

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

---



# ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

NGÀNH CÔNG NGHỆ THÔNG TIN

## GIẢI BÀI TOÁN PHÂN LỚP SỬ DỤNG HỆ LOGIC MỜ LOẠI HAI ĐẠI SỐ GIA TỬ

Sinh viên thực hiện : **Đào Đức Chính**  
Lớp KSTN CNTT - K52  
Giáo viên hướng dẫn : **PGS.TS Trần Đình Khang**

HÀ NỘI 5 - 2012

## PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

### 1. Thông tin về sinh viên

Họ và tên sinh viên: Đào Đức Chính

Điện thoại liên lạc: 01259119542

Email: chinhdao89@gmail.com

Lớp: KSTN Công nghệ thông tin K52

Hệ đào tạo: Chính quy

Đồ án tốt nghiệp được thực hiện tại: Trường đại học Bách Khoa Hà Nội.

Thời gian làm ĐATN: Từ ngày 2 / 1 /2012 đến 30 / 05 /2012

### 2. Mục đích nội dung của ĐATN

Tiếp cận phương pháp hệ logic mờ loại hai đại số gia tử giải bài toán phân lớp tổng quát.

### 3. Các nhiệm vụ cụ thể của ĐATN

- Tìm hiểu kiến thức về logic mờ và hệ logic mờ.
- Nghiên cứu về hệ logic mờ loại hai đại số gia tử.
- Tìm hiểu về bài toán phân lớp.
- Xây dựng mô hình giải bài toán phân lớp sử dụng hệ logic mờ loại hai đại số gia tử.

### 4. Lời cam đoan của sinh viên:

Tôi – **Đào Đức Chính** - cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của **PGS.TS Trần Đình Khang**.

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

*Hà Nội, ngày 30 tháng 05 năm 2012*

Tác giả ĐATN

*Đào Đức Chính*

### 5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

*Hà Nội, ngày 30 tháng 05 năm 2012*

Giáo viên hướng dẫn

*PGS.TS Trần Đình Khang*

## LỜI CẢM ƠN

Trước tiên, tôi xin gửi lời cảm ơn sâu sắc đến thầy giáo, PGS.TS Trần Đình Khang, Viện Công Nghệ Thông Tin và Truyền Thông đã tận tình hướng dẫn và giúp đỡ tôi trong suốt hơn một năm qua. Thầy rất nhiệt tình chỉ bảo và gợi mở tôi tìm tòi và khám phá tri thức mới.

Đồng thời, tôi xin được gửi lời cảm ơn chân thành đến thầy giáo, Ths Phan Anh Phong, giảng viên khoa Công Nghệ Thông Tin, Đại Học Vinh đã nhiệt tình giúp đỡ tôi trong suốt quá trình thực hiện đồ án và cung cấp những tài liệu chuyên môn đáng quý.

Tôi xin được gửi lời cảm ơn chân thành nhất đến tất cả các thầy cô Viện Công Nghệ Thông Tin và Truyền Thông đã truyền đạt những kiến thức quý báu trong suốt quá trình học tập tại trường Đại Học Bách Khoa Hà Nội.

Tôi cũng gửi lời cảm ơn sâu sắc đến tất cả các thầy cô giáo trường Đại Học Bách Khoa Hà Nội đã giảng dạy những kiến thức nền tảng và quan trọng cho tôi và tất cả các bạn sinh viên, những kỹ sư tương lai của đất nước.

Tôi xin gửi lời cảm ơn đến các anh chị sinh viên, các bạn sinh viên trong lớp đã cùng nhau phấn đấu và giúp đỡ nhau trong suốt những năm tháng sinh viên.

Cuối cùng, tôi xin gửi lời cảm ơn sâu sắc nhất đến gia đình tôi, bố mẹ và em gái đã luôn ở bên, động viên, giúp đỡ tôi trong suốt quá trình học tập và trưởng thành.

## TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP

Hiện nay, dữ liệu con người sử dụng và tạo ra trong các quá trình nghiệp vụ tăng lên một cách khổng lồ. Một thách thức đặt ra là tìm kiếm những thông tin có ý nghĩa trong tập dữ liệu khổng lồ đó. Vì vậy, lĩnh vực khai phá dữ liệu đang là một trong những hướng nghiên cứu lớn nhất của lĩnh vực khoa học máy tính và công nghệ tri thức. Khai thác dữ liệu đã và đang ứng dụng thành công vào nhiều lĩnh vực thương mại, tài chính, y học, môi trường, sinh học,...

Bài toán phân lớp là một trong những bài toán cơ bản và quan trọng của lĩnh vực khai phá dữ liệu và có nhiều ứng dụng trong thực tế. Trong nhiều năm trở lại đây, các bài toán phân lớp với nhiều thuộc tính và nhiều lớp thu hút được sự quan tâm của nhiều nhà khoa học trên thế giới. Các kỹ thuật chính được áp dụng giải quyết bài toán phân lớp phân lớp kế thừa từ lĩnh vực cơ sở dữ liệu, học máy, trí tuệ nhân tạo, lý thuyết thông tin, xác suất thống kê và các phương pháp tính toán mềm. Trong đó, một phương pháp tính toán mềm đang được sử dụng rất hiệu quả để giải quyết bài toán phân lớp, đó là sử dụng hệ logic mờ. Hệ logic mờ loại II đại số gia tử là một sự mở rộng của hệ logic mờ loại I truyền thống, với độ thuộc của các tập mờ không phải là số thực mà là các giá trị ngôn ngữ. Với ưu điểm về việc xử lý thông tin không chắc chắn, hệ logic mờ loại II đại số gia tử đang là một tiếp cận mới giải quyết bài toán phân lớp.

