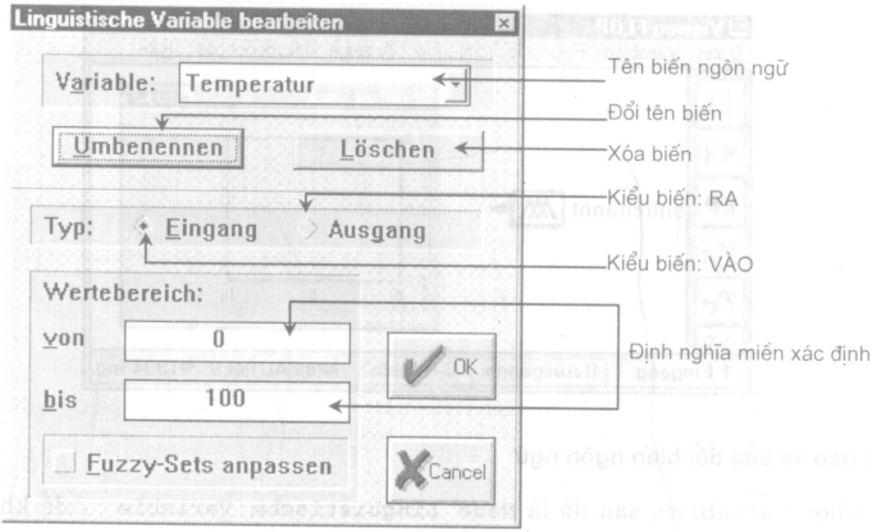


Khai báo và sửa đổi biến ngôn ngữ

Chọn `Variablen`, sau đó là **`Neue Linguistische Variable`** ... để khai báo biến mới hoặc **`Linguistische Variable bearbeiten`** để sửa đổi. Mỗi một biến ngôn ngữ được khai báo và biểu diễn trong một cửa sổ riêng. Người sử dụng có thể điều chỉnh kích thước của cửa sổ này theo ý mình. Biến ngôn ngữ đầu tiên có tên là **`unbenannt`** (tên mặc định). Để sửa lại tên biến ngôn ngữ thì phải vào **`Variablen`** và **`Linguistische Variable bearbeiten`**, tiếp theo chọn **`umbenennen`** để ghi tên mới, ví dụ là **`Temperatur`**. Một biến ngôn ngữ được xác định bởi:

- tên biến (nhiều nhất là 15 ký tự),
- kiểu biến (biến vào hoặc biến ra tương ứng với các mệnh đề điều kiện hay kết luận trong luật điều khiển),
- miền giá trị vật lý của biến.

Muốn xóa bất kỳ một biến ngôn ngữ nào (cùng với các giá trị của biến) chỉ cần chọn biến cần xóa, sau đó ấn phím **`Löschen`**, loại trừ trường hợp nó là biến ngôn ngữ duy nhất hay cuối cùng. Sau khi đã đổi tên biến thành **`Temperatur`**, bắt đầu xác định miền giá trị vật lý cho biến. Ví dụ nếu đã chọn **`Temperatur`** cho tên biến, **`Eingang`** (Input–biến vào) cho kiểu biến và $0^0C \div 100^0C$ cho miền giá trị vật lý của biến thì cửa sổ màn hình sẽ như sau



Khi thay đổi miền giá trị vật lý của biến ngôn ngữ đã có các giá trị mờ cũng cần phải thay đổi miền giá trị vật lý của tập mờ cho phù hợp. Điều này có thể thực hiện được khi ấn phím `Fuzzy-Set anpassen`. Nếu không ấn phím này thì tập mờ vẫn giữ nguyên giá trị như đã được định nghĩa trước đó. Kết thúc việc khai báo biến bằng cách ấn `OK`.

Định nghĩa và xác định các giá trị mờ cho biến ngôn ngữ

Các giá trị của biến ngôn ngữ vừa mới khai báo tất nhiên chưa được định nghĩa. Muốn khai báo các giá trị mờ cho biến ngôn ngữ ví dụ cho biến `Temperatur` thì phải thực hiện các bước

1) khai báo tên giá trị mờ,

2) định nghĩa kiểu giá trị mờ.

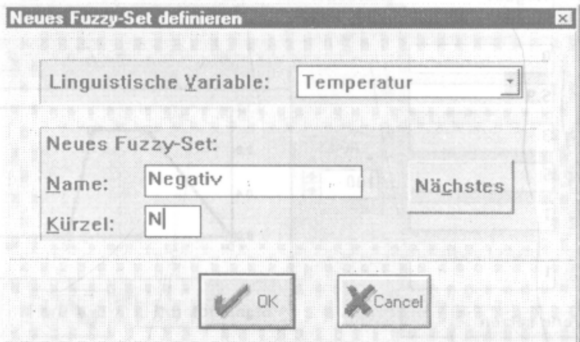
Muốn khai báo tên các giá trị mờ cho một biến ngôn ngữ thì vào `Variablen`, sau đó là `Neues Fuzzy-Sets`. Ở chế độ này sẽ định nghĩa được các tập mờ cho biến ngôn ngữ bao gồm:

a) tên biến ngôn ngữ sẽ được gán giá trị, ví dụ `Temperatur`,

b) khai báo tên các tập mờ (nhiều nhất bằng 12 ký tự, ví dụ `Negativ`).

c) tên rút gọn của tập mờ (nhiều nhất 2 ký tự, ví dụ `N`).

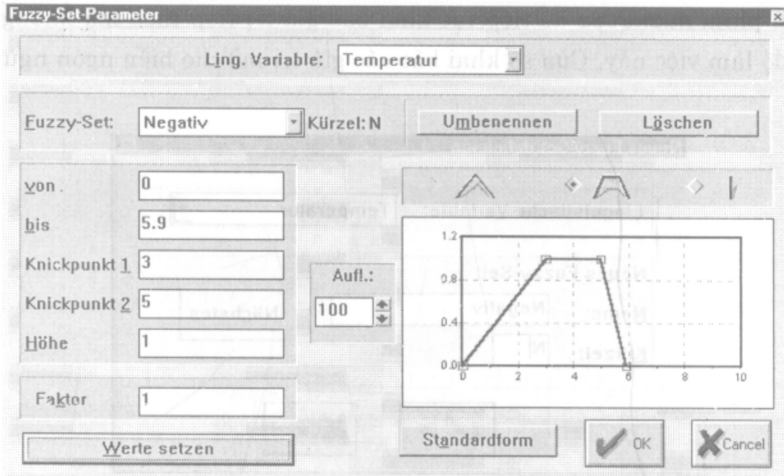
Ký tự rút gọn sẽ được dùng để biểu diễn các nguyên tắc điều khiển trong bảng biểu diễn luật điều khiển. Nếu cần soạn thảo nhiều tập mờ cho một biến ngôn ngữ chỉ cần ấn phím **Nächstes** để tiếp tục khai báo giá trị tiếp theo mà không phải ra khỏi chế độ làm việc này. Cửa sổ khai báo các giá trị mờ cho biến ngôn ngữ có dạng sau



Sau khi khai báo xong tên các giá trị mờ (tập mờ) cho biến ngôn ngữ thì kết thúc với phím **OK** và chuyển sang bước hai là định nghĩa kiểu tập mờ bằng cách chọn trong menu **Variable** mục **Fuzzy-Set bearbeiten** hoặc ấn phím trái của chuột hai lần lên cửa sổ có biến ngôn ngữ cần định nghĩa các giá trị. Định nghĩa một tập mờ bao gồm các công việc

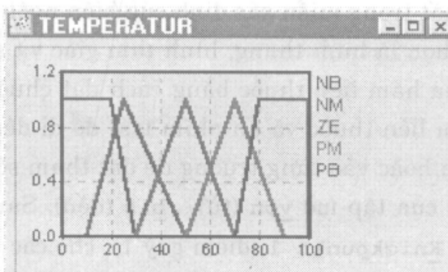
- Chọn tên biến ngôn ngữ.
- Chọn tên tập mờ trong biến.
- Chọn kiểu hàm liên thuộc cho tập mờ đó. Hàm liên thuộc mặc định là một tam giác cân có hai đáy tương ứng với điểm đầu và điểm cuối của tập mờ là giá trị nhỏ nhất và lớn nhất trong miền xác định của biến ngôn ngữ. Có ba dạng hàm liên thuộc có thể chọn là hình thang, hình tam giác và singleton. Có thể thay đổi các tham số của hàm liên thuộc bằng cách đặt chuột lên điểm được đóng khung đỏ trên hàm liên thuộc và ấn phím trái để di đến vị trí thiết kế mong muốn rồi nhả phím hoặc vào từng trường để đặt tham số, đầu tiên là xác định miền giá trị vật lý của tập mờ **von** (từ)... **bis** (đến). Sau đó đến xác định các thông số khác như **Knickpunkt 1** (điểm gãy 1) cho các tập mờ kiểu hình tam giác và hình thang, **Knickpunkt 2** (điểm gãy 2) cho các tập mờ hình thang.
- Ấn phím **Werte setzen** để nhập các giá trị vừa khai báo rồi quay lại bước a) hoặc b) để định nghĩa tiếp hàm liên thuộc tập mờ sau hoặc ấn **OK** để thoát khỏi chế độ khai báo tập mờ.

Cửa sổ khai báo tập mờ và hàm liên thuộc của nó có dạng sau:



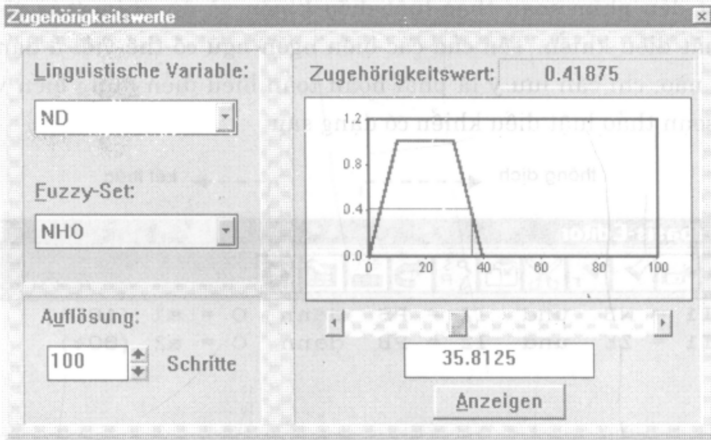
Nên quan tâm đến việc sử dụng phím `Standardform`. Phím này giúp cho việc chọn các tập mờ tiêu chuẩn nhanh nhất và có thể dùng làm cơ sở cho việc thay đổi các thông số của tập mờ. Dạng mặc định hình tam giác sẽ cho hàm liên thuộc phủ đầy miền giá trị xác định của biến ngôn ngữ.

Sau khi đã khai báo đầy đủ các tập mờ cho biến ngôn ngữ, ví dụ các giá trị *NB*, *NM*, *ZE*, *PM* và *PB* cho biến `Temperatur` và ra khỏi chế độ khai báo bằng phím `OK`, màn hình biểu diễn biến đó sẽ như sau, trong đó hàng thông báo trạng thái ở của sổ chính cho biết số lượng và kiểu tập mờ của biến ngôn ngữ cũng như số lượng luật điều khiển đã được soạn thảo.



Tính toán độ phụ thuộc của một giá trị rõ

Để xác định độ phụ thuộc của các giá trị rõ tương ứng với các tập mờ, chọn trong menu `Variablen` mục `Zugehörigkeitswerte`. Giá trị độ phụ thuộc được xác định cho từng giá trị rõ như sau



Để xác định được giá trị rõ của biến nhiệt độ có thể tiến hành theo hai cách:

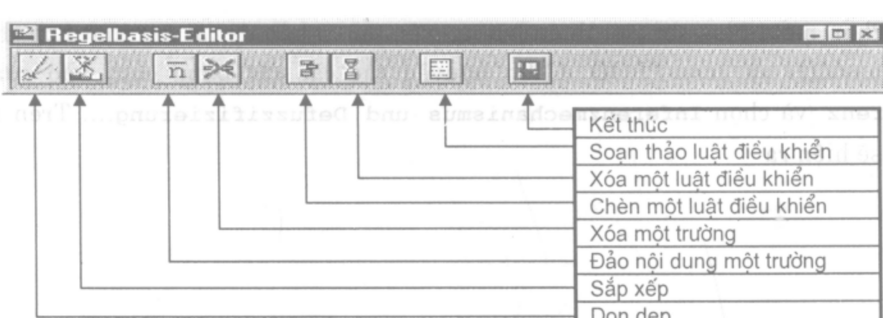
- Trực tiếp đưa giá trị rõ vào trường ở phía trên phím `Anzeigen`, sau đó nhấn phím `Anzeigen`.
- Di chuột trên tập mờ đến giá trị cần xác định, nhả phím trái.

Độ phân dải bằng 100 có nghĩa là 100 bước cho tất cả miền giá trị của biến ngôn ngữ, có thể tăng hay giảm tùy ý số bước (độ phân dải). Độ phụ thuộc có thể đọc được ở phía trên của cửa sổ tập mờ.

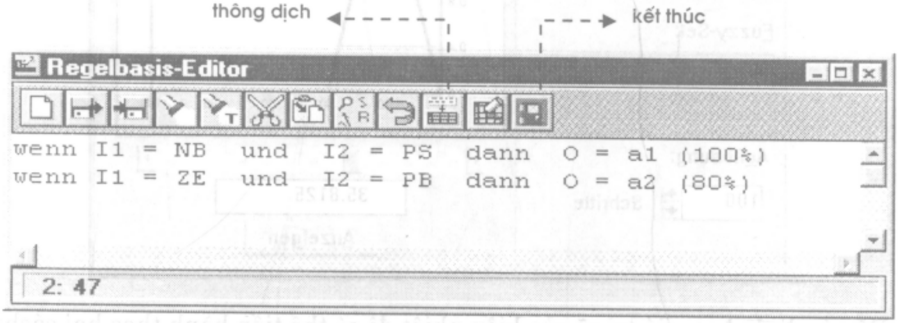
6.2.3 Xây dựng thiết bị hợp thành

Khai báo luật hợp thành

Luật hợp thành được định nghĩa và xây dựng trong menu **`Regelbasis`** mục **`Neu`**. Cửa sổ này cũng như cửa sổ khai báo biến ngôn ngữ **`Variable`** có thể khuếch đại hoặc thu nhỏ tùy ý. Từng icon trong thanh ghi điều khiển của cửa sổ khai báo luật hợp thành có ý nghĩa như sau:



Di chuột đến phím soạn thảo luật điều khiển và ấn phím trái để vào chế độ soạn thảo luật điều khiển. Tên của các biến ngôn ngữ có thể viết ở dạng rút gọn ở bất cứ điểm nào, chỉ cần lưu ý là phải hoàn toàn biểu diễn đúng biến và đúng tập mờ. Cửa sổ soạn thảo luật điều khiển có dạng sau



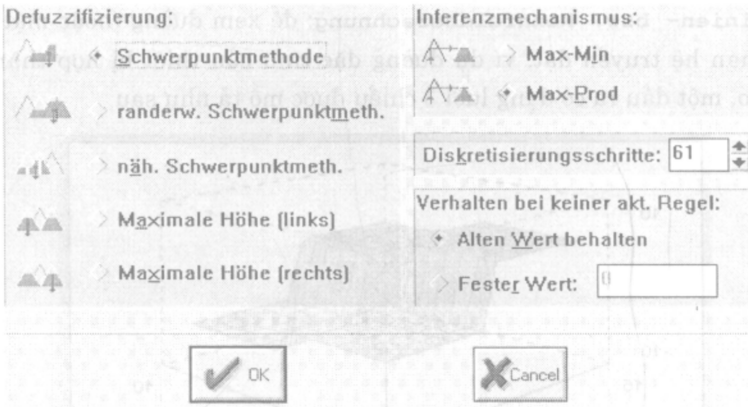
Mỗi luật điều khiển đều phải có đầy đủ tất cả các biến ngôn ngữ đầu vào trong mệnh đề điều kiện và tất cả các biến ngôn ngữ đầu ra trong mệnh đề kết luận, chẳng hạn cho bộ điều khiển mờ có hai biến đầu vào là I1, I2 với 5 giá trị *NB*, *NM*, *ZE*, *PM*, *PB* và một biến đầu ra O với 3 giá trị a1, a2 và a3 thì cấu trúc chung các luật điều khiển phải có dạng sau

```
wenn I1 = NB und I2 = PS dann O = a1 (100%)
...
```

trong đó "wenn" có nghĩa là NẾU, "und" có nghĩa là VÀ và "dann" có nghĩa là THÌ. Số % trong ngoặc đơn chỉ hệ số an toàn của luật đó, tức là độ tin cậy mà luật sẽ được tính để tham gia trong thiết bị hợp thành. Trước khi kết thúc việc soạn thảo các luật điều khiển phải dịch chúng bằng cách tác động vào phím thông dịch. Các lỗi cú pháp sẽ được thông báo ngay lên màn hình.