Đồ án sẽ trình bày nền tảng lý thuyết hệ logic mờ và chi tiết các bước xây dựng mô hình phân lớp sử dụng hệ logic mờ loại II đại số gia tử. Đầu tiên, các dữ liệu học (training) được chuẩn hoá về một miền xác định. Sau đó, toàn bộ các dữ liệu đã được chuẩn hoá là đầu vào cho một hệ phân lớp mờ loại II đại số gia tử. Cấu trúc của hệ logic mờ loại II đại số gia tử bao gồm hai pha. Công việc ở pha thứ nhất là xây dựng và đánh giá sai số của hệ logic mờ loại I dựa trên giải thuật phân cụm mờ (Fuzzy C - mean Clustering). Trong pha thứ hai, ta xây dựng cấu trúc đại số gia tử đồng thời dựa trên sai số của hệ logic mờ loại I để chuyển từ tập mờ loại I thành tập mờ loại II đại số gia tử. Sau đó, quá trình suy diễn trên tập mờ loại II đại số gia tử đưa ra kết quả phân lớp cho bộ dữ liệu Test.

Chương trình được thử nghiệm trên hai bộ dữ liệu phân lớp chuẩn của UCI Machine Learning Repository là bộ dữ liệu Glass và Ecoli. Hai bộ dữ liệu này đều là bộ dữ liệu nhiều thuộc tính và nhiều lớp. Kết quả thử nghiệm tương đối tốt so với phương pháp hệ logic mờ loại I và một số phương pháp khác. Cuối cùng, đồ án trình bày một số tồn tại và hướng phát triển hệ logic mờ loại II đại số gia tử.

## ABSTRACT OF THESIS

Nowadays, people are working with a giant amount of data, and also creating a huge amount of data, too. The problem is we have to find out the valuable information from that data. Therefore, data mining is one of the largest fields in computer science and knowledge engineering. Data mining is applied in economics, finance, medicine, environment and biology.

Classification is one of the most important problems of data mining. In a few years recently, many scientists are interested in problems classification with many classes and many attributes. Many approaches resolve problem classification, such as artificial intelligent, information theory, soft computing, etc. Fuzzy logic system is a soft computing method has applied to effectively resolve problem classification. This thesis researches a new fuzzy logic system, it is hedge algebraic type-2 fuzzy logic system.

Hedge algebraic type-2 fuzzy logic system (abbreviated as HaT2FLS) is one of the type-2 fuzzy logic system, with membership degree is a linguistic value, not a real number. Therefore, this method approaches inference in linguistic valuables, it is an advantage of this method. HaT2FLS has applied in several forecast problems, and the results are quite good. This thesis exploits the strength of HaT2FLS in inference, and expands from a hedge algebra structure to multi-hedge-algebra-structure, so improve ability to infer with uncertainty information.

The thesis brings up basic theory of fuzzy logic including fuzzy set, type-2 fuzzy set, type-2 fuzzy logic system, and hedge algebraic type-2 fuzzy logic system. After that, the thesis explains whole system resolve problem classification using hedge algebraic type-2 fuzzy logic system, it consists of two subsystems. The object of first subsystem is to construct a type-1 fuzzy logic system from training data. And the object of second subsystem is to convert from type-1 fuzzy logic system to HaT2FLS. The thesis will show clearly each component of the system, and the design of the program, including functional hierarchy diagram, class diagram and some major classes of the program. The datasets are selected from UCI Machine Learning Repository, those are Glass dataset and Ecoli dataset. The end of the thesis shows the experiment results, some conclusions and advanced development.

## NỘI DUNG ĐỒ ÁN

**Chương 1:** Giới thiệu tổng quan về bài toán phân lớp và hướng nghiên cứu của đề tài.

**Chương 2:** Trình bày nền tảng cơ sở lý thuyết liên quan tới đồ án như tập mờ loại I, tập mờ loại II, đại số gia tử, hệ logic mờ loại II đại số gia tử.

**Chương 3:** Trình bày giải pháp đề xuất của đồ án, mô hình giải quyết bài toán phân lớp sử dụng hệ logic mờ loại II đại số gia tử, cùng các bước chi tiết trong mô hình.

**Chương 4:** Trình bày chi tiết cài đặt chương trình giải bài toán phân lớp sử dụng hệ logic mờ loại II đại số gia tử, bao gồm phân tích thiết kế, môi trường cài đặt và tổng quan giao diện chương trình.

**Chương 5:** Trình bày kết quả thử nghiệm với hai bộ dữ liệu chuẩn Glass và Ecoli, qua đó đưa ra đánh giá kết quả giải pháp.

**Chương 6:** Đưa ra kết luận và hướng phát triển của đồ án.

# MỤC LỤC

<b>Phần .....</b>	<b>Trang</b>
PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP .....	2
LỜI CẢM ƠN .....	3
TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP .....	4
ABSTRACT OF THESIS .....	5
NỘI DUNG ĐỒ ÁN.....	6
MỤC LỤC .....	7
CHÚ GIẢI THUẬT NGỮ .....	9
DANH SÁCH HÌNH.....	10
DANH SÁCH BẢNG.....	11
<b>CHƯƠNG 1. GIỚI THIỆU VỀ BÀI TOÁN PHÂN LỚP .....</b>	<b>12</b>
1.1. Bài toán phân lớp .....	12
1.2. Các hướng tiếp cận giải bài toán phân lớp.....	13
1.2.1. Các kỹ thuật phổ biến.....	13
1.2.2. Tiếp cận hệ logic mờ loại II đại số gia tử .....	13
<b>CHƯƠNG 2. CƠ SỞ LÝ THUYẾT.....</b>	<b>14</b>
2.1. Giới thiệu về tập mờ.....	14
2.1.1. Tập mờ loại một .....	14
2.1.2. Tập mờ loại hai .....	17
2.2. Giới thiệu về tập mờ loại II đại số gia tử.....	20
2.2.1. Đại số gia tử.....	20
2.2.2. Tập mờ loại hai đại số gia tử .....	23
2.2.3. Các phép toán tập hợp.....	25
2.3. Hệ logic mờ loại II đại số gia tử.....	27
2.3.1. Giới thiệu về hệ logic mờ loại II đại số gia tử.....	27
2.3.2. Mờ hóa.....	28
2.3.3. Cơ sở luật .....	29
2.3.4. Mô tơ suy diễn.....	29
2.3.5. Bộ phận xử lý đầu ra .....	29
2.4. Kết luận.....	30

<b>CHƯƠNG 3. MÔ HÌNH GIẢI BÀI TOÁN PHÂN LỚP SỬ DỤNG HỆ LOGIC MỜ LOẠI II ĐẠI SỐ GIA TỬ</b> .....	31
3.1. Sơ đồ khối mô hình giải bài toán phân lớp sử dụng HaT2FLS .....	31
3.2. Các quá trình trong pha 1 .....	33
3.2.1. Tiền xử lý dữ liệu .....	33
3.2.2. Thuật toán phân cụm Fuzzy C-mean Clustering.....	34
3.2.3. Xây dựng các tập mờ tam giác.....	37
3.2.4. Xác định các luật.....	38
3.2.5. Suy diễn trên hệ logic mờ loại I .....	39
3.2.6. Đánh giá sai số hệ logic mờ loại I.....	41
3.2.7. Tổng kết pha 1 .....	42
3.3. Các quá trình trong pha 2 .....	42
3.3.1. Xây dựng cấu trúc đại số gia tử.....	42
3.3.2. Xây dựng tập mờ loại II đại số gia tử.....	45
3.3.3. Suy diễn trên hệ logic mờ loại II đại số gia tử.....	49
3.3.4. Tổng kết pha 2 .....	51
<b>CHƯƠNG 4. CHƯƠNG TRÌNH GIẢI BÀI TOÁN PHÂN LỚP</b> .....	52
4.1. Bộ dữ liệu thử nghiệm .....	52
4.2. Phân tích thiết kế ứng dụng .....	52
4.2.1. Biểu đồ phân cấp chức năng.....	52
4.2.2. Biểu đồ lớp của chương trình.....	54
4.2.3. Một số lớp chính của hệ thống .....	56
4.3. Môi trường và công cụ phát triển .....	59
4.4. Chương trình .....	60
4.4.1. Giao diện chính.....	60
4.4.2. Hệ thống tab chức năng của chương trình.....	60
4.5. Kết luận.....	63
<b>CHƯƠNG 5. ĐÁNH GIÁ KẾT QUẢ VÀ BÀN LUẬN</b> .....	64
5.1. Dữ liệu và các kịch bản thử nghiệm .....	64
5.2. Các kết quả thử nghiệm.....	64
5.3. Đánh giá kết quả.....	66
<b>CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b> .....	67
6.1. Kết luận.....	67
6.2. Hướng phát triển trong tương lai .....	68
<b>TỔNG KẾT</b> .....	69
<b>TÀI LIỆU THAM KHẢO</b> .....	70



## CHÚ GIẢI THUẬT NGỮ

THUẬT NGỮ	Ý NGHĨA
T1FS – Type 1 Fuzzy Set	Tập mờ loại một
T2FS – Type 2 Fuzzy Set	Tập mờ loại hai
IT2FS – Interval Type 2 Fuzzy Set	Tập mờ loại hai khoảng
HA - Hedge Algebra	Đại số gia tử
HaT2FS – Hedge Algebraic Type 2 Fuzzy Set	Tập mờ loại hai đại số gia tử
T1FLS – Type 1 Fuzzy Logic System	Hệ logic mờ loại một
T2FLS – Type 2 Fuzzy Logic System	Hệ logic mờ loại hai
HaT2FLS – Hedge Algebraic Type 2 Fuzzy Logic System	Hệ logic mờ loại hai đại số gia tử
Meet	Phép toán meet hai giá trị ngôn ngữ
Join	Phép toán join hai giá trị ngôn ngữ
Training Data	Dữ liệu huấn luyện
Testing Data	Dữ liệu kiểm tra
Glass - Glass Dataset	Bộ dữ liệu Glass
Ecoli - Ecoli Dataset	Bộ dữ liệu Ecoli
FCM – Fuzzy c-means	Thuật giải phân cụm mờ

## DANH SÁCH HÌNH

Hình.....	Trang
Hình 1. Tập mờ TuổiGià được biểu diễn là một đồ thị có dạng hình thang .....	15
Hình 2. Biểu diễn hàm thuộc của tập mờ A và B .....	16
Hình 3. Biểu diễn hàm thuộc của $A \cup B$ (hình a), $A \cap B$ (hình b), $\bar{A}$ (hình c) và $\bar{B}$ (hình c).....	16
Hình 4. Ví dụ về hàm thuộc loại II.....	18
Hình 5. Độ đo tính mờ $fm(True)$ .....	21
Hình 6. Cây đại số gia tử với nút gốc là True.....	23
Hình 7: Các thành phần tổng quát của một hệ logic mờ.....	28
Hình 8: Các thành phần của một hệ logic mờ loại II đại số gia tử .....	28
Hình 9. Sơ đồ khối của hệ thống HaT2FLS giải bài toán phân lớp.....	31
Hình 10: Ví dụ phân cụm mờ.....	34
Hình 11: Biểu đồ các giá trị tâm cụm của bộ dữ liệu Glass .....	36
Hình 12. Xây dựng tập mờ loại I từ các tâm cụm.....	37
Hình 13: Xây dựng tập mờ tam giác cho thuộc tính thứ nhất của bộ dữ liệu Glass .....	37
Hình 14. Tính độ thuộc của một giá trị thuộc tính đối với tập mờ .....	39
Hình 15: Ví dụ về sai số trong hệ logic mờ loại I.....	41
Hình 16. $fm(c-)$ chia khoảng $[0,1]$ thành hai miền tương ứng với giá trị ngôn ngữ phân tử sinh là True và False .....	45
Hình 17. Với $fm(c-) = 0.5$ , khi đó giá trị 0.42 có thể xem là Less False .....	46
Hình 18. Với $fm(c-) = 0.3$ , khi đó giá trị 0.42 có thể xem là Less True .....	46
Hình 19. Giá trị $fm(c-)$ phân cách True và False đối với mọi thuộc tính.....	46
Hình 20. Đa cấu trúc đại số gia tử đối với các thuộc tính. ....	47
Hình 21: Luật đúng và luật sai.....	48
Hình 22. Độ thuộc trên tập mờ của luật đúng (bên trái) nhỏ hơn độ thuộc trên tập mờ của luật sai (bên phải).....	48
Hình 23: Với tập mờ loại I độ thuộc là một số thực rõ (hình a), với tập mờ loại II đại số gia tử là một giá trị ngôn ngữ (hình b).....	50
Hình 14. Biểu đồ phân cấp chức năng hệ thống .....	53
Hình 25. Biểu đồ lớp của hệ thống.....	55
Hình 26. Giao diện chính.....	60
Hình 27. Giao diện tab problem.....	61
Hình 28: Giao diện tab Training .....	62
Hình 29. Giao diện tab Test.....	63