Nguyên tắc cài đặt luật hợp thành và giải mờ

Phần mềm Flop chỉ giới thiệu hai nguyên tắc cơ bản để cài đặt luật hợp thành là max-MIN và max-PROD. Chọn nguyên tắc cài đặt bằng cách vào cửa sổ `Inferenz` và chọn `Inferenzmechanismus und Defuzzifizierung`.... Trên màn hình sẽ hiện ra

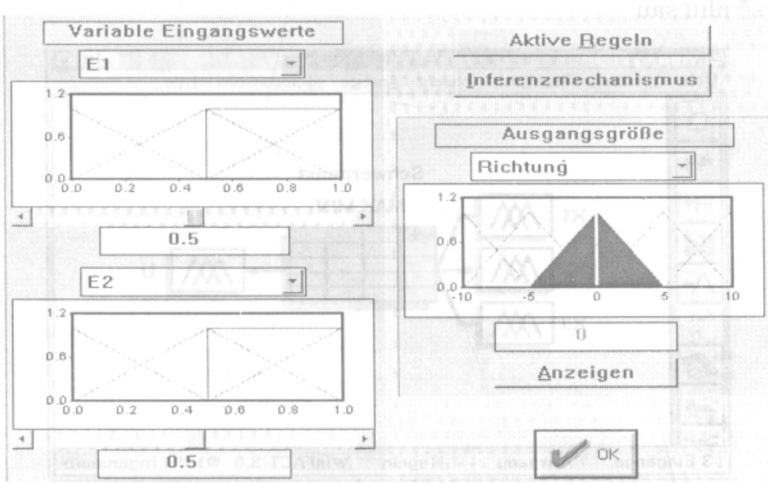


Chọn nguyên tắc cài đặt nào thì đánh dấu vào nguyên tắc đó, cũng như vậy cho phương pháp giải mờ. Flop cung cấp 5 phương pháp giải mờ khác nhau. Kết thúc việc chọn nguyên lý cài đặt luật hợp thành bằng cách ấn phím `OK`.

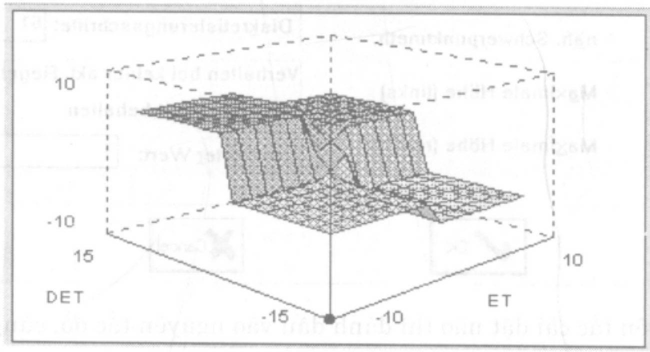
Ngoài ra, trong cửa sổ `Inferenz` còn có

1) `Simulation`: để mô phỏng với các số liệu vào bên ngoài. Ở đây, các giá trị vào được đọc từ FSI và đầu ra được biểu diễn dưới dạng đồ thị. Để các giá trị vào đều có độ phân dải như nhau, các giá trị này được chuẩn hóa theo giá trị cực đại, để các giá trị được chỉ thị đều nằm trong khoảng -1 đến +1.

2) `Einzelschritt- Inferenz`: để kiểm nghiệm thiết bị hợp thành, ví dụ có thể xem với giá trị rõ 0.5 cho cả hai biến E1 và E2 tại đầu vào thì những luật điều khiển nào được tích cực, dạng tập mờ đầu ra như thế nào và giá trị rõ đầu ra bằng bao nhiêu.... Cửa sổ của chức năng này có dạng như sau



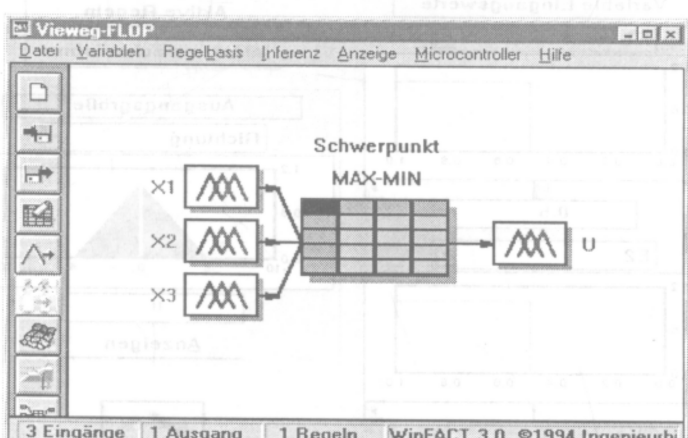
3) `Kennlinien- bzw. Kennfeldberechnung`: để xem đường (hoặc mặt cong) đặc tính quan hệ truyền đạt, ví dụ đường đặc tính của thiết bị hợp thành với hai đầu vào, một đầu ra có dạng lưới 3 chiều được mô tả như sau



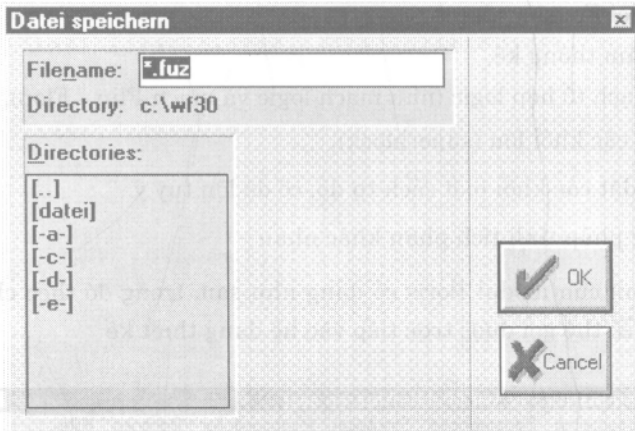
6.2.4 Hoàn thiện một bộ điều khiển mờ

Sau khi đã hoàn tất các thủ tục như định nghĩa biến ngôn ngữ vào ra, định nghĩa các giá trị mờ cho từng biến, chọn nguyên tắc cài đặt thiết bị hợp thành và phương pháp giải mờ, về nguyên tắc công việc xây dựng bộ điều khiển mờ bằng modul Flop đã hoàn tất.

Quay về cửa sổ chính của Flop, lúc nay trên cửa sổ chính của Flop là cấu trúc của một bộ điều khiển mờ với tất cả các biến ngôn ngữ vào ra đã được định nghĩa. Ví dụ sau khi khai báo 3 biến ngôn ngữ đầu vào, một biến ngôn ngữ đầu ra cùng tất cả các giá trị mờ của chúng cho một bộ điều khiển mờ được cài đặt theo nguyên lý max–MIN với phương pháp giải mờ theo nguyên lý điểm trọng tâm, cửa sổ màn hình Flop sẽ như sau



Công việc cuối cùng cần làm là cất giữ bộ điều khiển mờ vừa được xây dựng lên đĩa. Để làm việc đó thì phải vào cửa sổ `Datein`, chọn `Speichern unter` (giống như `Save As` của WinWord). Trên màn hình sẽ xuất hiện yêu cầu nơi cất giữ gồm ổ đĩa, thư mục, tên tệp trong cửa sổ đối thoại có dạng sau



Ghi các thông tin cần thiết đó và ấn `OK`. Trong trường hợp chỉ sửa đổi một bộ điều khiển mờ đã có và ghi đè lên tệp cũ thì chọn `Speichern` thay vì `Speichern unter`.

6.3 Mô phỏng và tối ưu hệ thống điều khiển mờ bằng BORIS

6.3.1 Vài nét về modul BORIS

Modul Boris mô phỏng các hệ động học có cấu trúc bất kỳ, có thể có hoặc không có bộ điều khiển mờ trong thành phần của hệ. Phần mềm cho phép mô phỏng các hệ thống điều khiển mờ từ đơn giản đến các hệ mờ lai, mờ thích nghi. Số khối nhiều nhất cho phép trong BORIS là 12. Phần có các những tiềm năng sau đây:

1) Thư viện hệ thống phong phú:
   a) Máy phát tín hiệu (xung tam giác, xung chữ nhật, hình sin, nhiễu, các hàm kiểm tra).
   b) Các khâu tuyến tính tiêu chuẩn (P, I, D, PT_1, PT_n ...).
   c) Các khâu truyền đạt bậc cao.
   d) Các khâu phi tuyến điển hình với đặc tính bất kỳ, các hàm đại số.
   e) Các bộ điều khiển tuyến tính và phi tuyến tiêu chuẩn, bộ điều khiển thích nghi.

f) Bộ điều khiển mờ có chế độ tìm lỗi mờ.

g) Thiết bị hiển thị (hiện thị theo thời gian, hiển thị quĩ đạo pha, thiết bị hiển thị số và tương tự, hiện thị trạng thái ...).

h) Vào ra số liệu từ tệp.

i) Biến đổi Fourier nhanh.

j) Các hàm thống kê.

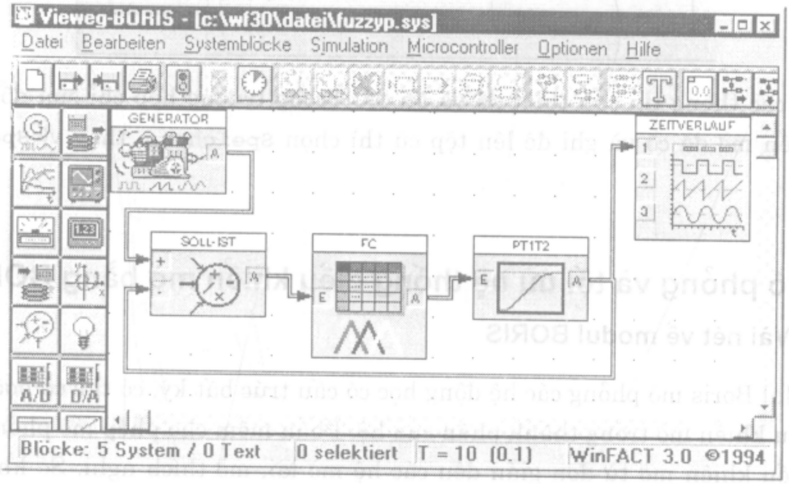
k) Các mạch tổ hợp logic (như mạch logic và mạch Flip - Flop).

2) Định nghĩa các khối lớn (superblock).

3) Có thể xếp đặt các khối một cách tự do, có độ lớn tuỳ ý.

4) Các phương pháp tính tích phân khác nhau

Cửa sổ chính của modul Boris có dạng như sau, trong đó theo chiều dọc là các khối chức năng có thể gọi được trực tiếp vào hệ đang thiết kế



6.3.2 Thành phần cửa sổ chính trong modul BORIS

Cửa sổ chính của BORIS bao gồm các thành phần sau: Tên của cửa sổ chính là tên tệp của hệ thống đang thiết kế. Sau đó đến các chức năng từ trái sang phải: `Datei` (tệp), `Bearbeiten` (sửa đổi), `Systemblöcke` (các khối cho các khâu động học), `Simulation` (mô phỏng), `Microcontrroller` (bộ điều khiển micro), `Optionen` (các chế độ), `Hilfe` (trợ giúp). Sau đó đến các lệnh điều hành được xếp hàng ngang phía trên của cửa sổ và hàng dọc phía bên trái của cửa sổ.

Cửa sổ bao gồm ngoài các thành phần cơ bản của Windows là các thành phần sau đây:

- Phía dưới menu là các thành phần của Toolbar với các phím cho các chức năng thường được sử dụng nhiều trong thiết kế.
- Toolbar dọc theo mép cửa sổ bên trái là các khối thường hay được sử dụng trong thiết kế. Có thể tự sắp xếp các Toolbar này theo ý người sử dụng, bằng cách vào **Optionen/Systemblocktoolbar konfigurieren**.
- Hàng biểu diễn trạng thái ở phía dưới cùng của cửa sổ thông báo các khối được sử dụng, tên khối , các thông số mô phỏng tức thời và thông số mô phỏng được tính theo $T - T_{Simu}(\Delta T)$, với T_{Simu} là thời gian mô phỏng và ΔT là bước mô phỏng. Trong suốt quá trình mô phỏng, thời gian mô phỏng tức thời sẽ được hiển thị ở trong hàng này cho đến khi chế độ hiển này bị bãi bỏ.
- Cửa sổ thiết kế để biểu diễn cấu trúc của hệ thống cần thiết kế. Có thể thiết kế cả theo phương nằm ngang lẫn phương thẳng đứng. Trạng thái ban đầu có thể đặt lại bất cứ lúc nào bằng **Optionen / Bildausschnitt in Ursprung**.

Cửa sổ thiết kế được phân chia theo các điểm cách đều nhau để có thể sắp đặt các khối một cách tự động. Có thể xoá các điểm này khi vào **Optionen / An Raster ausrichten**, sau đó thì có thể sắp đặt các khối tùy theo người sử dụng. Hiện thị của phân chia điểm cũng có thể xóa bằng cách vào **Optionen / Raster Anzeiger**.

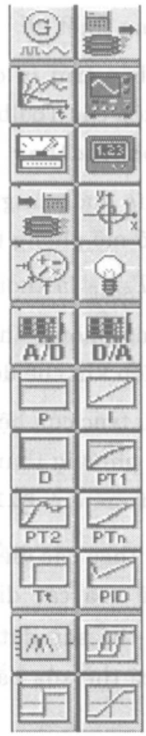
6.3.3 Gọi và lập trình cho các khối của hệ thống

Gọi và lập trình cho các khối của hệ thống bao gồm các chức năng sau đây:

- Gọi một khối mới từ thư viện các khối hệ thống.
- Chọn từng khối hoặc nhóm các khối.
- Dịch chuyển một khối hoặc nhóm các khối.
- Xóa một khối hay một nhóm khối.
- Quay từng khối.
- Sao chép hoặc gọi một khối hay một nhóm khối.
- Chọn thông số cho từng khối.
- Các đại lượng khác của khối.

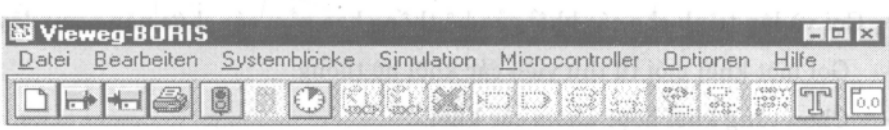
Gọi một khối

Thanh Toolbar dọc của màn hình Boris bao gồm các khối chức năng sau



Bộ phát tín hiệu	Đọc tệp
Hiển thị đặc tính thời gian	Oszillograph
Hiển thị tương tự	Hiển thị số
Cất tệp	Hiển thị quỹ đạo pha
Khối cộng/trừ/nhân/chia	Hiển thị trạng thái
Biến đổi tương tự/số	Biến đổi số/tương tự
Khâu khuếch đại	Khâu tích phân
Khâu vi phân	Khâu quán tính bậc 1
Khâu dao động	Khâu quán tính bậc n
Khâu trễ	Bộ điều khiển PID
Bộ điều khiển mờ	Khâu hai vị trí có trễ
Relay hai vị trí	Khâu khuếch đại bão hòa

Thanh Toolbar ngang bên trên của Boris bao gồm

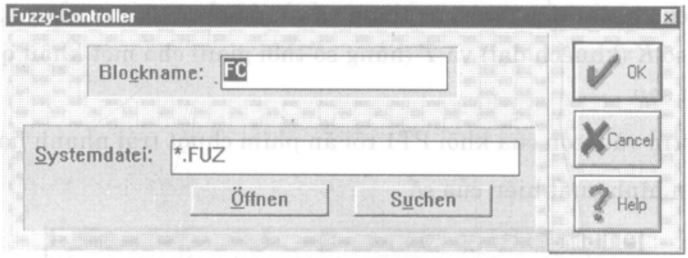


Các khối chức năng được gọi vào chương trình mô phỏng như sau:

- Di chuột vào khối cần sử dụng, ấn phím trái, hoặc
- Chọn khối cần sử dụng trong **Systemblöcke**. Ví dụ muốn gọi một khối mờ được thiết kế trước đó bằng modul Flop và đã được lưu cất vào `c:\fuzzy` dưới tên `temp.fuz` thì thực hiện các công việc
  - \+ vào của sổ **Systemblöcke**
  - \+ chọn **Statische Blöcke**

+ chọn **Fuzzy-Controller** rồi ấn ↵ hoặc phím chuột trái. Trên màn hình Boris xuất hiện icon

+ đưa con trỏ vào icon này rồi bấm phím chuột trái nhanh hai cái, trên màn hình xuất hiện cửa sổ



khai báo tên tệp của khối mờ, ví dụ **c:\fuzzy\temp.fuz** hoặc chọn **Suchen** để đi tìm tên tệp rồi ấn **OK**.

Các khối được gọi ra sẽ được sắp đặt tự động ở phía trên màn hình thiết kế và theo thứ tự từ trái qua phải, từ trên xuống dưới trong khi cửa sổ vẫn còn chỗ.

Trước khi bắt đầu các phép toán với một khối như khai báo tham số cho khối, nối khối... cần phải chọn khối đó bằng cách đưa con trỏ vào icon của khối rồi ấn phím chuột trái. Có thể chọn một nhóm các khối bằng cách tương tự nhưng thay vì chỉ ấn phím chuột trái thì ấn **Shift** hoặc phím **Ctrl** cùng với phím chuột trái. Cũng có thể nhóm các khối lại với nhau bằng cách giữ phím chuột trái và di con trỏ thành hình chữ nhật bao các khối cần chọn rồi nhả phím. Có thể chọn cùng một lúc tất cả các khối bằng cách vào **Bearbeiten** chọn **Alle Blöcke selektieren** hoặc bằng cách nhấn đồng thời hai phím **Ctrl** và **A**.

Dịch chuyển khối hoặc nhóm khối

Để dịch chuyển các khối hay một nhóm các khối phải thực hiện các bước như sau:

- Xác định khối hoặc nhóm khối cần dịch chuyển.
- Nhấn phím chuột trái trong lòng khối hoặc nhóm khối cần dịch chuyển và giữ phím trong suốt thời gian dịch chuột đến vị trí mới. Trong khi thực hiện chức năng này con trỏ sẽ được chuyển sang hình chữ thập. Sau khi chuyển dịch đến

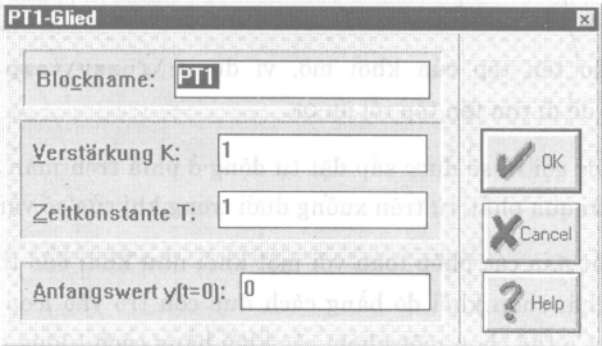
vị trí mới, các khối vẫn phải được xếp đặt theo nguyên tắc không được xếp chồng lên nhau.