## DANH SÁCH BẢNG

**Bảng ..... Trang**

Bảng 1. Ví dụ về hàm SIG, với bốn gia tử là Very, More, Possibly, Less và phần tử sinh $c = \{\text{True}, \text{False}\}$ .....	21
Bảng 2: Bài toán phân lớp thời tiết mưa và không mưa.....	32
Bảng 3: Kết quả phân cụm bài toán phân lớp Glass .....	36
Bảng 4. Kết quả thử nghiệm bộ dữ liệu Glass .....	65
Bảng 5. Kết quả thử nghiệm bộ dữ liệu Ecoli .....	65

# CHƯƠNG I

## GIỚI THIỆU VỀ BÀI TOÁN PHÂN LỚP

### 1.1. Bài toán phân lớp

Bài toán phân lớp hay phân lớp dữ liệu là một dạng của phân tích dữ liệu nhằm trích rút ra một mô hình mô tả các lớp dữ liệu quan trọng hay dự đoán xu hướng dữ liệu tương lai. Phân lớp dự đoán giá trị của một nhãn xác định rời rạc (được gọi là nhãn lớp) dựa trên tập các giá trị của phần dữ liệu còn lại (được gọi là các thuộc tính).

Bài toán phân lớp dữ liệu gồm hai bước xử lý chính sau đây:

**Bước 1:** Học (training), mục đích của bước này là xây dựng một mô hình xác định các lớp dữ liệu. Mô hình này được xây dựng bằng cách phân tích các bộ dữ liệu của một cơ sở dữ liệu, mỗi bộ dữ liệu được xác định bởi giá trị của các thuộc tính. Trong bước này mỗi bộ dữ liệu đã thuộc về một trong các lớp đã được định nghĩa trước. Trong ngữ cảnh của bài toán phân lớp, mỗi bộ dữ liệu được xem như là một mẫu, hay một bản ghi. Những bộ dữ liệu được phân tích để xây dựng mô hình phân lớp hay hàm phân lớp được lấy từ trong tập dữ liệu học hay dữ liệu huấn luyện (training data set). Những bộ dữ liệu riêng lẻ tạo thành tập dữ liệu huấn luyện còn gọi là những mẫu huấn luyện (training samples) và được chọn ngẫu nhiên từ một kho các mẫu. Bước này được xem là học có giám sát, ngược lại với học có giám sát là học không có giám sát (unsupervised learning), tiêu biểu là bài toán gom cụm (clustering) trong đó các lớp mà các mẫu huấn luyện thuộc về là không biết trước và số lớp dữ liệu cũng không được biết trước.

**Bước 2:** Phân lớp, kiểm tra và đánh giá, bước này sử dụng mô hình phân lớp đã được xây dựng ở bước 1 vào việc phân lớp. Sử dụng một tập các bản ghi khác với bản ghi học để kiểm tra độ chính xác của mô hình phân lớp học được, gọi là tập kiểm thử (test data set). Thông thường, tập dữ liệu ban đầu được chia thành 2 tập không giao nhau: training data set (để học hàm phân lớp) và test data set (để kiểm thử hàm phân lớp học được)

## 1.2. Các hướng tiếp cận giải bài toán phân lớp

### 1.2.1. Các kỹ thuật phổ biến

Các kỹ thuật chính được áp dụng giải bài toán phân lớp phần lớn được kế thừa từ các lĩnh vực cơ sở dữ liệu, học máy, trí tuệ nhân tạo, lý thuyết thông tin, xác suất thống kê và các phương pháp tính toán mềm. Một vài kỹ thuật phân lớp đã được sử dụng trên thế giới:

- Phân lớp cây quyết định (Decision tree classification)
- Bộ phân lớp Bayesian (Bayesian classifier)
- Mô hình phân lớp K-hàng xóm gần nhất (K-nearest neighbor classifier)
- Mạng nơron
- Phân tích thống kê
- Các thuật toán di truyền
- Phương pháp tập thô (Rough set Approach)

### 1.2.2. Tiếp cận hệ logic mờ loại II đại số gia tử

Một phương pháp tính toán mềm đang được nghiên cứu và sử dụng rất hiệu quả để giải bài toán phân lớp, đó là sử dụng hệ logic mờ. Tuy nhiên, hệ logic mờ loại I truyền thống đã bộc lộ những khuyết điểm trong việc xử lý thông tin không chắc chắn, nguyên nhân chủ yếu đó là độ thuộc của tập mờ là các số thực rõ, và quá trình suy diễn là hoàn toàn rõ. Hệ logic mờ loại II đại số gia tử là một sự mở rộng của hệ logic mờ loại I, với độ thuộc của các tập mờ không phải là số thực mà là các giá trị ngôn ngữ. Với sự mở rộng độ thuộc là các giá trị ngôn ngữ trong cấu trúc đại số gia tử đối xứng tuyến tính và đầy đủ, hệ logic mờ loại II đại số gia tử thể hiện sự phù hợp trong xử lý thông tin không chắc chắn và đã được áp dụng giải quyết bài toán dự báo, thu được kết quả tương đối khả quan. Đồ án nghiên cứu sâu thêm quá trình suy diễn trong hệ logic mờ loại II đại số gia tử, từ đó đề xuất kỹ thuật đa cấu trúc đại số gia tử cải tiến một cách hiệu quả hệ logic mờ loại II trong việc giải quyết bài toán phân lớp. Chương trình được thử nghiệm trên hai bộ dữ liệu nổi tiếng của UCI Machine Learning Repository là bộ dữ liệu Glass và Ecoli.