Đặt tham số cho khối

Chọn tham số cho một khối được thực hiện đơn giản nhất là chuyển con trỏ của chuột đến vị trí đó và ấn phím chuột trái hai lần. Cách thường làm là vào menu sau khi đã chọn block `Bearbeiten / Block bearbeiten`, sau đó có thể trực tiếp chọn các thông số cho khối. Mặc định các tham số của khối có giá trị bằng 1. Ví dụ muốn đặt lại tham số *K* (khuếch đại) và *T* (hằng số thời gian) cho một khâu quán tính bậc 1 thì làm như sau

– đưa con trỏ vào icon của khối PT1 rồi ấn phím chuột trái nhanh hai lần,

– trên màn hình xuất hiện cửa sổ



– cho tham số mới rồi ấn `OK`.

Thay đổi kích thước của khối

Bình thường các khối được biểu diễn theo kích thước tiêu chuẩn, nhưng trong những trường hợp cần thiết vẫn có thể thu nhỏ lại theo 3 mức 80%, 60% và 40%.

Để thay đổi kích thước của các khối cần thực hiện các bước sau đây:

– chọn khối,

– để làm nhỏ một lần so với kích thước đang có thì vào cửa sổ `Bearbeiten` chọn `Block verkleinern`. Muốn tăng kích thước của khối được chọn lên một mức thì vào cửa sổ `Bearbeiten` chọn `Block vergrößern`,

– để có kích thước vừa ý cho một khối, vào cửa sổ `Bearbeiten` rồi chọn `Block/Blockgröße` 40% hoặc chọn giá trị khác như 60%, 80%....

6.3.4 Nối các khối với nhau

Việc nối các khối với nhau được thực hiện với sự trợ giúp của chuột. Bởi vì các đường nối được tích hợp một cách tự động dạng gấp góc chữ nhật hoặc tạo ra các đường không giao nhau, nên chỉ cần phải chọn hai khối cần nối với nhau, theo nguyên tắc: "*Đường nối được bắt đầu từ đầu ra của khối này đến đầu vào của khối kia*". Để nối hai khối với nhau cần thực hiện:

- Trước tiên là bắt đầu từ đầu ra của khối chủ định nối bằng cách nhấn chuột vào cổng của khối có ký hiệu là A. Con trỏ của chuột chuyển sang dạng của một chiếc "mỏ hàn" và bây giờ bắt đầu thực hiện nối hai khối.
- Chuyển "mỏ hàn" tới đầu vào của khối cần được nối với khối kia và nhấn phím chuột trái vào cổng có ký hiệu E của khối đó hoặc đầu vào cộng hay trừ của khối cộng. Khi di chuột sẽ xuất hiện một đường nối chạy theo sau "mỏ hàn". nếu muốn thay đổi đường nối, để vị trí của "mỏ hàn" ở bất kỳ chỗ trống nào trong màn hình và nhấn phím chuột trái.

Khi kết thúc quá trình nối đúng như qui định trên, đường nối sẽ có mũi tên tự động theo đúng chiều. Đường nối được thực hiện theo nguyên tắc là không cắt qua các khối khác trong hệ thống. Nếu chẳng may có những đường giao nhau xảy ra, thì chỉ cần di chuyển khối một cách nhẹ nhàng sẽ có được đường nối mong muốn. Những đường nối cùng xuất từ đầu ra của một khối sẽ được nối lại với nhau một cách tự động. Từ một đầu ra của một khối có thể có những đường nối với các khối bất kỳ nào. Còn mỗi một đầu vào thì chỉ được nối với một đầu ra của một trong các khối khác trong hệ thống. Cũng có thể thực hiện các công việc nối trên đây trong cửa sổ **`Optinen / Anzeige`**.

Xoá một đường nối như thế nào?

Để xoá một đường nối giữa hai khối đã có, chọn một trong hai khối đó, hoặc khối đi hoặc khối đến. (khối có đường nối bắt đầu từ cổng ra A của khối đó gọi là *khối đi*, khối có đường nối vào cổng vào E gọi là *khối đến)*, sau đó có thể xoá đường nối giữa hai khối bằng hai cách:

- Xoá đường nối đầu ra A của khối đã được chọn bằng cách vào cửa sổ **`Bearbeiten`** rồi chọn **`Ausgangsverbindungen löschen`** hoặc
- Xoá đường nối đầu vào E của khối đã được chọn bằng cách vào cửa sổ **`Bearbeiten`** rồi chọn **`Eingangsverbindung löschen`**.

Trong cả hai trường hợp các đường nối vào ra đều bị xóa sạch.

6.3.5 Khối văn bản và đóng khung hàm

Làm việc với khối văn bản

Ngoài khả năng cung cấp các khối sử dụng vào việc mô phỏng hệ thống, modul BORIS còn cho phép viết các câu bình cho sơ đồ cấu trúc được thiết kế. Văn bản có thể được thể hiện bằng mầu sắc và kích thước tự chọn. Các đoạn văn bản soạn thảo có thể di chuyển vị trí được cũng như có thể xoá được. Muốn soạn thảo đầu tiên phải chọn cửa sổ `Bearbeiten` rồi vào `Text einfügen` hoặc là nhấn phím tương đương trong `Toolbar`. Sau đó hiện ra trong cửa sổ chữ "`Text`" có mầu sắc và độ lớn được chọn tự động.

Chức năng đóng khung của BORIS

Để nâng cao chất lượng tư liệu của hệ thống, phần mềm BORIS cung cấp hàm chức năng đóng khung. Hàm đóng khung có chức năng nhóm tất cả các khối trong cùng một hệ thống vào một khung quang học. Hàm này không làm ảnh hưởng đến quá trình mô phỏng hệ thống. Đóng khung các khối được thực hiện như sau:

- Chọn các khối có dự định đóng khung với nhau.
- Vào cửa sổ `Bearbeiten` và chọn `Rahmen einfügen` hoặc nhấn phím tương ứng với chức năng này trong Toolbar.

Chú ý: khung này là một khung tĩnh, có thể dịch chuyển và xóa các khung này cùng với việc xóa các khối.

Khung có một khoảng cách nhất định đến các khối, khoảng 8 pixel hình chữ nhật, có đường viền mầu đen dày 1 pixel. Những thông số này có thể thay đổi được theo ý muốn của người sử dụng, bằng cách:

- Chọn một khối bất kỳ trong khung.
- Vào cửa sổ Bearbeiten rồi chọn Rahmen bearbeiten.

Để xóa một khung cần thực hiện các bước dưới đây:

- Chọn một khối bất kỳ trong khung.
- Vào `Bearbeiten` sau đó chọn `Rahmen löschen`.

6.3.6 Chỉnh định các thông số cho quá trình mô phỏng

Các thông số cơ bản của một quá trình mô phỏng bao gồm: Thời gian mô phỏng T_{Simu}, bước tính ΔT, phương pháp mô phỏng (phương pháp tính tích phân), kiểm tra phạm vi giá trị. Trước khi tiến hành mô phỏng hệ thống phải chọn tất cả các thông số đó. Đó là:

1) *Thời gian mô phỏng* T_{Simu}: Là khoảng thời gian tiến hành quá trình mô phỏng, được tính bằng

   $T_{Simu} = T_e - T_b$

   với T_b là thời gian bắt đầu quá trình mô phỏng và T_e là thời gian kết thúc quá trình mô phỏng. Thời gian mặc định là 10s.

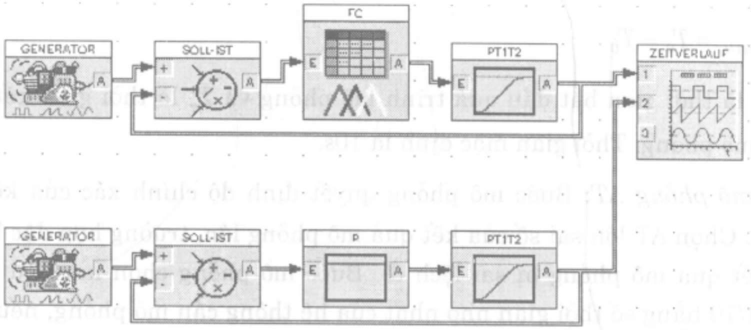
2) *Bước mô phỏng* ΔT: Bước mô phỏng quyết định độ chính xác của kết quả mô phỏng. Chọn ΔT lớn sai số của kết quả mô phỏng lớn, trường hợp đặc biệt có thể làm kết quả mô phỏng bị sai lệch đi. Bước mô phỏng chọn lần đầu có thể lấy bằng 1/10 hằng số thời gian nhỏ nhất của hệ thống cần mô phỏng, nếu chưa hợp lý có thể chọn lại. Giá trị mặc định là 0,1s.

3) *Phương pháp mô phỏng*: Đó chính là phương pháp tính phân số để mô phỏng các khâu động. Modul Boris cho phép chọn một trong ba phương pháp tích phân số sau đây: phương pháp *Euler*, phương pháp *Runge - Kutta* bậc 4 và phương pháp ma trận của các hàm mũ. Trong ba phương pháp, phương pháp *Euler* là phương pháp tích phân số đơn giản nhất đòi hỏi thời gian tính nhỏ nhất vì mỗi bước chỉ cần tính 1 giá trị. Phương pháp này có sai số bậc hai $S(\Delta T^2)$ và thường sử dụng để mô phỏng các hệ động học đơn giản với bước tính tương ứng nhỏ. Đối với các hệ thống có dao động cũng như các hệ thống có các hằng số thời gian rất khác nhau không nên chọn phương pháp tích phân *Euler*, mà nên chọn phương pháp *Runge-Kutta*. Phương pháp náy có sai số rất bé khoảng bậc năm $S(\Delta T^5)$. Vì phương pháp này ở mỗi bước phải tính 4 giá trị hàm nên thời gian tính lâu hơn so với phương pháp *Euler*. Trong trường hợp tín hiệu chủ đạo có dạng hàm bậc thang, phương pháp này có thể dẫn đến kết quả mô phỏng sai do vậy nên chọ phương pháp *Euler* với bước tính nhỏ. Đối với các hệ thống tuyến tính nên chọn phương pháp tích phân thứ ba, phương pháp này cho độ chính xác cao, bậc của sai số có thể rất lớn. Thường được dùng cho các hệ thống có bậc lớn hơn 3.

6.3.7 Mô phỏng

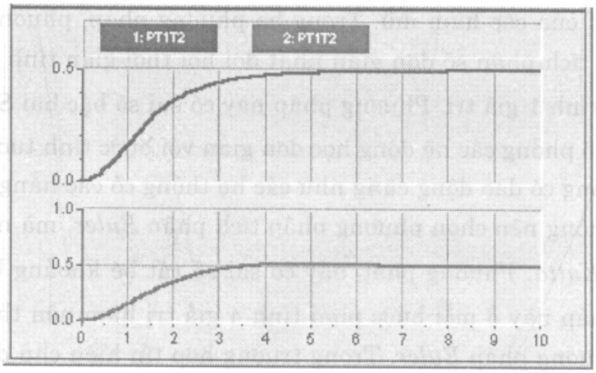
Sau khi đã xây dựng xong mô hình mô phỏng thì vào cửa sổ **Simulation**, chọn **Parameter** để đặt các tham số cho quá trình mô phỏng như thời gian mô phỏng, tần số lượng tử. Sau khi đặt xong tham số thì lại vào sổ **Simulation** rồi chọn **Start** để bắt đầu quá trình mô phỏng. Có thể kết thúc quá trình mô phỏng vào bất cứ thời điểm nào bằng cách chọn `Stop` trong cửa sổ `Simulation`.

Ví dụ sau khi đã xây dựng xong hệ thống



Fuzzy-P-Controller

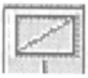
và vào cửa sổ `Simulation` rồi chọn `Start` thì sẽ có được kết quả mô phỏng sau



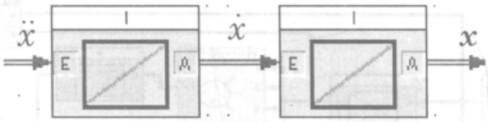
Để minh họa cụ thể cho việc sử dụng phần mềm BORIS, ta xét thêm một ví dụ về mô phỏng hệ phi tuyến không có kích thích được mô tả bởi phương trình Duffing

$$\ddot{x} = k_1 x - k_2 x^3 - k_3 \dot{x}$$

Hệ trên là một hệ bậc hai, bởi vậy ta cần hai khâu tích phân để có x từ $\ddot{x}$.

Đưa chuột vào ô  tại thanh toolbar dọc mép bên trái màn hình rồi ấn chuột trái hai lần để gọi khối.

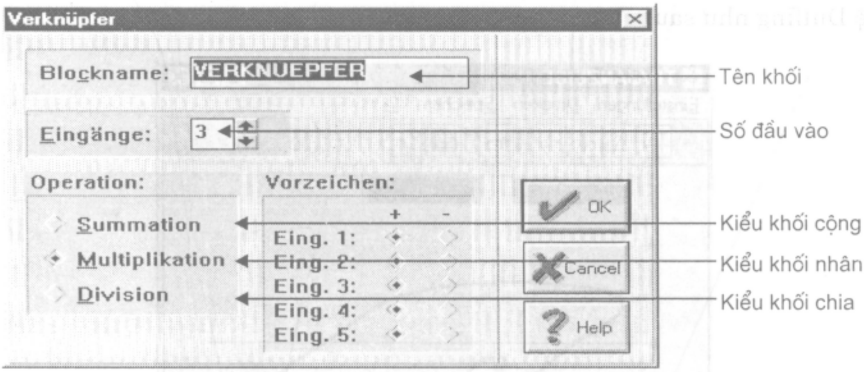
Nối đầu ra của khối thứ nhất với đầu vào của khối thứ hai thì $\ddot{x}$ chính là đầu vào của khối thứ nhất và x là đầu ra của khối thứ hai.



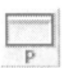
Tiếp tục, để tạo được phương trình vi phân Dufing đã cho, ta cần phải xây dựng vế phải $k_1x - k_2x^3 - k_3\dot{x}$, tức là phải cần thêm 3 khâu P và một khâu nhân cho x^3.

Trước hết là khâu nhân. Đưa chuột vào ô tại thanh toolbar dọc mép bên trái màn hình rồi ấn chuột trái lần để gọi khối cộng/trừ/nhân/chia 

Sau khi khối được gọi, kích vào khối để đặt tham số. Trên màn hình xuất hiện:



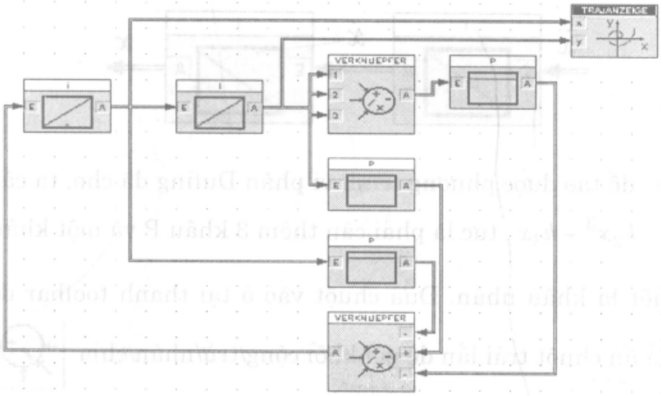
Chọn số đầu vào là 3, kiểu khối là nhân rồi ấn OK. Nối x với cả 3 đầu vào của khối nhân ta sẽ có ở đầu ra x^3.

Cuối cùng là 3 khối P. Ta gọi chúng cũng bằng cách đưa chuột vào ô có biểu tượng  và ấn phím trái ba lần.

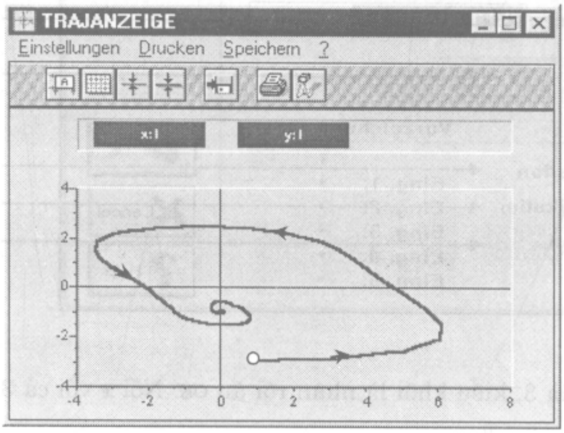
Mặc định, khi được gọi, các khối đều có chiều tín hiệu từ trái qua phải (đầu vào nằm bên trái và đầu ra nằm bên phải của khối). Có thể thay đổi tín hiệu theo chiều ngược lại, chẳng hạn cho các khối trong mạch phản hồi, ta dùng lệnh:

`Bearbeiten → Block drehen`.

Sau khi đã gọi xong các khối cần thiết, ta tiến hành việc nối khối theo đúng trình tự của phương trình Duffing. Hệ Duffing khi đó sẽ được biểu diễn thành



Nếu chọn tham số $k_1=k_2=k_3$ và vào chế độ mô phỏng, ta thu được quỹ đạo pha của hệ Duffing như sau:



7 ĐIỀU KHIỂN MỜ VÀ MẠNG NƠ-RON

7.1 Cơ sở về mạng nơ-ron

Ngay từ khi được khai sinh bằng sự ra đời cuốn sách "*Điều khiển học, hay điều chỉnh và sự truyền thông trong cơ thể sống, trong máy móc*" của tác giả Nobert Wieners xuất bản năm 1948. Điều khiển học đã đặt ra mục đích nghiên cứu áp dụng nguyên lý làm việc của hệ thống thần kinh động vật vào điều khiển. Công cụ giúp Điều khiển học thực hiện được mục đích này là trí tuệ nhân tạo và mạng nơ-ron.

Với logic mờ, trí tuệ nhân tạo phát triển mạnh mẽ trong những năm gần đây tạo ra cơ sở xây dựng các hệ chuyên gia, những hệ có khả năng cung cấp "kinh nghiệm điều khiển hệ thống" hay còn gọi là các hệ trợ giúp quyết định. Trí tuệ nhân tạo được xây dựng dựa trên mạng nơ-ron nhân tạo. Sự kết hợp giữa logic mờ và mạng nơ-ron trong thiết kế hệ thống điều khiển tự động là một khuynh hướng hoàn toàn mới, phương hướng thiết kế hệ điều khiển thông minh, một hệ thống mà bộ điều khiển có khả năng tư duy như bộ não của con người.

7.1.1 Cấu trúc và mô hình của nơ-ron

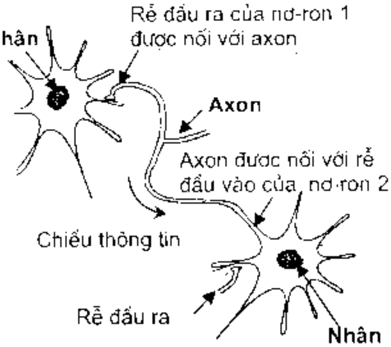
Mạng nơ-ron là sự tái tạo bằng kỹ thuật những chức năng của hệ thần kinh con người. Trong quá trình tái tạo không phải tất cả các chức năng của bộ não con người có đều được tái tạo, mà chỉ có những chức năng cần thiết. Bên cạnh đó còn có những chức năng mới được tạo ra nhằm giải quyết một bài toán điều khiển đã định hướng trước.

Mạng nơ-ron bao gồm vô số các nơ-ron được liên kết truyền thông với nhau trong mạng. *Hình 7.1* là một phần của mạng nơ-ron bao gồm hai nơ-ron.

Một nơ-ron chứa dựng các thành phần cơ bản:

– Thân nơ-ron được giới hạn trong một màng membran và trong cùng là nhân. Từ thân nơ-ron còn có rất nhiều đường rẽ nhánh tạm gọi là rễ.

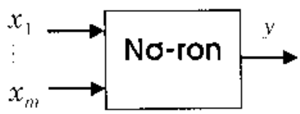
– "Bus" liên kết nơ-ron này với các nơ-ron khác được gọi là axon, trên axon có các đường rẽ nhánh. Nơ-ron còn có thể liên kết với các nơ-ron khác qua các rễ. Chính vì cách liên kết đa dạng như vậy nên mạng nơ-ron có độ liên kết rất cao.



Hình 7.1: Một mạng nơ-ron đơn giản gồm hai nơ-ron.

Các rễ của nơ-ron được chia thành hai loại: loại nhận thông tin từ nơ-ron khác qua axon, mà ta sẽ gọi là *rễ đầu vào* và loại đưa thông tin qua axon tới các nơ-ron khác, gọi là *rễ đầu ra*. Một nơ-ron có thể có nhiều rễ đầu vào, nhưng chỉ có một rễ đầu ra. Bởi vậy nếu xem nơ-ron như một khâu điều khiển thì nó chính là khâu có nhiều đầu vào, một đầu ra (khâu MISO) – *hình 7.2*.

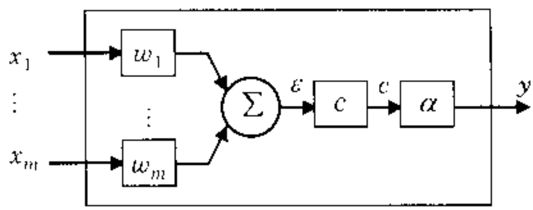
Hình 7.2:: Nơ-ron là khâu MISO



Quá trình hoạt động của một nơ-ron là một quá trình điện hóa tự nhiên. Ở trạng thái cân bằng (trạng thái tĩnh) điện áp của màng membran khoảng 75mV. Khi có tác động từ bên ngoài vào nơ-ron (mức điện áp kích thích khoảng 35mV), trong tế bào nơ-ron xảy ra hàng loạt các phản ứng hóa học tạo thành lực tác động làm nơ-ron bị kích thích hoàn toàn (bốc cháy). Thế năng sinh ra khi nơ-ron ở trạng thái bị kích thích hoàn toàn này chỉ tồn tại khoảng vài mili giây sau đó nơ-ron lại trở về trạng thái cân bằng cũ. Thế năng này được truyền vào mạng qua axon và có khả năng kích thích hoặc kìm hãm tự nhiên các nơ-ron khác trong mạng. Một nơ-ron sẽ ở trạng thái kích thích khi tại đầu vào xuất hiện một tín hiệu tác động vượt qua ngưỡng cân bằng của nơ-ron.

Một tính chất rất cơ bản của mạng nơ-ron sinh học là các đáp ứng theo kích thích có khả năng thay đổi theo thời gian. Các đáp ứng có thể tăng lên, giảm đi hoặc hoàn toàn biến mất. Qua các nhánh axon liên kết tế bào nơ-ron này với các nơ-ron khác, sự thay đổi trạng thái của một nơ-ron cũng sẽ kéo theo sự thay đổi trạng thái của những nơ-ron khác và do đó là sự thay đổi của toàn bộ mạng nơ-ron. Việc thay đổi trạng thái của mạng nơ-ron có thể thực hiện qua một quá trình "dạy" hoặc do khả năng "học" tự nhiên.

Sự thay thế những tính chất này bằng một mô hình toán học tương dương được gọi là mạng nơ-ron nhân tạo. Mạng nơ-ron nhân tạo có thể được chế tạo bằng nhiều cách khác nhau vì vậy trong thực tế tồn tại rất nhiều kiểu mạng nơ-ron nhân tạo.



Hình 7.3 : Mô hình nơ-ron có *m* đầu vào và một đầu ra.

Hình 7.3 biểu diễn một nơ-ron nhân tạo gồm *m* đầu vào $x_1, \ldots, x_m$ và một đầu ra *y*. Mô hình này gồm ba thành phần cơ bản:

- Các kích thích (các đầu vào) của tế bào nơ-ron có thế năng tác động vào màng membran khác nhau được biểu diễn qua trọng lượng w_i, $i=1,\ldots, m$ tương ứng với cường độ kích thích của từng đầu vào. Tổng giá trị của các kích thích đầu vào được thực hiện qua một bộ cộng, đó là giá trị đo kích thích đầu vào tác động vào tế bào nơ-ron.
- Đầu ra của bộ cộng được đưa đến khâu tiền đáp ứng **c**. Khâu này không chỉ có chức năng tạo ra đáp ứng tức thời *c* mà còn có khả năng lưu giữ các đáp ứng theo thời gian. Thành phần này làm việc theo nguyên lý "nhớ" động.
- Nơ-ron bị kích thích trong thời gian thế năng của màng membran vượt quá ngưỡng, mô hình nơ-ron ở trạng thái tích cực. Quan hệ này được thực hiện

nhờ khâu tạo tín hiệu ra α, nó có chức năng của khâu tạo tín hiệu ngưỡng, xác định sự phụ thuộc của tín hiệu ra y vào các kích thích đầu vào.

Cách thành lập nơ-ron nhân tạo như vậy tạo ra một độ tự do trong thiết kế. Việc lựa chọn khâu cộng tín hiệu vào, khâu tiền đáp ứng **c** và khâu tạo tín hiệu đáp ứng α, sẽ cho ra các kiểu nơ-ron nhân tạo khác nhau và tương ứng là các mô hình mạng khác nhau. Ở đây chỉ quan tâm đến các loại mạng nơ-ron phục vụ cho bài toán thiết kế bộ điều khiển mờ-nơ-ron.

Khâu cộng Σ: Khâu cộng tín hiệu vào có chức năng thực hiện phương trình sau:

$$\varepsilon = \sum_{i=1}^{m} w_i x_i$$

tức là thực hiện phép nhân vô hướng hai vec tơ $\underline{w}=\begin{pmatrix} w_1 \\ \vdots \\ w_m \end{pmatrix}$ và $\underline{x}=\begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}$

$$\varepsilon = \underline{w}^T \cdot \underline{x}. \tag{7.1}$$

Khâu tiền đáp ứng c: Những khả năng hoạt động của nơ-ron hoàn toàn phụ thuộc vào khâu tạo chức năng đáp ứng **c**. Thế năng của một màng membran càng cao nếu như giá trị ε càng lớn.

Khâu tạo chức năng đáp ứng tạo giá trị đáp ứng tăng giảm phụ thuộc vào giá trị đầu vào, một cách đơn giản nhất có thể tạo đáp ứng theo phương trình

$$c = \varepsilon. \tag{7.2a}$$

Quan hệ giữa tín hiệu vào và ra được biểu diễn theo phương trình trên là một quan hệ tĩnh và tuyến tính. Đây cũng là cách thiết kế nơ-ron nhân tạo một cách đơn giản nhất.

Để tăng độ chính xác người ta tìm cách xây dựng mô hình động cho nơ-ron. Thực tế, khi có kích thích đầu vào, thế năng của màng membran tăng dần lên cho dù ngay tại thời điểm được kích thích vẫn chưa có đáp ứng đầu ra. Hoàn toàn tương tự, khi kích thích mất đi thì nơ-ron cũng không thể trở về ngay trạng thái cân bằng cũ mà sự trở về diễn ra cũng dần dần như một quá trình liên tục theo thời gian. Quá trình đó có thể mô tả qua phương trình vi phân bậc nhất

$$T\dot{c}(t) + c(t) - c_0 = \varepsilon(t), \tag{7.2b}$$

trong đó c_0 là thế năng của mạng membran ở trạng thái không bị kích thích. Đó là phương trình động học của một khâu quán tính bậc nhất với hằng số thời gian quán tính T. Khâu tạo chức năng đáp ứng kiểu này còn có tên là BSB.

Bên cạnh khâu tạo đáp ứng **c** kiểu tuyến tính và kiểu BSB còn tồn tại các kiểu khâu theo kiểu gián đoạn. Thuộc nhóm khâu kiểu gián đoạn có khâu tạo đáp ứng theo hàm *Hopfield*.

$$c = \begin{cases} -1 & \text{khi} \quad c < 0 \\ +1 & \text{khi} \quad c \geq 0 \end{cases} \tag{7.2c}$$

Khâu tạo đáp ứng α: Giá trị ra y của một nơ-ron biểu diễn trạng thái kích thích đến các nơ-ron tiếp theo trong mạng. Tín hiệu ra y phụ thuộc vào độ kích thích của nơ-ron, thông thường được so sánh theo kiểu cắt ngưỡng. Quan hệ này được mô tả qua khâu tạo đáp ứng α của mô hình nơ-ron. Thông thường giá trị ra y phải thay đổi liên tục theo sự thay đổi của tiền đáp ứng c hoặc không thay đổi nếu c nhỏ hơn giá trị ngưỡng. Sự thay đổi này là đồng biến (monoton). Có rất nhiều phương pháp xây dựng khâu tạo đáp ứng, do vậy nó được chọn trong phần lớn các phương án đã có.

Xuất phát từ quan điểm logic kinh điển, có thể coi nơ-ron như một phần tử làm việc theo nguyên tắc đóng mở, tức là chỉ tồn tại ở hai trạng thái: nơ-ron tích cực (bị kích thích) và ở trạng thái không tích cực, ta tạo ra được bộ tạo chức năng ra đơn giản nhất. Khâu tạo đáp ứng, trong trường hợp này, có thể biểu diễn dưới dạng hàm bậc thang

$$y = \alpha(c) = \begin{cases} 0 & \text{khi} \quad c < c_0 \\ 1 & \text{khi} \quad c \geq c_0 \end{cases} \tag{7.3a}$$

với bước nhảy tại thời điểm $c=c_0$. Giá trị ra y có thể là một hằng số nào đó, không nhất thiết phải là 0 và 1. Giá trị c_0 chính là ngưỡng quyết định trạng thái của nơ-ron (bị kích thích hay không bị kích thích).

Thông thường sự chuyển đổi trạng thái của nơ-ron từ không tích cực sang tích cực và ngược lại là quá trình liên tục. Quá trình chuyển đổi này có thể so sánh với quá trình chuyển đổi từ tập rõ sang tập mờ, đó là quá trình chuyển đổi trơn từ giá trị 0 sang giá trị 1. Một trong những khâu mô tả được quá trình liên tục đó là khâu *Sigma* biểu diễn dưới dạng hàm *Fermi*

$$y = \alpha(c) = \frac{1}{1 + e^{-c}}. \tag{7.3b}$$

Tất nhiên cũng tồn tại nhiều khâu α mô tả quá trình chuyển đổi tuyến tính khác thường được sử dụng. Một số khâu như vậy là

– Khâu lưỡng tuyến tính $y = \begin{cases} 1 + \operatorname{sgn}(c) & \text{khi} \quad |c| > 1 \\ 1 + c & \text{khi} \quad |c| \le b \end{cases}$ (7.3c)

– Khâu tuyến tính $y = k\,c$, với k là hằng số. (7.3d)

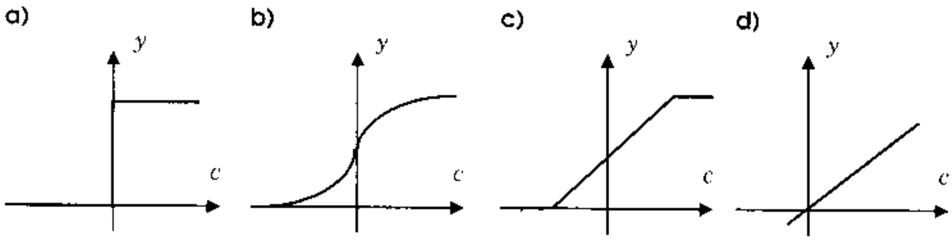
– Đồng dạng $y = c$. (7.3e)

Ngoài ra, bên cạnh các khâu tiền định vừa nêu còn có loại khâu tạo đáp ứng ngẫu nhiên. Khâu này cho ra giá trị y nhị phân (hoặc bằng 0 hoặc bằng 1) tại đầu ra. Điểm khác biệt so với loại khâu tiền định là hàm mô tả khâu không ở dạng tiền định như (7.3a) hay (7.3b) mà là một hàm thông báo xác suất để có được $y=1$ tại đầu ra. Một trong đại diện cho mô hình mô tả khâu ngẫu nhiên này là hàm *Boltzman* định nghĩa như sau

$$p(y=1) = \frac{1}{1 + e^{-\frac{c - c_0}{T}}} , \tag{7.3f}$$

trong đó c_0 là giá trị ngưỡng và tham số T là một đại lượng vật lý biểu diễn độ nhạy cảm của nơ-ron.

Hình 7.4 biểu diễn đồ thị các khâu tạo đáp ứng tiền định vừa nêu.



Hình 7.4 : Một số khâu liên tục thường dùng để tạo đáp ứng y từ tiền đáp ứng c.

Tất cả các dạng hàm mô tả trên của khâu tạo đáp ứng α đều cho ra đáp ứng y là một tín hiệu nằm trong khoảng từ 0 đến 1. Tuy vậy không bắt buộc ta phải giữ nguyên miền giá trị đó của y. Tùy vào từng bài toán ứng dụng, ta có thể thay đổi (7.3a), (7.3b) hoặc (7.3c) sao cho y nhận giá trị chẳng hạn như từ -1 đến $+1$.

7.1.2 Những mô hình nơ-ron thường sử dụng

Mỗi một kết nối từ vector tín hiệu vào $\underline{x}$ tới tín hiệu ra y, qua đặc tính của khâu cộng $\sum$ với hàm mô tả (7.1), khâu tiền đáp ứng c với (7.2a)÷(7.2c) và khâu tạo đáp ứng α với (7.3a)÷(7.3e) sẽ cho ra một mô hình nơ-ron. Như vậy tổng cộng sẽ có tất cả là 15 mô hình nơ-ron. Giá trị đầu ra y của nơ-ron là

$$y = \alpha(c) = \alpha(c(\varepsilon)) = \alpha(c(\underline{w}^T \cdot \underline{x})).$$

Chẳng hạn, nếu chọn (7.2a) cho khâu tiền đáp ứng c, (7.3b) cho khâu tạo đáp ứng α ta sẽ có

$$y = \frac{1}{1+e^{-c}} = \frac{1}{1+e^{-\varepsilon}} = \frac{1}{1+e^{-\underline{w}^T\underline{x}}} \tag{7.4}$$

và mô hình này có tên là *mô hình Fermi*.

Tuy nhiên, phổ biến nhất trong số 15 mô hình nơ-ron là 6 loại cho trong bảng sau.