## CHƯƠNG 2

# CƠ SỞ LÝ THUYẾT

### 2.1. Giới thiệu về tập mờ

#### 2.1.1. Tập mờ loại một

Lý thuyết tập mờ lần đầu tiên được Lotfi.A.Zadeh, một giáo sư thuộc trường Đại học California, Berkley, giới thiệu trong một công trình nghiên cứu vào năm 1965. Lý thuyết tập mờ bao gồm logic mờ, số học mờ, quy hoạch toán học mờ, hình học tô pô mờ, lý thuyết đồ thị mờ, và phân tích dữ liệu mờ, mặc dù thuật ngữ logic mờ thường được dùng chung cho tất cả.

Xét một tập hợp toán học thông thường, ví dụ tập hợp Tuổi Già bao gồm các giá trị tuổi  $x$  thỏa mãn:

$$\text{Tuổi Già} = \{x \mid x \geq 60 \text{ và } x \leq 130\}$$

Nhận xét rằng thật vô lý khi ta coi 60 tuổi là già và cũng coi 90 tuổi là già. Với tập hợp toán học thông thường coi tất cả các phần tử trong một tập hợp vai trò như nhau.

Từ đó mở rộng tập mờ toán học thông thường, với vai trò các phần tử trong tập hợp được đặc trưng bởi độ thuộc của phần tử đó, khi đó ta có khái niệm tập mờ (hay tập mờ loại I).

**Định nghĩa 2-1:** Một tập mờ  $A$  xác định trên không gian nền  $X$  được đặc trưng bằng một hàm thuộc  $\mu_A(x) : X \rightarrow [0,1]$

Trong đó,  $A$  là nhãn mờ của biến  $x$ , thường mang một ý nghĩa ngôn ngữ nào đó, mô tả định tính thuộc tính của đối tượng, chẳng hạn như *cao*, *thấp*, *nóng*, *lạnh*, *sáng*, *tối*,...

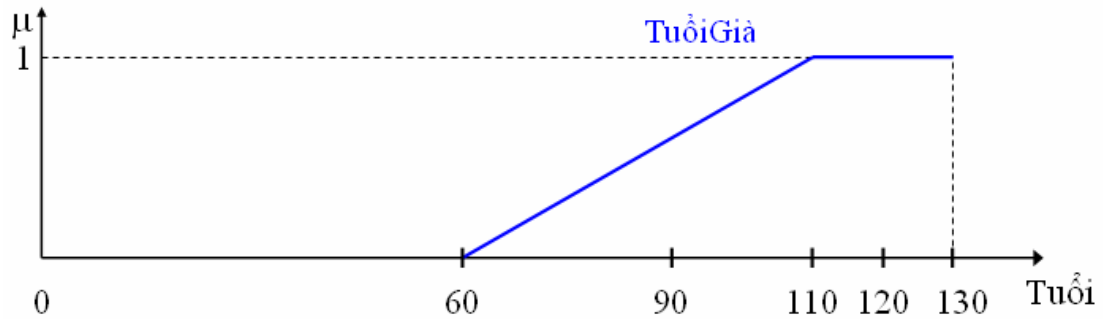
Một tập mờ có thể hữu hạn hoặc vô hạn phần tử, nếu tập mờ  $A$  là hữu hạn thì được viết dưới dạng:  $A = \sum_{i=1}^N \mu_A(x_i)/x_i$  và khi tập mờ  $A$  vô hạn được viết là:

$$A = \int_{x \in X} \mu_A(x)/x$$

Ví dụ tập mờ Tuổi Già trên có thể được biểu diễn như sau:

$$\text{Tuổi Già} = \left\{ \frac{0.01}{60} + \frac{0.03}{61} + \dots + \frac{0.99}{109} + \frac{1}{110} + \dots + \frac{1}{130} \right\}$$

Biểu diễn về mặt hình học ta có tập mờ Tuổi Già như sau:



Hình 1. Tập mờ Tuổi Già được biểu diễn là một đồ thị có dạng hình thang.

Về mặt logic, *tập mờ* diễn đạt *mức độ chân lý* của một phát biểu, với 0.0 đại diện cho trường hợp phát biểu *hoàn toàn sai* và 1.0 biểu diễn trạng thái *hoàn toàn đúng*. Chẳng hạn, khi ta nói:

“Ông Khôi đã già”.

Nếu như ông Khôi đang ở tuổi 75, chúng ta có thể gán cho phát biểu trên một giá trị chân lý là 0.31. Điều này có ý nghĩa là ông Khôi ít nhiều đã già, và mức độ già của ông được đánh giá bằng một con số tương ứng là 0.31.



### Các phép toán tập hợp trên tập mờ loại I

Trong lý thuyết tập mờ, các phép toán tập hợp được định nghĩa thông qua các hàm thuộc của chúng.

Giả sử A và B là hai tập mờ xác định trên không gian X được đặc trưng bởi các hàm thuộc tương ứng là  $\mu_A(x)$  và  $\mu_B(x)$

#### Định nghĩa 2-2:

Hợp của hai tập mờ A và B, ký hiệu  $A \cup B$  có hàm thuộc được định nghĩa:

$$\mu_{A \cup B}(x) = \mu_A(x) \bullet \mu_B(x) \quad (2-1)$$

Trong đó ký hiệu  $\bullet$  là toán tử s-conorm, và thường được sử dụng là phép *max*.

#### Định nghĩa 2-3:

Giao của hai tập mờ A và B, ký hiệu  $A \cap B$  có hàm thuộc được định nghĩa:

$$\mu_{A \cap B}(x) = \mu_A(x) * \mu_B(x) \quad (2-2)$$

Trong đó ký hiệu  $*$  là toán tử t-norm, và thường được sử dụng là phép *min*.