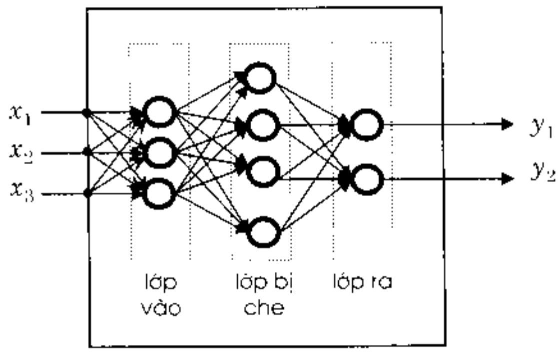
Tên gọi	Khâu c	Khâu α	Tên gọi	Khâu c	Khâu α
McCulloch–Pitts	(7.2a)	(7.3a)	Adeline	(7.2a)	(7.3e)
Fermi	(7.2a)	(7.3b)	Boltzmann	(7.2a)	(7.3f)
BSB	(7.2b)	(7.3c)	Hopfield	(7.2c)	(7.3e)

7.1.3 Cấu tạo mạng nơ-ron

Dựa trên những phương pháp xây dựng nơ-ron đã trình bày trong mục 1.1, ta có thể coi nơ-ron như một hệ MISO truyền đạt và xử lý tín hiệu. Đặc tính truyền đạt của nơ-ron phần lớn là đặc tính truyền đạt tĩnh, chỉ khi chọn khâu chức năng đáp ứng kiểu BSB thì lúc đó nơ-ron có đặc tính động. Trong mọi trường hợp do đặc tính phi tuyến của khâu tạo chức năng ra kết hợp và/hoặc với đặc tuyến phi tuyến của khâu tạo chức năng đáp ứng mà nơ-ron là một hệ có tính phi tuyến mạnh.

Liên kết các đầu vào và ra của nhiều nơ-ron với nhau ta được một mạng nơ-ron. Việc ghép nối các nơ-ron trong mạng với nhau có thể theo một nguyên tắc bất kỳ nào đó, vì về nguyên tắc một nơ-ron là một hệ MISO. Từ đó có thể phân biệt các loại nơ-ron khác nhau như các loại nơ-ron mà các đầu vào nhận thông tin từ môi trường bên ngoài với các loại nơ-ron mà các đầu vào được nối với các nơ-ron khác trong mạng. Các nơ-ron mà đầu vào giữ chức năng nhận thông tin từ môi trường bên ngoài đóng chức năng "đầu vào" của mạng. Cũng tương tự như vậy, một nơ-ron có một đầu ra, đầu ra của nơ-ron này có thể là đầu vào của nhiều nơ-ron khác hoặc có thể đưa ra môi môi trường bên ngoài. Những nơ-ron có đầu ra đưa tín hiệu vào môi trường bên ngoài được gọi là "đầu ra" của mạng. Như vậy một mạng nơ-ron cũng có chức năng của một hệ truyền đạt và xử lý tín hiệu từ đầu vào đến đầu ra của mạng. Các nơ-ron trong một mạng thường được chọn cùng một loại, chúng được phân biệt với nhau qua vec tơ hàm trọng lượng ở đầu vào $\underline{w}$.

Nguyên lý cấu tạo của một mạng nơ-ron bao gồm nhiều lớp, mỗi lớp bao gồm nhiều nơ-ron có cùng một chức năng trong mạng. Trong *hình 7.5* là mô hình của một mạng nơ-ron ba lớp với 9 nơ-ron. Mạng có ba đầu vào x_1, x_2, x_3 và hai đầu ra y_1, y_2. Các tín hiệu đầu vào được đưa đến ba nơ-ron đầu vào, ba nơ-ron này làm thành lớp đầu vào của mạng (input layer). Các nơ-ron trong lớp này được gọi là *nơ-ron đầu vào.* Đầu ra của các nơ-ron này được đưa đến đầu vào của bốn nơ-ron tiếp theo, bốn nơ-ron này không trực tiếp tiếp xúc với môi trường xung quanh và làm thành lớp trung gian trong mạng (hidden layer). Các nơ-ron trong lớp này có tên là *nơ-ron nội* hay *nơ-ron bị che*. Đầu ra của các nơ-ron này được đưa đến hai nơ-ron đưa tín hiệu ra môi trường bên ngoài. Các nơ-ron trong lớp đầu ra này có tên là *nơ-ron đầu ra* (output layer).

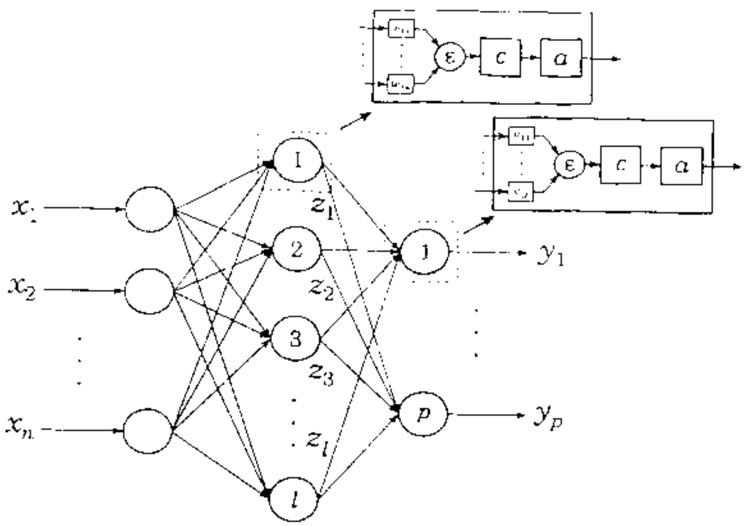


Hình 7.5 : Mạng nơ-ron ba lớp

Mạng nơ-ron được xây dựng như trên là mạng gồm ba lớp mắc nối tiếp nhau đi từ đầu vào đến đầu ra. Trong mạng không tồn tại bất kỳ một mạch hồi tiếp nào kể cả hồi tiếp nội lẫn hồi tiếp từ đầu ra trở về đầu vào. Một mạng nơ-ron có cấu tạo như vậy được gọi là mạng truyền thẳng (*feedforward network*). Mạng nơ-ron có đường phản hồi từ đầu ra của một nơ-ron tới đầu vào của nơ-ron cùng lớp hoặc thuộc lớp phía trước có tên gọi là mạng hồi tiếp (*feedback network*).

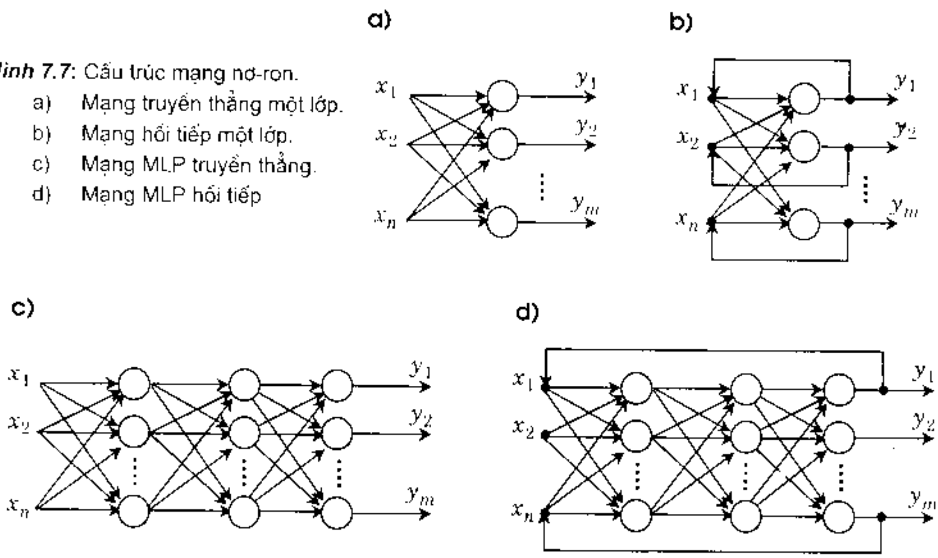
Mạng nơ-ron bao gồm một hay nhiều lớp trung gian được gọi là mạng MLP (*multilayer perceptrons Network*). Còn mạng chỉ có một lớp, vừa là lớp vào vừa là lớp trung gian và cũng là lớp ra thì mạng đó có tên là *mạng một lớp*.

Hình 7.6 : Mạng MLP.



Hình 7.7: Cấu trúc mạng nơ-ron.

a) Mạng truyền thẳng một lớp.
b) Mạng hồi tiếp một lớp.
c) Mạng MLP truyền thẳng.
d) Mạng MLP hồi tiếp



Mạng nơ-ron có cấu trúc mạng ghép nối hoàn toàn, tức là bất cứ một nơ-ron nào trong mạng cũng được nối với một hoặc vài nơ-ron khác. Trong trường hợp các nơ-ron trong mạng có khâu tạo chức năng đáp ứng là khâu tuyến tính, tính phi tuyến chỉ nằm ở khâu tạo chức năng ra thì việc mắc nối tiếp các nơ-ron trong mạng không còn ý nghĩa nữa và lúc đó ta hoàn toàn có thể thay thế mạng nơ-ron nhiều lớp thành mạng nơ-ron một lớp.

7.1.4 Phương thức làm việc của mạng nơ-ron

Phương thức làm việc của một mạng nơ-ron nhân tạo có thể phân chia làm hai giai đoạn:

- tự tái tạo lại (*reproduction*)

  và giai đoạn học (*learning phase*).

Ở một mạng nơ-ron có cấu trúc bền vững có nghĩa là vectơ hàm trọng lượng đầu vào, khâu tạo đáp ứng và khâu tạo tín hiệu đầu ra đều cố định không bị thay đổi về mặt cấu trúc cũng như tham số thì mạng có một quá trình truyền đạt xác định chắc chắn, tĩnh hoặc động phụ thuộc vào cấu tạo của các nơ-ron trong mạng. Ở đầu vào của mạng xuất hiện thông tin thì tại đầu ra cũng xuất hiện một đáp ứng tương ứng. Đối với mạng nơ-ron có quá trình truyền đạt tĩnh, đáp ứng đầu ra xuất hiện ngay sau khi đầu vào nhận được thông tin, còn đối với mạng nơ-ron có quá trình truyền đạt động thì phải sau một thời gian quá độ ở đầu ra của mạng nơ-ron mới xuất hiện đáp ứng. Xuất phát từ quan điểm mọi đáp ứng của nơ-ron đều tiền định tự nhiên, có nghĩa là khi xuất hiện các kích thích ơ đầu vào của mạng ở các thời điểm khác nhau các giá trị như nhau thì đáp ứng ở đầu ra ở các thời điểm tương ứng cũng hoàn toàn giống nhau. Quá trình làm việc như vậy của một mạng nơ-ron được gọi là quá trình tái diễn lại (*reproduction phase*). Khi có thông tin ở đầu vào mạng lưu giữ thông tin đó và dựa trên các tri thức của mình đưa ra đáp ứng ở đầu ra phù hợp với lượng thông tin thu được từ đầu vào.

Mạng nơ-ron khi mới hình thành còn chưa có tri thức, tri thức của mạng hình thành dần sau một quá trình học. Mạng nơ-ron được dạy bằng cách đưa vào đầu vào những kích thích và mạng hình thành những đáp ứng tương ứng, những đáp ứng phù hợp với từng loại kích thích sẽ được lưu giữ, giai đoạn này được gọi là giai đoạn học của mạng. Khi đã hình tri thức mạng có thể giải quyết các vấn đề cụ thể một cách đúng đắn. Đó có thể là những vấn đề ứng dụng rất khác nhau, được giải quyết chủ yếu dựa trên sự tổ chức hợp nhất giữa các thông tin đầu vào của mạng và các đáp ứng đầu ra:

1) Nhiệm vụ của một mạng liên kết là hoàn chỉnh hoặc hiệu chỉnh các thông tin thu thập được không đầy đủ hoặc bị tác động của nhiễu. Mạng nơ-ron kiểu này được ứng dụng trong lĩnh vực hoàn thiện mẫu, mà một trong những lĩnh vực cụ thể đó là nhận dạng chữ viết.

2) Nhiệm vụ tổng quát của một mạng nơ-ron là lưu giữ động các thông tin. Dạng thông tin lưu giữ đó chính là quan hệ giữa các thông tin đầu vào của mạng và các đáp ứng đầu ra tương ứng, để khi có một kích thích bất kỳ tác động vào mạng, mạng có khả năng suy diễn và đưa ra một đáp ứng phù hợp. Đó chính là chức năng nhận dạng theo mẫu của mạng nơ-ron. Để thực hiện chức năng này, mạng nơ-ron đóng vai trò như một bộ phận tổ chức các nhóm thông tin đầu vào và tương ứng với mỗi nhóm là một đáp ứng đầu ra phù hợp. Như vậy một nhóm bao gồm một loại thông tin đầu vào và một đáp ứng ra. Các nhóm có thể được hình thành trong quá trình học và cũng có thể hình thành không trong quá trình học.

Trong lĩnh vực ứng dụng, mạng nơ-ron có khả năng tạo ra các đáp ứng đầu ra dựa trên thông tin thu thập vào mạng, điều đó có nghĩa là ứng với một thông tin xác định ở đầu vào mạng cung cấp một đáp ứng tương ứng xác định ở đầu ra. Nhìn trên quan điểm lý thuyết hệ thống, mạng nơ-ron được coi như một bộ xấp xỉ thông tin, thiết bị này có khả năng cung cấp một quá trình xử lý mong muốn một cách chính xác. Mục đích của quá trình học là tạo ra một tri thức cho mạng thông qua rèn luyện. Nguyên tắc học được thực hiện cho một mạng mà cấu trúc của mạng cũng như các phần tử nơ-ron cố định, chính là thay đổi giá trị của các phần tử trong vector hàm trọng lượng, vector ghép nối giữa các phần tử nơ-ron trong mạng. Các phần tử này được chọn sao cho quá trình truyền đạt mong muốn được xấp xỉ một cách đủ chính xác như bài toán yêu cầu. Để đạt được mục đích đó, người ta cho tác động vào đầu vào của mạng hàng loạt các tác động $\underline{x}^{(k)}$, k=1, 2, ... có khả năng lặp lại trong quá trình mạng làm việc. Những tác động này được gọi là tác động mẫu. Các đáp ứng $\underline{\widetilde{y}}^{(k)}$ của tác động mẫu được so sánh với đáp ứng mẫu $\underline{y}^{(k)}$ chọn trước và các phần tử của vector hàm trọng lượng $\underline{w}$ được hiệu chỉnh sao cho sai lệch so với mẫu mong muốn là nhỏ nhất. Quá trình chỉnh định này sẽ được thực hiện cho đến khi đạt được sai số mong muốn nào đó. Mạng lúc này đã có được một đáp ứng đầu ra hoàn toàn phù hợp với tác động mẫu đầu vào và kết quả này sẽ được cất giữ.

Như vậy, học chính là quá trình giải bài toán tối ưu tham số. Để thực hiện được bài toán tối ưu tham số này phải xây dựng được phiếm hàm mục đích mô tả sai lệch giữa $\underline{y}^{(k)}$ và $\underline{\widetilde{y}}^{(k)}$.

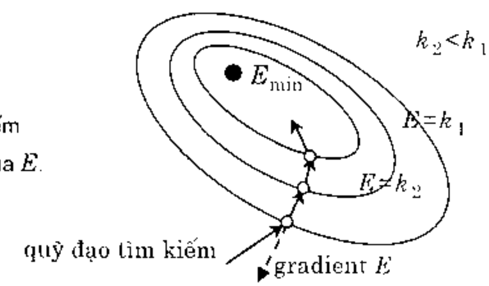
Cùng quan sát một mạng nơ-ron MLP ba lớp: lớp vào có n đầu vào $x_1, x_2, \cdots, x_n$ lớp bị che gồm l nơ-ron và lớp ra với p đầu ra $y_1, y_2, \cdots, y_p$ như trong *hình 7.6* mô tả. Các nơ-ron đầu vào chỉ đóng vai trò thuần túy thu thập thông tin tác động vào mạng và không tham gia vào quá trình học. Quá trình học chỉ thực hiện thông qua các nơ-ron ở lớp trung gian và ở lớp ra. Vector hàm trọng lượng $\underline{w}_i$ của nơ-ron thứ i của lớp trung gian bao gồm n phần tử $w_{i1}, w_{i2}, \cdots, w_{in}$ và vector hàm trọng lượng $\underline{w}_j$ của nơ-ron ở lớp đầu ra bao gồm l phần tử $w_{j1}, w_{j2}, \cdots, w_{jl}$. Đầu ra của các nơ-ron lớp trung gian có ký hiệu là $z_1, z_2, \cdots, z_l$.

Giả sử, bắt đầu một quá trình học với m cặp vec tơ vào/ra $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, ứng với mỗi vec tơ đầu vào $\underline{x}^{(k)}$ nhận được một vector đáp ứng đầu ra $\underline{\tilde{y}}^{(k)}$. Sai lệch E_k ứng với mẫu học thứ k được biểu diễn dưới dạng hàm mục đích bằng bình phương sai lệch giữa mẫu $\underline{y}^{(k)}$ và đáp ứng ra thực của mạng $\underline{\tilde{y}}^{(k)}$ có dạng :

$$E_k = \frac{1}{2}\left[\underline{y}^{(k)} - \underline{\tilde{y}}^{(k)}\right]^2 = \frac{1}{2}\sum_{j=1}^{p}\left[y_j^{(k)} - \tilde{y}_j^{(k)}\right]^2$$

Mặc dù quá trình học được thực hiện theo từng bước, nhưng hàm mục đích vẫn phải được xây dựng cho toàn bộ chu trình học:

$$E = \sum_{(k)} E_k = \frac{1}{2}\sum_{(k)}\sum_{j=1}^{p}\left[y_j^{(k)} - \tilde{y}_j^{(k)}\right]^2 .$$



Hình 7.7: Minh họa phương pháp tìm kiếm $E_{\min}$ theo hướng ngược gradient của E.

Quá trình học là thực hiện nhiệm vụ *tìm giá trị cực tiểu* $E_{\min}$ cho hàm mục đích E bằng cách thay đổi giá trị của các phần tử trong vector hàm trọng lượng sao cho sau mỗi lần thay đổi như vậy giá trị của E chẳng hạn đang là k_1 sẽ được giảm đi một ít thành $k_2 < k_1$ (*hình 7.7*). Thông thường việc giải bài toán tối ưu này được thực hiện theo phương pháp truy hồi qua nhiều bước.

Một trong những cách thay đổi vector hàm trọng lượng nhằm làm giảm giá trị của E là thay đổi theo hướng ngược gradient vì vector gradient E là $\left(\frac{\partial E}{\partial \underline{w}}\right)^T$ luôn có hướng chỉ chiều tăng giá trị của E. Nguyên tắc học này có thể diễn đạt dưới dạng

$$\Delta w_{pr} = -s\frac{\partial E}{\partial w_{qr}} \quad ; \; q = 1, 2, \dots, l \qquad r = 1, 2, \dots, n$$

cho lớp nơ-ron trung gian và

$$\Delta v_{pr} = -s\frac{\partial E}{\partial w_{qr}} \quad ; \quad q = 1, 2, \dots, p \qquad r = 1, 2, \dots, l$$

cho lớp nơ-ron đầu ra và w_{qr} mới được xác định từ giá trị w_{qr} cũ và một lượng gia tăng

$$w_{qr}^{(mới)} = w_{qr}^{(cũ)} + \Delta w_{qr}$$

và

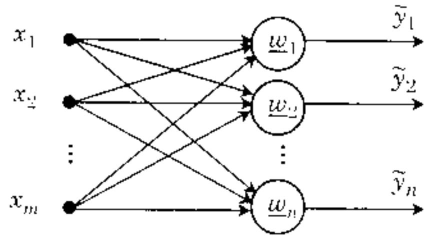
$$v_{qr}^{(mới)} = v_{qr}^{(cũ)} + \Delta v_{qr}$$

Đây thực sự là một bài toán tối ưu số và có thể giải quyết dễ dàng với sự trợ giúp của các thuật toán tìm kiếm ngược gradient. Tham số s được gọi là *bước học* và thường có thể được chọn trước trong khoảng từ 0,1 đến 0,9.

Bên cạnh phương thức học có hướng dẫn theo hàm mục tiêu E (*supervised learning*) mô tả ở trên mạng nơ-ron còn có thể được học một cách hoàn toàn tự do không cần phiếm hàm mục tiêu (*unsupervised learning*). Mẫu cho trước là đáp ứng ở đầu ra của mạng ứng với mẫu đầu vào. Hay nói một cách khác ứng với một mẫu đầu vào có một mẫu đầu ra. Quá trình học là quá trình sắp xếp các nhóm tín hiệu đầu vào và đầu ra phù hợp với nhau. Đây là một quá trình học không được theo dõi và điều khiển mờ ít quan tâm đến các mạng nơ-ron xây dựng theo nguyên lý học kiểu như vậy.

7.2 Mạng truyền thẳng một lớp

Mạng nơ-ron một lớp là mạng có cấu trúc đơn giản nhất. Tính đơn giản này sẽ hoàn toàn không hạn chế miền ứng dụng của nó vì mọi mạng MLP với các nơ-ron thuộc lớp trung gian (nơ-ron bị che) và lớp đầu vào có tính tuyến tính đều chuyển được về dạng một lớp. Do có khả năng đại diện tổng quát cho mạng MLP với tính tuyến tính của các nơ-ron lớp trung gian và lớp đầu vào như vậy nên mạng nơ-ron một lớp còn có tên gọi là *mạng tuyến tính*.



Hình 7.8 : Mạng nơ-ron truyền thẳng một lớp với m đầu vào, n đầu ra.

7.2.1 Mạng Adaline

Xét mạng một lớp cho ở hình 7.8 với m đầu vào x_j, j=1, 2, ... , m viết chung dưới dạng vector là $\underline{x}=\begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}$ và n đầu ra $\underline{\widetilde{y}}=\begin{pmatrix} \widetilde{y}_1 \\ \vdots \\ \widetilde{y}_n \end{pmatrix}$. Như vậy mạng cũng sẽ có n nơ-ron.

Gọi vector trọng lượng của nơ-ron thứ i là $\underline{w}_i=\begin{pmatrix} w_{i1} \\ \vdots \\ w_{im} \end{pmatrix}$ thì mạng với

$$\widetilde{y}_i=\underline{w}_i^T\underline{x}\,, \quad i=1,2,\ldots,n \tag{7.5}$$

có tên gọi là mạng Adeline (mạng có phần tử thích nghi tuyến tính *Adaptive Linear Element*).

Nhiệm vụ việc dạy mạng là xác định w_{ij}, i=1, 2, ... , n ; j=1, 2, ... , m để đầu ra $\underline{\widetilde{y}}^{(k)}=\begin{pmatrix} \widetilde{y}_1^{(k)} \\ \vdots \\ \widetilde{y}_n^{(k)} \end{pmatrix}$ ứng với tác động $\underline{x}^{(k)}=\begin{pmatrix} x_1^{(k)} \\ \vdots \\ x_m^{(k)} \end{pmatrix}$ được "giống như" mẫu $\underline{y}^{(k)}=\begin{pmatrix} y_1^{(k)} \\ \vdots \\ y_n^{(k)} \end{pmatrix}$ mong muốn, tức là để có

$$y_i^{(k)} \approx \tilde{y}_i^{(k)} = \underline{w}_i^T \underline{x}^{(k)}, \qquad i=1, 2, \ldots, n$$

$$\Leftrightarrow \qquad \underline{y}^{(k)} \approx \underline{\tilde{y}}^{(k)} = W \underline{x}^{(k)} \tag{7.6}$$

trong đó $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, $k=1, 2, \ldots, p$ là những mẫu học cho trước và

$$\underline{y}^{(k)} = \begin{pmatrix} y_1^{(k)} \\ \vdots \\ y_n^{(k)} \end{pmatrix}, \quad \underline{\tilde{y}}^{(k)} = \begin{pmatrix} \tilde{y}_1^{(k)} \\ \vdots \\ \tilde{y}_n^{(k)} \end{pmatrix}, \quad \underline{x}^{(k)} = \begin{pmatrix} x_1^{(k)} \\ \vdots \\ x_m^{(k)} \end{pmatrix}, \quad W = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nm} \end{pmatrix}.$$

Hệ phương trình (7.6) gồm có $n \times p$ phương trình cho $n \times m$ ẩn số. Để thu được $\underline{y}^{(k)} = \underline{\tilde{y}}^{(k)}$ cho toàn bộ p mẫu học thì cần thiết phải có $p \le m$ mà điều này thường không được thỏa mãn. Bởi vậy việc dạy học cho mạng sẽ được thực hiện theo sự hướng dẫn (*supervised learning*) của hàm mục tiêu như sau

$$E = \frac{1}{2} \sum_{k=1}^{p} \left[\underline{y}^{(k)} - \underline{\tilde{y}}^{(k)}\right]^2 = \frac{1}{2} \sum_{k=1}^{p} \left[\underline{y}^{(k)} - W\underline{x}^{(k)}\right]^2 \rightarrow \min.$$

Từ đây ta suy ra được với bước học s

$$\Delta w_{ij} = -s \frac{\partial E}{\partial w_{ij}} = s \sum_{k=1}^{p} \left[\underline{y}^{(k)} - W\underline{x}^{(k)}\right] \frac{\partial W \underline{x}^{(k)}}{\partial w_{ij}}$$

$$= s \sum_{k=1}^{p} \left[y_i^{(k)} - \underline{w}_i^T \underline{x}^{(k)}\right] x_j^{(k)}, \tag{7.7}$$

vì w_{ij} chỉ ảnh hưởng tới một mình nơ-ron thứ i.

Nhược điểm của công thức (7.7) để hiệu chỉnh trọng số w_{ij} là phải có ngay một lúc toàn bộ p mẫu học $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, $k=1, 2, \ldots, p$. Nhằm khắc phục, Widnow năm 1962 đã đề nghị sử dụng dạng cải biên của nó ở bước học thứ k thành:

$$\Delta w_{ij}^{(k)} = s\left[\, y_i^{(k)} - \underline{w}_i^T \underline{x}^{(k)}\right] x_j^{(k)} \tag{7.8}$$

và với giá trị gia tăng Δw_{ij} trên ta có được sự hiệu chỉnh w_{ij} qua từng bước học k một cách đơn giản như sau

$$w_{ij}^{(k+1)} = w_{ij}^{(k)} + \Delta w_{ij}^{(k)} \tag{7.9}$$

7.2.2 Nơ-ron Hopfield và mạng tuyến tính có ngưỡng (LTU)

Gần giống như mạng Adeline, mạng tuyến tính ở *hình 7.8* với

$$\tilde{y}_i = \text{sgn}(\underline{w}_i^T \underline{x}), \qquad i=1, 2, \dots, n \tag{7.10}$$

được gọi là mạng với nơ-ron Hopfield. Từ (7.10) ta có hàm mục tiêu cho quá trình xác định tri thức w_{ij} từ các mẫu học $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, $k=1, 2, \dots$

$$E = \frac{1}{2}\sum_{k=1}^{p}\left[\underline{y}^{(k)} - \underline{\tilde{y}}^{(k)}\right]^2 = \frac{1}{2}\sum_{k=1}^{p}\left[\underline{y}^{(k)} - \text{sgn}(W\underline{x}^{(k)})\right]^2 \to \min. \tag{7.11}$$

Do đó nếu gọi s là bước học, thì với (7.11) ta có công thức tính giá trị gia tăng Δw_{ij} ở bước học thứ k đã được cải biên theo công thức Widnow (7.8) như sau

$$\Delta w_{ij}^{(k)} = s[\, y_i^{(k)} - \text{sgn}(\underline{w}_i^T \underline{x}^{(k)})]\, x_j^{(k)} = \begin{cases} 2sy_i^{(k)}x_j^{(k)} & \text{nếu} \quad y_i^{(k)} \neq \tilde{y}_i^{(k)} \\ 0 & \text{cho các trường hợp khác} \end{cases} \tag{7.12}$$

trong đó

$$\tilde{y}_i^{(k)} = \text{sgn}(\underline{w}_i^T \underline{x}^{(k)}) \qquad \text{với} \qquad \underline{x}^{(k)} = \begin{pmatrix} x_1^{(k)} \\ \vdots \\ x_m^{(k)} \end{pmatrix}$$

vì $y_i^{(k)}$ chỉ có giá trị là -1 hoặc 1.

Mạng tuyến tính có ngưỡng (*Linear Threshhold Units*, viết tắt thành LTU) là mạng một lớp với

$$\tilde{y}_i = \text{sgn}(\underline{w}_i^T \underline{x} - c_0), \qquad i=1, 2, \dots, n \tag{7.13}$$

Ghép $\underline{w}_i$ và c_0 chung lại thành $\underline{\tilde{w}}_i = \begin{pmatrix} w_i \\ c_0 \end{pmatrix}$ cũng như $\underline{x}$ và -1 thành $\underline{\tilde{x}} = \begin{pmatrix} x \\ -1 \end{pmatrix}$ thì mạng LTU sẽ đưa được về dạng Hopfield như sau

$$\tilde{y}_i = \text{sgn}(\underline{\tilde{w}}_i^T \underline{\tilde{x}}), \qquad i=1, 2, \dots, n$$

Bởi vậy cho mạng LTU ta cũng có giá trị gia tăng

$$\Delta w_{ij}^{(k)} = \begin{cases} 2sy_i^{(k)}x_j^{(k)} & \text{nếu} \quad y_i^{(k)} \neq \tilde{y}_i^{(k)} \\ 0 & \text{cho các trường hợp khác} \end{cases} \tag{7.14}$$

Ví dụ (theo [1]): Xét mạng LTU một đầu vào x và một đầu ra $\tilde{y}$ với bốn mẫu học $\{x^{(k)}$ $y^{(k)}\}$, k=1, 2, 3, 4 như sau

$$\{0{,}5 \quad 1\}, \qquad \{-1 \quad -1\}, \qquad \{2 \quad 1\}, \qquad \{-2 \quad -1\}.$$

Tạo ra các vector

$$\underline{\tilde{x}}^{(1)} = \begin{pmatrix} x^{(1)} \\ -1 \end{pmatrix} = \begin{pmatrix} 0{,}5 \\ -1 \end{pmatrix}, \qquad \underline{\tilde{x}}^{(2)} = \begin{pmatrix} x^{(2)} \\ -1 \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \end{pmatrix},$$

$$\underline{\tilde{x}}^{(3)} = \begin{pmatrix} x^{(3)} \\ -1 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \qquad \underline{\tilde{x}}^{(4)} = \begin{pmatrix} x^{(4)} \\ -1 \end{pmatrix} = \begin{pmatrix} -2 \\ -1 \end{pmatrix}.$$

Vậy thì khi xuất phát từ giá trị

$$\underline{\tilde{w}}^{(1)} = \begin{pmatrix} w \\ c_0 \end{pmatrix} = \begin{pmatrix} -2 \\ 1{,}5 \end{pmatrix}$$

ta sẽ có với s=0,5

1) k=1: $\quad \tilde{y}^{(1)} = \mathrm{sgn}[\left(\underline{\tilde{w}}^{(1)}\right)^T \underline{\tilde{x}}^{(1)}] = \mathrm{sgn}[\left(-2 \quad 1{,}5\right)\begin{pmatrix} 0{,}5 \\ -1 \end{pmatrix}] = -1 \neq y^{(1)} = 1$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(2)} = \underline{\tilde{w}}^{(1)} + 2s\, y^{(1)}\, \underline{\tilde{x}}^{(1)} = \underline{\tilde{w}}^{(1)} + \underline{\tilde{x}}^{(1)} = \begin{pmatrix} -1{,}5 \\ 0{,}5 \end{pmatrix}$$

2) k=2: $\quad \tilde{y}^{(2)} = \mathrm{sgn}[\left(-1{,}5 \quad 0{,}5\right)\begin{pmatrix} -1 \\ -1 \end{pmatrix}] = 1 \neq y^{(2)} = -1$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(3)} = \underline{\tilde{w}}^{(2)} + 2s\, y^{(2)}\, \underline{\tilde{x}}^{(2)} = \underline{\tilde{w}}^{(2)} - \underline{\tilde{x}}^{(2)} = \begin{pmatrix} -0{,}5 \\ 1{,}5 \end{pmatrix}$$

3) k=3: $\quad \tilde{y}^{(3)} = \mathrm{sgn}[\left(-0{,}5 \quad 1{,}5\right)\begin{pmatrix} 2 \\ -1 \end{pmatrix}] = -1 \neq y^{(3)} = 1$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(4)} = \underline{\tilde{w}}^{(3)} + 2s\, y^{(3)}\, \underline{\tilde{x}}^{(3)} = \underline{\tilde{w}}^{(3)} + \underline{\tilde{x}}^{(3)} = \begin{pmatrix} 1{,}5 \\ 0{,}5 \end{pmatrix}$$

4) k=4: $\quad \tilde{y}^{(4)} = \mathrm{sgn}[\left(1{,}5 \quad 0{,}5\right)\begin{pmatrix} -2 \\ -1 \end{pmatrix}] = -1 = y^{(4)}$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(5)} = \underline{\tilde{w}}^{(4)} = \begin{pmatrix} 1{,}5 \\ 0{,}5 \end{pmatrix}.$$

Suy ra, sau 4 bước học mạng sẽ có tri thức là:

$$w = 1{,}5 \qquad \text{và} \qquad c_0 = 0{,}5.$$

7.2.3 Mạng LGU

Dạng tổng quát của mô hình mạng tuyến tính một lớp ở *hình 7.8* là mạng LGU (*linear graded units*) với

$$\tilde{y}_i = \alpha(\underbrace{\underline{w}_i^T \underline{x}}_{c_i}) = \alpha(c_i), \qquad i=1, 2, \ldots, n \tag{7.15}$$

Dạy học cho mạng (7.15) là công việc xác định bộ tham số $\underline{w}_i$, i=1, 2, ... , n từ những mẫu học $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, k=1, 2, ... cho trước sao cho sai lệch

$$E = \frac{1}{2}\sum_{(k)}\left[\underline{y}^{(k)} - \underline{\tilde{y}}^{(k)}\right]^2 = \frac{1}{2}\sum_{(k)}\left[\underline{y}^{(k)} - \alpha(c_i^{(k)})\right]^2$$

với $\quad c_i^{(k)} = \underline{w}_i^T \underline{x}^{(k)}$

là nhỏ nhất. Điều này dẫn đến công thức xác định độ gia tăng Δw_{ij}

$$\Delta w_{ij} = -s\frac{\partial E}{\partial w_{ij}} = s\sum_{(k)}\left[\underline{y}^{(k)} - \alpha(c_i^{(k)})\right]\left.\frac{d\alpha}{dc_i}\right|_{c_i^{(k)}} x_j^{(k)}$$

Ở đây ta lại sử dụng lại một lần nữa tư tưởng cải biên của Widnow để tránh việc phải có tất cả toàn bộ mẫu học $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, k=1, 2, ... cùng một lúc khi tính Δw_{ij} và đi đến công thức sau cho riêng bước học thứ k

$$\Delta w_{ij}^{(k)} = s\left[\underline{y}^{(k)} - \alpha(c_i^{(k)})\right]\left.\frac{d\alpha}{dc_i}\right|_{c_i^{(k)}} x_j^{(k)} \tag{7.16a}$$

$$w_{ij}^{(k+1)} = w_{ij}^{(k)} + \Delta w_{ij}^{(k)} \tag{7.16b}$$

Ví dụ (theo [1]): Xét mạng LGU bốn đầu vào x_1, x_2, x_3, x_4 và một đầu ra $\tilde{y}$ với ba mẫu học $\{\underline{x}^{(k)}\ y^{(k)}\}$, k=1, 2, 3 như sau

$$\underline{x}^{(1)} = \begin{pmatrix} 1 \\ -2 \\ 0 \\ -1 \end{pmatrix}, \qquad \underline{x}^{(2)} = \begin{pmatrix} 0 \\ 1{,}5 \\ -0{,}5 \\ -1 \end{pmatrix}, \qquad \underline{x}^{(3)} = \begin{pmatrix} -1 \\ 1 \\ 0{,}5 \\ -1 \end{pmatrix},$$

$$y^{(1)} = -1, \qquad y^{(2)} = -1, \qquad y^{(3)} = 1$$

Mạng sử dụng

$$\tilde{y} = \frac{2}{1+e^{-\underline{w}^T\underline{x}}} - 1 = \frac{2}{1+e^{-c}} - 1 = \alpha(c) \qquad \Rightarrow \qquad \frac{d\alpha(c)}{dc} = \frac{1}{2}\left[1-\alpha^2(c)\right] = \frac{1}{2}(1-\tilde{y}^2)$$