#### Định nghĩa 2-4:

Phản bù của tập mờ A, ký hiệu  $\bar{A}$  có hàm thuộc được định nghĩa:

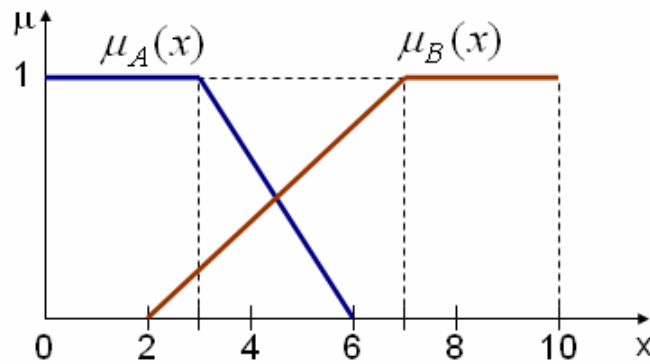
$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (2-3)$$

Xét ví dụ sau:

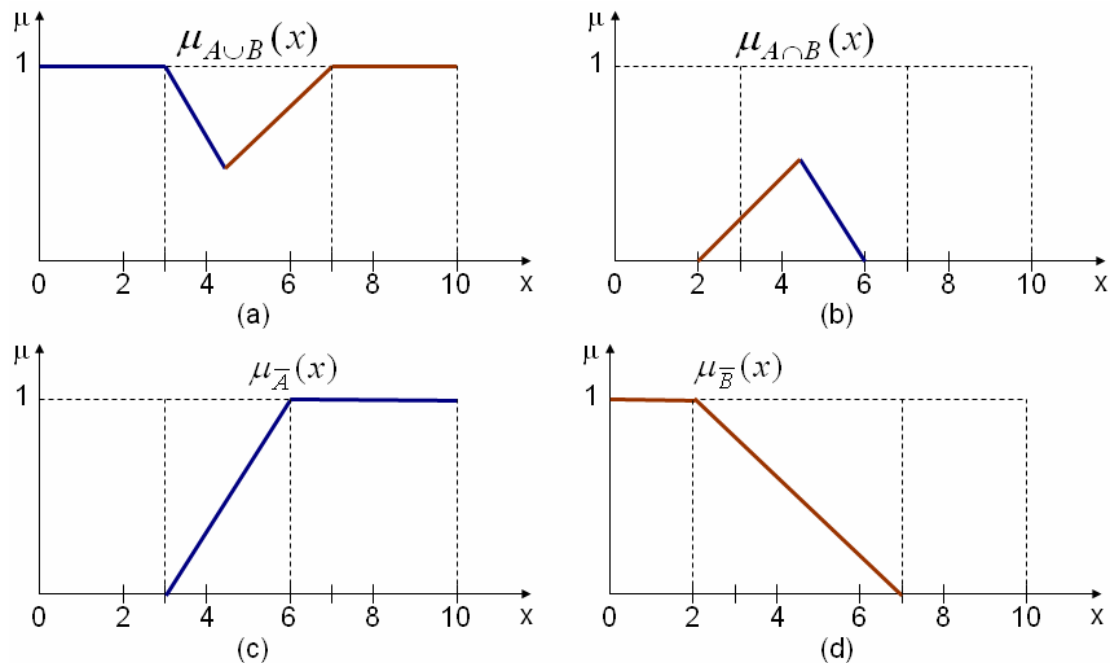
**Ví dụ 2-1:** Cho hai tập mờ A và B cùng xác định trên không gian nền  $X = [0..10]$  có hàm thuộc được xác định như sau:

$$\mu_A(x) = \begin{cases} 1, & \text{khi } 0 \leq x \leq 3 \\ -\frac{1}{3} \times x + 2, & \text{khi } 3 \leq x \leq 6 \\ 0, & \text{khi } 6 \leq x \leq 10 \end{cases} \quad (2-4)$$

$$\mu_B(x) = \begin{cases} 0, & \text{khi } 0 \leq x \leq 2 \\ \frac{1}{5} \times x - \frac{2}{5}, & \text{khi } 2 \leq x \leq 7 \\ 1, & \text{khi } 7 \leq x \leq 10 \end{cases} \quad (2-5)$$



Hình 2. Biểu diễn hàm thuộc của tập mờ A và B



Hình 3. Biểu diễn hàm thuộc của  $A \cup B$  (hình a),  $A \cap B$  (hình b),  $\bar{A}$  (hình c) và  $\bar{B}$  (hình d)



## 2.1.2. Tập mờ loại hai

Tập mờ loại I đã bộc lộ một vài khuyết điểm, theo G.J.Klir [3] “một vấn đề, nếu không nói là nghịch lý, rằng việc biểu diễn tính mờ lại sử dụng độ thuộc mà bản thân nó lại là một số thực rõ”. Hơn nữa, quá trình suy diễn đối với tập mờ loại I là hoàn toàn rõ ở tất cả các công đoạn. Những khuyết điểm này của tập mờ loại I thúc đẩy quá trình nghiên cứu mở rộng tập mờ loại I sao cho vẫn giữ được những ưu điểm trong suy luận không chắc chắn và loại bỏ khuyết điểm của tập mờ loại I. Từ đó, mở rộng tập mờ loại I với độ thuộc được mờ hoá, hay độ thuộc lại là một tập mờ loại I, ta có tập mờ loại II.

**Định nghĩa 2-5:** Một tập mờ loại II được xác định bởi một hàm thuộc mờ, độ thuộc của mỗi phần tử là một tập mờ trên  $[0,1]$ . Một tập mờ loại II  $\tilde{A}$  trên  $X$  là tập mờ được đặc trưng bởi hàm thuộc mờ  $\mu_{\tilde{A}}(x)$  như sau:  $\mu_{\tilde{A}}(x): \tilde{A} \rightarrow [0,1]^J$ , trong đó  $\mu_{\tilde{A}}(x)$  được gọi là độ thuộc mờ và là một tập mờ loại I trên  $J \subseteq [0,1]$ .