Bắt đầu từ điểm xuất phát

$$\underline{w}^{(1)} = (w_1 \quad w_2 \quad w_3 \quad w_4)^T = (1 \quad -1 \quad 0 \quad 0{,}5)^T$$

và với bước học $s = 0{,}1$ ta được

1) k=1: $c^{(1)} = \left(\underline{w}^{(1)}\right)^T \underline{x}^{(1)} = 2{,}5$

$$\tilde{y}^{(1)} = \alpha[c^{(1)}] = 0{,}848 \qquad \text{và} \qquad \left.\frac{d\alpha}{dc}\right|_{c^{(1)}} = \frac{1}{2}\left[1-\left(\tilde{y}^{(1)}\right)^2\right] = 0{,}14$$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(2)} = \underline{\tilde{w}}^{(1)} + s\left[y^{(1)} - \tilde{y}^{(1)}\right]\left.\frac{d\alpha}{dc}\right|_{c^{(1)}} \underline{x}^{(1)} = (0{,}974 \quad -0{,}948 \quad 0 \quad 0{,}526)^T$$

2) k=2: $c^{(2)} = \left(\underline{w}^{(2)}\right)^T \underline{x}^{(2)} = -1{,}948$

$$\tilde{y}^{(2)} = \alpha[c^{(2)}] = -0{,}75 \qquad \text{và} \qquad \left.\frac{d\alpha}{dc}\right|_{c^{(2)}} = \frac{1}{2}\left[1-\left(\tilde{y}^{(2)}\right)^2\right] = 0{,}218$$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(3)} = \underline{\tilde{w}}^{(2)} + s\left[y^{(2)} - \tilde{y}^{(2)}\right]\left.\frac{d\alpha}{dc}\right|_{c^{(2)}} \underline{x}^{(2)} = (0{,}974 \quad -0{,}956 \quad 0{,}002 \quad 0{,}531)^T$$

3) k=3: $c^{(3)} = \left(\underline{w}^{(3)}\right)^T \underline{x}^{(3)} = -2{,}46$

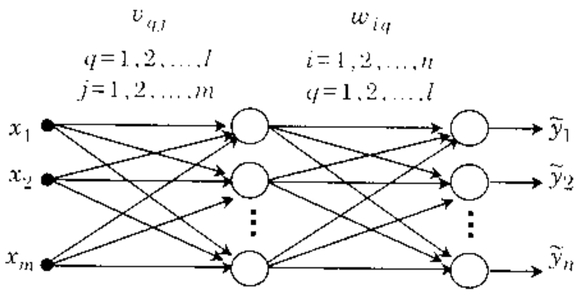
$$\tilde{y}^{(3)} = \alpha[c^{(3)}] = -0{,}842 \qquad \text{và} \qquad \left.\frac{d\alpha}{dc}\right|_{c^{(3)}} = \frac{1}{2}\left[1-\left(\tilde{y}^{(3)}\right)^2\right] = 0{,}145$$

$$\Rightarrow \qquad \underline{\tilde{w}}^{(4)} = \underline{\tilde{w}}^{(3)} + s\left[y^{(3)} - \tilde{y}^{(3)}\right]\left.\frac{d\alpha}{dc}\right|_{c^{(3)}} \underline{x}^{(3)} = (0{,}974 \quad -0{,}929 \quad 0{,}016 \quad 0{,}505)^T$$

7.3 Mạng MLP truyền thẳng

Có rất nhiều mẫu học $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, k=1, 2, ... mà mạng một lớp không thể có được tri thức thông qua việc học các mẫu đó, tức là thuật toán xác định vector tham số không hội tụ. Người ta phải nghĩ tới mạng nhiều lớp (MLP) và đặc biệt mạng MLP này phải có quan hệ phi tuyến không chỉ nằm riêng ở lớp nơ-ron đầu ra.

Mạng MLP có cấu trúc đơn giản nhất là mạng truyền thẳng. *Hình 7.9* mô tả một mạng nơ-ron hai lớp truyền thẳng với m đầu vào x_j, j=1,2, ... , m và n đầu ra $\tilde{y}_i$, i=1,2, ... , n.



Hình 7.9 : Mạng nơ-ron hai lớp truyền thẳng

Lớp nơ-ron thứ nhất có l nơ-ron với l đầu ra $z_1, z_2, \ldots, z_l$

$$c_q = \sum_{j=1}^{m} v_{qj} x_j \qquad \Rightarrow \qquad z_q = \alpha(c_q) = \alpha\left(\sum_{j=1}^{m} v_{qj} x_j\right)$$

Lớp nơ-ron thứ hai có n nơ-ron với n đầu ra $\widetilde{y}_1, \widetilde{y}_2, \ldots, \widetilde{y}_n$

$$c_i = \sum_{q=1}^{l} w_{iq} z_q \qquad \Rightarrow \qquad \widetilde{y}_i = \alpha(c_i) = \alpha\left(\sum_{q=1}^{l} w_{iq} z_q\right).$$

Suy ra

$$\widetilde{y}_i = \alpha\left(\sum_{q=1}^{l} w_{iq} \alpha\left(\sum_{j=1}^{m} v_{qj} x_j\right)\right). \tag{7.17}$$

Nhiệm vụ của việc dạy học cho mạng là xác định các trọng số w_{iq}, v_{qj} từ mẫu học $\{\underline{x}^{(k)}, \underline{y}^{(k)}\}$, k=1, 2, ..., sao cho sai lệch

$$E = \frac{1}{2}\sum_{(k)}\sum_{i=1}^{n}\left[y_i^{(k)} - \widetilde{y}_i^{(k)}\right]^2 \tag{7.18}$$

là nhỏ nhất (học có hướng dẫn – *supervised learning*).

Nguyên tắc xác định w_{iq}, v_{qj} theo hướng ngược gradient của E đã được trình bày tại mục 7.1.4. Gọi $w_{iq}^{(c\tilde{u})}, v_{qj}^{(c\tilde{u})}$ là giá trị các trọng số đã có. Những giá trị này sẽ được hiệu chỉnh bằng một đại lượng gia tăng Δw_{iq}, Δv_{qj} để có

$$w_{iq}^{(m\acute{o}i)} = w_{iq}^{(c\tilde{u})} + \Delta w_{iq} \tag{7.19}$$

$$v_{qj}^{(mới)} = v_{qj}^{(cũ)} + \Delta v_{qj} \tag{7.20}$$

trong đó Δw_{iq}, Δv_{qj} được tính theo hướng ngược gradient

$$\Delta w_{iq} = -s\frac{\partial E}{\partial w_{iq}} = -s\frac{\partial E}{\partial \widetilde{y}_i}\frac{\partial \widetilde{y}_i}{\partial c_i}\frac{\partial c_i}{\partial w_{iq}} = s\sum_{(k)}\left[y_i^{(k)} - \widetilde{y}_i^{(k)}\right]\frac{d\alpha}{dc_i}\bigg|_{c_i^{(k)}} z_q \tag{7.21}$$

$$\Delta v_{qj} = -s\frac{\partial E}{\partial v_{qj}} = -s\frac{\partial E}{\partial c_q}\frac{\partial c_q}{\partial v_{qj}} = s\sum_{(k)}\sum_{i=1}^{n}\left[\left(y_i^{(k)} - \widetilde{y}_i^{(k)}\right)\frac{d\alpha}{dc_i} w_{iq}^{(cũ)}\right]\frac{d\alpha}{dc_q}\bigg|_{c_q^{(k)}} x_j \tag{7.22}$$

Tuy nhiên phương pháp trên lại đòi hỏi phải có ngay một lúc toàn bộ mẫu học mà điều này là không thể. Một phương pháp khác được ứng dụng nhiều trong thực tế để tìm kiếm w_{iq}, v_{qj} là thuật toán lan truyền ngược (***back propagation***) sẽ được trình bày sau đây.

7.3.1 Thuật toán lan truyền ngược

Lan truyền ngược là thuật toán được dùng phổ thông nhất trong việc dạy mạng nhiều lớp, kể cả mạng với nơ-ron có tính động học BSB.

Từ một mẫu học cụ thể $\underline{x}^{(k)}$, $\underline{y}^{(k)}$ và các trọng số đã có của mạng, chẳng hạn như $\underline{w}^{(k)}$, $\underline{v}^{(k)}$ ở mạng hai lớp, người ta xác định đầu ra thực $\underline{\widetilde{y}}^{(k)}$. Sau đó trên cơ sở so sánh với mẫu học $\underline{y}^{(k)}$, các trọng số của lớp nơ-ron đầu ra, ví dụ $\underline{w}^{(k)}$, được hiệu chỉnh thành $\underline{w}^{(k+1)}$. Tiếp tục, từ trọng số mới $\underline{w}^{(k+1)}$ người ta lại hiệu chỉnh trọng số của các nơ-ron thuộc lớp phía trước, ví dụ như $\underline{v}^{(k)}$ thành $\underline{v}^{(k+1)}$. Cứ như vậy cho đến trọng số của lớp nơ-ron đầu vào.

Để phần giải thích chi tiết thuật toán lan truyền ngược được đơn giản, sau đây ta sẽ lấy mạng hai lớp ở *hình 7.9* làm ví dụ.

Với sai lệch cho riêng mẫu học thứ *k* là $\underline{y}^{(k)} - \underline{\widetilde{y}}^{(k)}$, giá trị gia tăng $\Delta w_{iq}^{(k)}$ được xác định theo công thức cải biên của Widnow từ (7.21) như sau

$$\Delta w_{iq}^{(k)} = s\left[y_i^{(k)} - \widetilde{y}_i^{(k)}\right]\frac{d\alpha}{dc_i}\bigg|_{c_i^{(k)}} z_q = s\,\delta_{oi} z_q, \tag{7.23a}$$

trong đó hằng số

$$\delta_{oi} = \left[y_i^{(k)} - \tilde{y}_i^{(k)}\right]\frac{d\alpha}{dc_i}\bigg|_{c_i^{(k)}} \tag{7.23b}$$

có tên gọi tín hiệu sai lệch của nơ-ron đầu ra thứ i. Rõ ràng $\Delta w_{iq}^{(k)}$ phụ thuộc vào z_q. Để tính z_q ta sử dụng các trọng số cũ hiện có của mạng là $\underline{v}^{(k)}$ như sau:

$$z_q = \alpha(c_q^{(k)}) = \alpha\left(\sum_{j=1}^{m} v_{qj}^{(k)} x_j^{(k)}\right) \tag{7.23c}$$

Cùng với $\Delta w_{iq}^{(k)}$, trọng số cũ $w_{iq}^{(k)}$ được hiệu chỉnh thành

$$w_{iq}^{(k+1)} = w_{iq}^{(k)} + \Delta w_{iq}^{(k)} \tag{7.23d}$$

Sau khi đã có $w_{iq}^{(k+1)}$, ta xác định giá trị gia tăng $\Delta v_{qj}^{(k)}$ cho trọng số cũ $v_{qj}^{(k)}$ của nơ-ron thuộc lớp đầu vào nhờ công thức (7.22) nhưng đã được cải biên theo tư tưởng Widnow thành

$$\Delta v_{qj}^{(k)} = s\sum_{i=1}^{n}\left[\left(y_i^{(k)} - \tilde{y}_i^{(k)}\right)\frac{d\alpha}{dc_i} w_{iq}^{(k+1)}\right]\frac{d\alpha}{dc_q} x_j = s\,\delta_{hq} x_j, \tag{7.24a}$$

trong đó

$$\delta_{hq} = \sum_{i=1}^{n}\left[\left(y_i^{(k)} - \tilde{y}_i^{(k)}\right)\frac{d\alpha}{dc_i} w_{iq}^{(k+1)}\right]\frac{d\alpha}{dc_q}\bigg|_{c_q^{(k)}} = \frac{d\alpha}{dc_q}\bigg|_{c_q^{(k)}} \sum_{i=1}^{n} \delta_{oi} w_{iq}^{(k+1)} . \tag{7.24b}$$

Từ $\Delta v_{qj}^{(k)}$ ta được

$$v_{qj}^{(k+1)} = v_{qj}^{(k)} + \Delta v_{qj}^{(k)} . \tag{7.24c}$$

Ví dụ: Xét mạng hai lớp như ở *hình 7.10* với hai nơ-ron ở lớp đầu vào và một nơ-ron ở lớp đầu ra. Các nơ-ron trong mạng là nơ-ron Fermi

$$y = \alpha(c) = \frac{1}{1+e^{-c}} \quad \Rightarrow \quad \frac{d\alpha}{dc} = y(1-y) .$$

Giả sử hiện tại mạng đang có trọng số:

$v_{qj}^{(k)}$ trong đó $q = 1,2$; $j = 1,2$ và $w_{1q}^{(k)}$, $q = 1,2$

Khi có thêm một mẫu học mới $\underline{x}^{(k)}=\begin{pmatrix} x_1^{(k)} \\ x_2^{(k)} \end{pmatrix}$, $y^{(k)}$ thì trước hết trọng số cũ $w_{1q}^{(k)}$ ở nơ-ron lớp ra sẽ được hiệu chỉnh thành

$$w_{1q}^{(k+1)}=w_{1q}^{(k)}+\Delta w_{1q}^{(k)},$$

trong đó

$$\Delta w_{1q}^{(k)}=s\,\delta_{o1}z_q=s\,\delta_{o1}\underbrace{\alpha\left(\sum_{j=1}^{2}v_{qj}^{(k)}x_j^{(k)}\right)}_{z_q}$$

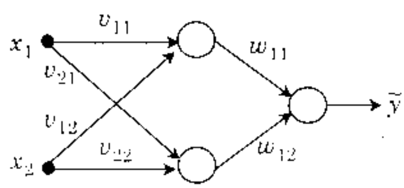
$$\delta_{o1}=\left(y^{(k)}-\tilde{y}^{(k)}\right)\tilde{y}^{(k)}\left(1-\tilde{y}^{(k)}\right) \qquad \text{với} \qquad \tilde{y}^{(k)}=\alpha\left(\sum_{q=1}^{2}w_{1q}^{(k)}z_q\right).$$

Sau khi đã hiệu chỉnh xong lớp đầu ra để có $w_{1q}^{(k+1)}$, trọng số $v_{qj}^{(k)}$ của lớp đầu vào sẽ được sửa đổi thành

$$v_{qj}^{(k+1)}=v_{qj}^{(k)}+\Delta v_{qj}^{(k)}=v_{qj}^{(k)}+s\,\delta_{hq}x_j\,.$$

trong đó

$$\delta_{hq}=z_q(1-z_q)\delta_{o1}w_{1q}^{(k+1)}.$$



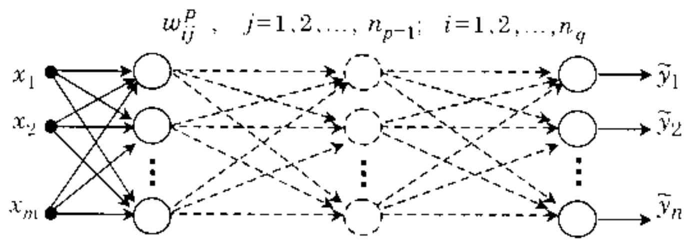
Hình 7.10 : Minh họa cho ví dụ.

Nội dung thuật toán

Bây giờ ta xét tổng quát mạng MLP truyền thẳng với m đầu vào x_j , j =1,2, ... , m và n đầu ra $\tilde{y}_i$, i=1,2, ... , n gồm Q lớp với mỗi lớp có n_p nơ-ron, p =1,2, ... , Q như *hình 7.11* mô tả. Gọi e_i^p , z_i^p , p =1,2, ... , Q là đối số hàm $\alpha(\cdot)$ cũng như đầu ra của nơ-ron thứ i thuộc lớp thứ p trong mạng. Ký hiệu w_{ij}^p, $j=1,2,\dots,n_{p-1}$; $i=1,2,\dots,n_p$ là trọng số chuyển z_j^{p-1} thành z_i^p. Như vậy thì:

$$z_i^p=\alpha(c_i^p)=\alpha\left(\underbrace{\sum_{j=1}^{n_{p-1}} w_{ij}^p z_j^{p-1}}_{c_i^p}\right), \tag{7.25a}$$

$$z_j^0=x_j, \qquad n_0=m, \qquad z_i^Q=\tilde{y}_i \qquad \text{và} \qquad n_Q=n, \tag{7.25b}$$



Hình 7.11 : Mạng MLP truyền thẳng

Giả sử mạng đang có các trọng số $w_{ij}^{p,(k)}$ và một mẫu học $\underline{x}^{(k)}$, $\underline{y}^{(k)}$. Thuật toán lan truyền ngược xác định trọng số mới $w_{ij}^{p,(k+1)}$ cho mạng MLP như trên sẽ gồm:

1) Đặt $z_j^0=x_j$, $n_0=m$ và $n_Q=n$

2) Tính $c_i^{p,(k)}$ và z_i^p lần lượt cho $p=1,2,\dots,Q$ bằng cách ứng với mỗi giá trị p ta lại thực hiện lần lượt cho $i=1,2,\dots,n_p$ hai bước sau:

   a) $c_i^{p,(k)}=\sum_{j=1}^{n_{p-1}} w_{ij}^{p,(k)} z_j^{p-1}$,

   b) $z_i^p=\alpha(c_i^{p,(k)})$

3) Tính $\delta_{Qi}=\left[y_i^{(k)}-z_i^Q\right]\left.\frac{d\alpha}{dc_i^Q}\right|_{c_i^{Q,(k)}}$ với $i=1,2,\dots,n$

4) Thực hiện ngược hướng lần lượt theo $p = Q, Q-1, \ldots, 1$ các bước:

a) $w_{ij}^{p,(k+1)} = w_{ij}^{p,(k)} + s\delta_{pi}\, z_j^{p-1}$ với $j = 1,2, \ldots, n_{p-1}$ và $i = 1,2, \ldots, n_p$

b) Nếu $p>1$ thì tính

$$\delta_{p-1,i} = \left.\frac{d\alpha}{dc_i^{p-1}}\right|_{c_i^{p-1,(k)}} \sum_{j=1}^{n_p} \delta_{pj} w_{ij}^{p,(k+1)} \qquad \text{với} \quad i = 1,2, \ldots, n_{p-1}$$

7.3.2 Hệ số chỉnh hướng học (momentum)

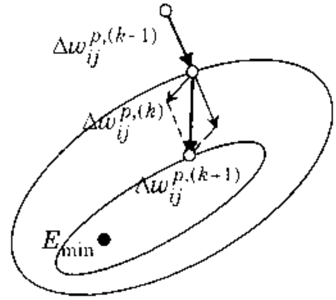
Để tiếp cận đơn giản tới ý tưởng đưa thêm hệ số chỉnh hướng học vào thuật toán ta lại quay về mạng hai lớp đã mô tả ở hình 7.9, cụ thể là hướng học tổng quát (7.20) ngược gradient và hướng học đã được cải biên (7.23a).