Đồ án nghiên cứu một loại tập mờ loại II đặc biệt, khi mà độ thuộc là các giá trị chân lý ngôn ngữ, với quy ước hoàn toàn đúng được coi là 1, hoàn toàn sai được coi là 0, vì vậy giá trị chân lý ngôn ngữ {đúng, sai} có thể coi là một tập mờ loại I trên  $[0,1]$ , và tập mờ loại II với độ thuộc là các giá trị chân lý ngôn ngữ được gọi là tập mờ loại II đại số gia tử.

Để minh họa cho tập mờ loại I và tập mờ loại II ta xét ví dụ sau:

*Ví dụ:* Một hội đồng giám khảo bảo vệ tốt nghiệp gồm có 5 thầy giáo. Điểm bảo vệ tốt nghiệp của một sinh viên là trung bình cộng điểm của cả 5 thầy. Với một thí sinh A, xét về các tiêu chí khác nhau có thể xứng đáng được 8, 9 hoặc 10 điểm. Ta xét một vài trường hợp cho điểm như sau:

Cách 1 (thông thường): 8 hoặc 9 hoặc 10.

Cách 2 (tập mờ loại I):  $\frac{0.2}{8} + \frac{0.4}{9} + \frac{0.4}{10}$ .

Cách 3 (tập mờ loại II khoảng):  $\frac{[0.1,0.3]}{8} + \frac{[0.3,0.4]}{9} + \frac{[0.4,0.5]}{10}$

Cách 4 (tập mờ loại II đại số gia tử):  $\frac{LessTrue}{8} + \frac{VeryTrue}{9} + \frac{MoreTrue}{10}$

*Nhận xét:* Theo cách 1, rõ ràng sai số cho điểm là khá lớn, phụ thuộc vào tâm lý cân nhắc của thầy giáo. Theo cách 2, chính xác hơn một chút, tuy nhiên nhiều trọng số, dễ gây ra sai số mới của chính những trọng số này, mệt mỏi cho người chấm. Cách 3 có lẽ càng phức tạp với người chấm. Cách 4, khá nhẹ nhàng, rõ ràng hợp lý nhất và điểm chính xác hơn.

Qua ví dụ này, ta có thể thấy tập mờ loại II đại số gia tử chính là một bước tiếp cận gần hơn nữa với suy nghĩ của con người, phù hợp với các hệ thống cần tham khảo ý kiến của các chuyên gia. Đó chính là ý nghĩa *nhân văn* của tập mờ loại II đại số gia tử.

Nếu coi tập mờ loại I là sự mở rộng của tập hợp toán học thông thường thành không gian 2 chiều, thì tập mờ loại II chính là sự mở rộng của tập mờ loại I từ

không gian 2 chiều thành không gian 3 chiều. Điều này được khẳng định trong định nghĩa dưới đây:

**Định nghĩa 2-6:** Một tập mờ loại hai, ký hiệu  $\tilde{A}$ , được mô tả bởi một hàm thuộc loại hai  $\mu_{\tilde{A}}(x, u)$ , với  $x \in X$  và  $u \in J_x \subseteq [0,1]$ , tức là:

$$\tilde{A} = \{((x, u), \mu_{\tilde{A}}(x, u)): \forall x \in X, \forall u \in J_x \subseteq [0,1]\} \quad (2-6)$$

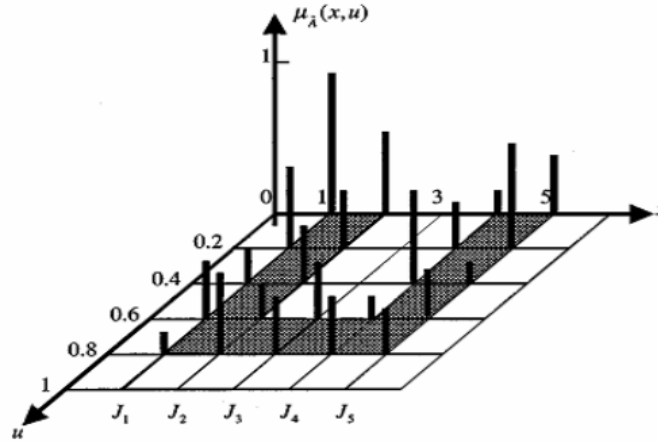
trong đó,  $0 \leq \mu_{\tilde{A}}(x, u) \leq 1$

Tập mờ loại II  $\tilde{A}$  cũng có thể được biểu diễn là:

$$\tilde{A} = \int_{x \in X} \int_{u \in U} \mu_{\tilde{A}}(x, u) / (x, u), J_x \subseteq [0,1] \quad (2-7)$$

ở đây dấu  $\iint$  được hiểu là sự kết hợp tất cả những giá trị có thể của  $x$  và  $u$ . Trong không gian rời rạc, dấu  $\int$  được thay thế bởi  $\sum$ .

Một ví dụ về hàm thuộc loại II cho trong hình vẽ sau:



Hình 4. Ví dụ về hàm thuộc loại II

Hình 4 mô tả một hàm thuộc loại II  $\mu_{\tilde{A}}(x, u)$  với  $x$  và  $u$  rời rạc. Cụ thể,  $X = \{1, 2, 3, 4, 5\}$ ,  $U = \{0, 0.2, 0.4, 0.6, 0.8\}$ ,  $J_x \subseteq [0,1]$  là tập các giá trị có thể của  $u$  tại  $x$ , được gọi là *hàm thuộc sơ cấp* của  $x$ , chẳng hạn  $J_1 = \{0, 0.2, 0.4, 0.6, 0.8\}$ ,  $J_3 = \{0.6, 0.8\}$

Như vậy, việc biểu diễn tập mờ loại II được mở rộng thành 3 chiều, gồm có:

- Không gian nền  $X$
- Độ thuộc sơ cấp  $J_x \subseteq [0,1]$ , cũng chính là giá đỡ của hàm thuộc mờ.
- Độ thuộc thứ cấp  $\mu_{\tilde{A}}(x, u)$ .