Qua so sánh hai hướng học ta thấy rõ ràng hướng học (7.23a) được sử dụng trong thuật toán có một sai lệch đáng kể so với hướng ngược gradient (7.21) do ta đã chỉ giữ lại số hạng cuối cùng và bỏ qua tất các số hạng còn lại trong (7.21).

Bởi vậy, tổng quát cho mạng nhiều lớp truyền thẳng thì để có được một hướng học mới gần sát (7.21) hơn nữa người ta đã sử dụng $\Delta w_{ij}^{p,(k+1)}$ cho nơ-ron thứ i thuộc lớp thứ p theo công thức

$$\begin{aligned}\Delta w_{ij}^{p,(k+1)} &= \mu\, \Delta w_{ij}^{p,(k-1)} + \Delta w_{ij}^{p,(k)} \\ &= \mu\, \Delta w_{ij}^{p,(k-1)} + s\delta_{pi}\, z_j^{p-1},\end{aligned}$$

trong đó $\mu \in [0,1]$ được gọi là hệ số chỉnh hướng học (*hình 7.12*).



Hình 7.12:: Nguyên lý chỉnh hướng học nhờ hệ số momentum.

7.4 Điều khiển mờ và mạng nơ-ron

7.4.1 Ghép nối bộ điều khiển mờ với mạng nơ-ron

Sau khi đã tìm hiểu rõ bản chất cũng như nguyên lý làm việc của các bộ điều khiển mờ và của mạng nơ-ron thì bây giờ đã đến lúc chúng ta có thể thực hiện việc so sánh ưu nhược điểm của chúng trong các ứng dụng điều khiển:

1) Trước hết ta thấy ngay được rằng cả hai, bộ điều khiển mờ và bộ điều khiển sử dụng mạng nơ-ron, về nguyên tắc, đều là những *bộ điều khiển tĩnh, phi tuyến*. Chúng có thể được thiết kế với chất lượng hệ thống cho trước theo một độ chính xác tuỳ ý và làm việc theo nguyên lý tư duy của con người.

2) Tính năng của mạng nơ-ron được quyết định bởi chủng loại nơ-ron sử dụng và cấu trúc mạng ghép nối các nơ-ron đó với nhau. Nó hoàn toàn độc lập với đối tượng được điều khiển. Thậm chí những người thiết kế nếu có kiến thức và hiểu biết về đối tượng thì điều đó cũng không giúp ích gì cho việc lựa chọn nơ-ron và xây dựng cấu trúc mạng. Ngược lại, đối với người thiết kế bộ điều khiển mờ thì những kiến thức, những hiểu biết về đối tượng lại rất cần thiết. Tuy rằng các hiểu biết này không nhất thiết phải được thể hiện dưới dạng mô hình toán học mô tả đối tượng mà có thể chỉ là những kinh nghiệm thu thập được trong quá trình tiếp cận đối tượng, song không có chúng, người thiết kế không thể xây dựng được luật hợp thành, không thể mờ hoá giá trị tín hiệu (xác định hàm thuộc). Thực tế thiết kế bộ điều khiển mờ cho thấy người thiết kế luôn lúng túng nhất khi xây dựng hàm thuộc mô tả giá trị ngôn ngữ. Lý do là trong khi luật hợp thành được xây dựng trên cơ sở chỉ cần có những hiểu biết định tính về đối tượng (vốn là ưu điểm cơ bản của nguyên lý điều khiển mờ) thì việc xác định hàm thuộc cho từng giá trị ngôn ngữ lại đòi hỏi các hiểu biết chi tiết mang tính định lượng (tuy rằng chỉ giới hạn ở giá trị tín hiệu vào–ra của đối tượng) mà điều này lại là điểm mấu chốt để phân biệt điều khiển mờ với điều khiển kinh điển và cũng là điểm cơ bản của mạng nơ-ron.

3) Ngay khi mới được thiết kế, mạng nơ-ron chưa có tri thức. Tri thức của nó được hình thành qua giai đoạn học theo các mẫu học. Mẫu học càng tốt, càng đa dạng và bao nhiêu trường hợp thì tri thức ban đầu của mạng càng gần với thực tế. Song nếu điều đó là chưa đủ thì tri thức của mạng vẫn có thể được bổ sung, và hoàn thiện thêm trong quá trình làm việc với đối tượng. Với bộ điều khiển mờ thì lại hoàn toàn ngược lại. Khi được thiết kế xong, bộ điều khiển mờ đã có ngay một cơ chế làm việc nhất định và cơ chế này sẽ không thay đổi và được giữ cố

định trong suốt thời kỳ làm việc. Nói cách khác mạng nơ-ron có khả năng học còn bộ điều khiển mờ thì không.

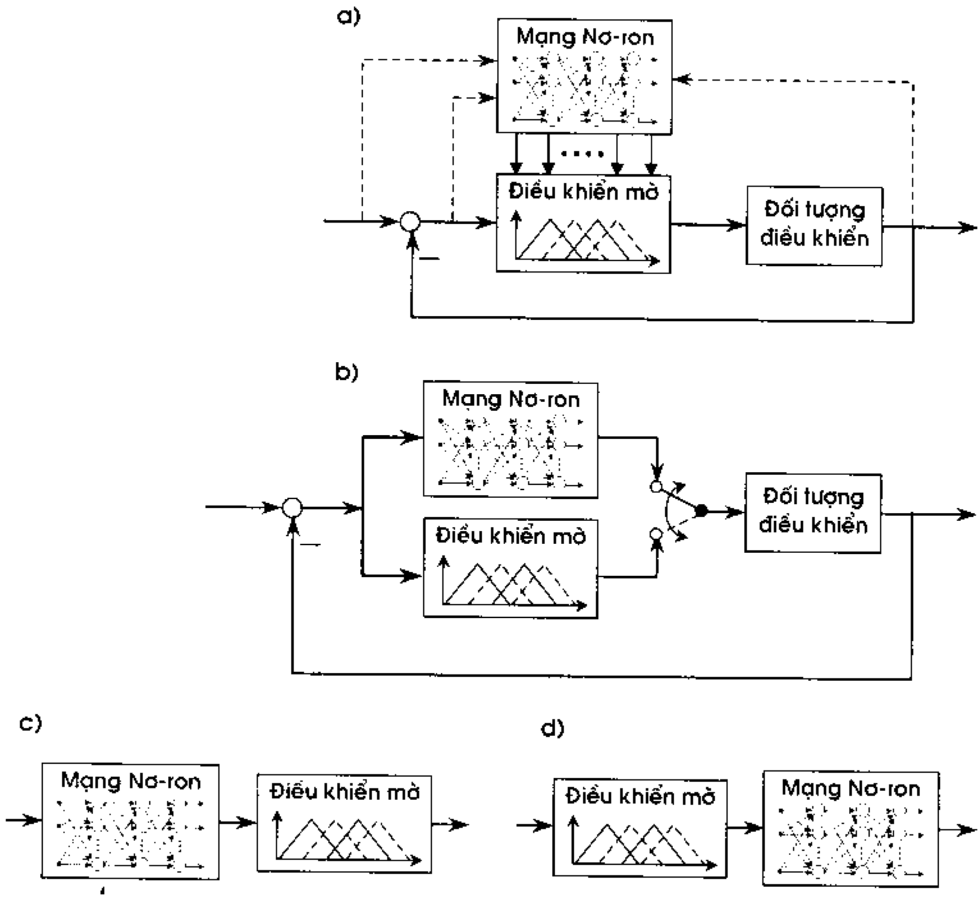
4) Mạng nơ-ron vẫn được xem như giải pháp ứng dụng mà ở đó người thiết kế "mù tịt" về đối tượng (giải pháp black–box). Tri thức của mạng nằm ở các trọng số được rải đều khắp trong mạng. Một thay đổi nhỏ giá trị các trọng số chưa đủ làm thay đổi tính năng của mạng nơ-ron, do đó khó đánh giá được tri thức hiện có của mạng nếu không có mẫu so sánh. Bộ điều khiển mờ thì lại khác, một thay đổi nhỏ ở hàm thuộc, hay luật hợp thành sẽ kéo theo ngay một sự thay đổi tương ứng rất dễ nhận biết về bản chất của nó. Bởi vậy bộ điều khiển mờ có tính thích nghi thời gian thực cao hơn mạng nơ-ron.

Như vậy, rõ ràng những ưu điểm của mạng nơ-ron lại là nhược điểm của bộ điều khiển mờ và ngược lại. Bảng sau trình bày tóm tắt lại ưu nhược điểm của hai bộ điều khiển.

Số TT	Tính chất	Mạng nơ-ron	Bộ điều khiển mờ
1	Thể hiện tri thức (hình thức của tri thức).	Thông qua trọng số, được thể hiện ẩn trong mạng.	Được thể hiện ngay tại luật hợp thành.
2	Nguồn của tri thức.	Từ các mẫu học.	Từ kinh nghiệm chuyên gia.
3	Xử lý thông tin không chắc chắn.	Định lượng.	Định tính và định lượng.
4	Lưu giữ tri thức	Trong nơ-ron và trọng số của từng đường ghép nối nơ-ron	Trong luật hợp thành và hàm thuộc.
5	Khả năng cập nhật và nâng cao tri thức.	Thông qua quá trình học.	Không có.
6	Tính nhạy cảm với các thay đổi của mô hình.	Thấp.	Cao.

Từ mong muốn có được ưu điểm của cả nguyên lý mờ và mạng nơ-ron trong một bộ điều khiển, người ta đã ghép chung bộ điều khiển mờ với mạng nơ-ron thành bộ điều khiển mờ–nơron (và nơron–mờ). *Hình 7.13* mô tả các nguyên lý ghép nối thường gặp.

Việc ghép nối bộ điều khiển mờ với mạng nơ-ron có thể được thực hiện song song hoặc nối tiếp. *Hình 7.13a*) là một ví dụ về hình thức ghép nối song song mà ở đó bộ điều khiển mờ giữ trọng trách điều khiển trực tiếp đối tượng còn mạng nơ-ron không điều khiển trực tiếp song lại có nhiệm vụ theo dõi sự thay đổi của hệ thống để chỉnh định lại tham số cho bộ điều khiển mờ. Nguyên lý ghép nối này đã được ứng dụng thành công nhiều trong thực tế, chẳng hạn như bộ điều khiển là PID mờ có các tham số được chỉnh định theo nguyên lý tối ưu độ lớn và được thực hiện thích nghi bởi mạng nơ-ron RBF (mạng xuyên tâm).



Hình 7.13: Kết hợp điều khiển mờ và mạng nơ-ron

Tất nhiên rằng cấu trúc ghép nối song song ở *hình 7.13a)* chỉ là một ví dụ. Một hình thức ghép nối song song khác là hai bộ điều khiển mờ và nơ-ron làm việc độc lập với nhau cho ở *hình 7.13b)*. Hệ thống sẽ có thêm khâu quyết định sự chuyển đổi công tắc từ mờ sang nơ-ron hoặc ngược lại.

Hình 7.13c) và *7.13d*) minh họa cấu trúc ghép nối nối tiếp. Hình thức ghép nối này rất phù hợp cho các bài toán mà ở đó mạng nơ-ron có nhiệm vụ xử lý các tín hiệu vào -ra cho bộ điều khiển mờ.

7.4.2 Vài nét về lịch sử phát triển

Thời điểm đánh dấu sự ra đời của bộ điều khiển mờ–nơron và nơron–mờ là công trình nghiên cứu của Lee về mối liên quan giữa lý thuyết tập mờ với mạng nơ-ron McCulloch–Pitts vào năm 1970. Phát triển trên nền công trình đó, năm 1971 đã xuất hiện thiết bị tự động với cơ chế suy diễn mờ theo nguyên lý mạng nơ-ron, tuy nhiên vẫn còn ở mức độ thấp.

Thập kỷ 80–90 được xem là thời kỳ nở rộ của các công trình của mờ–nơron cũng như nơron–mờ với những ứng dụng trong nhận dạng ảnh, trong hệ thống hỗ trợ quyết định, trong cơ chế suy diễn nơron–mờ. Nguyên nhân của sự phát triển đó là do sự ra đời của mạng nơ-ron Hopfield, Tank, tiếp nối là sự hoàn thiện thuật toán lan truyền ngược của Rumelhart, Hinton, Williams, Nauck và Kruse cho mạng MLP. Nguyên nhân nữa thúc đẩy sự phát triển này chính là các sản phẩm logic mờ ở Nhật Bản phát triển mạnh mẽ và các chip mờ đã được ứng dụng trong điều khiển máy giặt, nồi cơm điện, máy điều hoà, ...

Hiện nay, sự phát triển của hệ mờ–nơron đang vẫn tiếp tục phát triển mạnh theo hướng tìm tòi và xây dựng các thuật toán học định hướng cho các ứng dụng ở nhiều lĩnh vực khác nhau như hệ thống hỗ trợ quyết định, hệ chuyên gia, tính toán mềm, hệ hỗn loạn, điều khiển thích nghi, xử lý tín hiệu bất định ....

Tài liệu tham khảo

[1] **Chin Teng Lin and George Lee:** Neural Fuzzy Systems. Prentice Hall Inc. 1996.

[2] **Bùi Công Cường và Nguyễn Doãn Phước** (nhóm chủ biên): Hệ mờ, mạng nơ-ron và ứng dụng. Nhà xuất bản Khoa học và Kỹ thuật, 2001.

[3] **Driankov, D.:** An Introduction to Fuzzy Control. SpringerVerlag 1993.

[4] **Dubois, D. and Prade, H.:** Fuzzy Sets and Systems: Theory and Applications. New York: Academic Press, 1980.

[5] **Fossard, A. and Gueguen, C.:** Multivariable System Control. North-Holand Publishing Company, 1979.

[6] **Föllinger, O.** Regelungstechnik.Hüthig Buch Verlag Heidelberg, 1992.

[7] **Hellendoorn, H.:** Fuzzy Control: An Overview. Vieweg Verlag, 1994.

[8] **Kahler, J.** Fuzzy control für Ingenieure. Vieweg Verlag Wiesbaden ,1995.

[9] **Kahlert, J.:** Entwurf, Analyse und Synthese von Fuzzy-Regelungssysteme mit dem Programmsystem WinFACT. 9th Symposium Simulationstechnik ASIM, 1994.

[10] **Klir, G.J. and Yuan, B.:** Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice Hall, 1995

[11] **Kosko, B.:** Neural Networks and Fuzzy Control. Prentice Hall, 1991.

[12] **Kruse, R.; Gebhard, J. and Klawonn, F.:** Foundations of Fuzzy Systems. John Wiley & Sons,1994.

[13] **Lauzi, M.:** Anwendung der Fuzzy-Logic in automatisierungstechnischen Entscheidungs-strukturen VDI Verlag, 1995.

[14] **Nguyễn Doãn Phước & Phan Xuân Minh:** Hệ phi tuyến. Nhà xuất bản Khoa học và Kỹ thuật, 1999.

[15] **Nguyễn Doãn Phước & Phan Xuân Minh:** Điều khiển tối ưu và bền vững. Nhà xuất bản Khoa học và Kỹ thuật, 1999.

[16] **Nguyễn Doãn Phước:** Lý thuyết điều khiển tuyến tính. Nhà xuất bản Khoa học và Kỹ thuật, 2002.

[17] **N.H. Phương, B.C. Cường, N.D. Phước, P.X. Minh và C.V. Hỷ** (nhóm chủ biên): Hệ mờ và ứng dụng. Nhà xuất bản Khoa học và Kỹ thuật, 2000.

[18] **Sontag, E. D.:** Mathematical Control Theory. Spring Verlag New York, 1990.

[19] **Wang, L.X.:** A Course in Fuzzy Systems and Control. Prentice-Hall International, Inc. 1997.

[20] **Wechler, W.:** The Concept of Fuzziness in Automata and Language Theory. Akademie-Verlag, 1978.

[21] **Zimmermann, H.J.:** Fuzzy-Set-Theory and its Applications. Boston, Dordrecht, London: Kluwer Academic Publishers, 1991

ĐẠI HỌC
BÁCH KHOA
1956 - 2006

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

50 NĂM XÂY DỰNG VÀ PHÁT TRIỂN

206266

8 935048 962664

Giá: 33.000đ