**Định nghĩa 2-7:** Sự không chắc chắn trong hàm thuộc sơ cấp của một tập mờ loại II  $\tilde{A}$  là một miền bị giới hạn và được gọi là *chân đế của sự không chắc chắn* - FOU (Footprint Of Uncertainty), đó chính là hợp của các độ thuộc sơ cấp, tức là:

$$\text{FOU}(\tilde{A}) = \bigcup_{x \in X} J_x \quad (2-8)$$

Trong ví dụ hình 4, chân đế của sự không chắc chắn FOU chính là phần được tô đen, là một phần mặt phẳng trong không gian  $(x, u)$

Về mặt ý nghĩa hình học, FOU mô tả trực quan độ không chắc chắn của tập mờ loại hai, nó là biểu diễn hình học toàn bộ miền trị cho tất cả các độ thuộc thứ cấp của một hàm thuộc loại hai. Trong các ứng dụng, FOU là một căn cứ đầu tiên để chúng ta lựa chọn các hàm thuộc loại hai phù hợp.

## 2.2. Giới thiệu về tập mờ loại II đại số gia tử

### 2.2.1. Đại số gia tử

#### 2.2.1.1. Giới thiệu về đại số gia tử

Để có thể nghiên cứu một cách sâu sắc tập mờ loại II đại số gia tử, ta phải đi sâu tìm hiểu cấu trúc đại số gia tử.

Đại số gia tử là một hướng tiếp cận biểu diễn và xử lý các giá trị của một biến ngôn ngữ dựa trên cấu trúc ngữ nghĩa của chúng, với các giả thuyết:

- Mỗi gia tử làm tăng hay giảm nghĩa của các phần tử sinh, ví dụ Very tác động vào True, tạo thành VeryTrue, làm tăng nghĩa của True.
- Mỗi gia tử làm tăng hay giảm nghĩa của các gia tử khác, ví dụ Possible tác động vào VeryTrue, tạo thành PossibleVeryTrue, làm giảm nghĩa của gia tử Very trước đó.
- Khi có gia tử tác động thì giá trị ngôn ngữ mới được tạo ra sẽ “kế thừa” nghĩa của giá trị ngôn ngữ trước đó. Ví dụ, có hai giá trị ngôn ngữ là VeryTrue và MoreTrue. Tác động bao nhiêu gia tử Very vào MoreTrue, thành VeryVery...VeryMoreTrue cũng không “gần” VeryTrue bằng LessVeryTrue được.

**Định nghĩa 2-8:** Đại số gia tử là bộ bốn  $(AX, G, H, \leq)$  với  $AX$  là tập các giá trị,  $G$  là tập các phần tử sinh nguyên thủy,  $H$  là tập các gia tử, và “ $\leq$ ” là quan hệ thứ tự giữa các phần tử của đại số gia tử được cảm sinh bởi ngữ nghĩa của chúng.

Ký hiệu gia tử là  $h$ , phần tử sinh là  $c$ , cụ thể phần tử sinh dương là  $c+$ , phần tử sinh âm là  $c-$ .

Nghiên cứu của đề án chỉ xét đại số gia tử đối xứng, tuyến tính và đầy đủ của biến ngôn ngữ TRUTH, tức là  $G = \{true, false\}$ , và  $H = H+ \cup H-$  trong đó  $H+$  là tập các gia tử dương,  $H-$  là tập các gia tử âm, toàn bộ các gia tử trong  $H$  được sắp thứ tự. Ký hiệu số gia tử dương là  $p$ , số gia tử âm là  $q$ , ta có thể sắp thứ tự các gia tử như sau:  $h_1 < h_2 < \dots < h_p$  trong  $H+$ , và  $h_{-1} < h_{-2} < \dots < h_{-q}$  trong  $H-$ . Khi đó, các phần tử trong cấu trúc đại số gia tử  $AX$  đều được sắp thứ tự. Thứ tự đó được xác định nhờ: (i) Thứ tự giữa các gia tử, (ii) Sự tác động của một gia tử vào các gia tử khác hoặc phần tử sinh.

Quan hệ một gia tử tác động lên một gia tử hay phần tử sinh khác được ký hiệu là  $SIG$ , được định nghĩa:  $SIG: H \times (H \cup G) \rightarrow \{1, -1\}$

$SIG(h, k) = 1$ , nếu gia tử  $h$  làm tăng nghĩa của gia tử  $k$ .

$SIG(h, c) = 1$ , nếu gia tử  $h$  làm tăng nghĩa của phần tử sinh  $c$ .

$SIG(h, k) = -1$ , nếu gia tử  $h$  làm giảm nghĩa của gia tử  $k$ .

$SIG(h, c) = -1$ , nếu gia tử  $h$  làm giảm nghĩa của phần tử sinh  $c$ .

Một ví dụ về hàm  $SIG$  với bốn gia tử Very, More, Possible, Less cho trong bảng dưới đây:

SIG	Very	More	Possibly	Less	c
Very	1	1	-1	1	1
More	1	1	-1	1	1
Possible	-1	-1	1	-1	-1
Less	-1	-1	1	-1	-1

Bảng 1. Ví dụ về hàm SIG, với bốn gia tử là Very, More, Possibly, Less và phần tử sinh  $c = \{\text{True}, \text{False}\}$

Từ quan hệ SIG, dấu của một giá trị ngôn ngữ có thể được xác định theo hàm  $\text{sign}: AX \rightarrow \{1, -1\}$  như sau:

$$(i) \text{sign}(\text{True}) = 1, \text{sign}(\text{False}) = -1$$

$$(ii) \text{sign}(h_n h_{n-1} \dots h_1 c) = \text{SIG}(h_n, h_{n-1}) \times \dots \times \text{SIG}(h_1, c) \times \text{sign}(c)$$

Căn cứ vào dấu của các giá trị ngôn ngữ trước và sau khi tác động một gia tử ta có thể biết quan hệ thứ tự về ngữ nghĩa của chúng.

Một tính chất thể hiện khi gia tử tác động vào một giá trị ngôn ngữ được phát biểu như sau: Với bất kỳ gia tử  $h \in H$  và với mọi phần tử  $\hat{x} \in AX$ , nếu  $\text{sign}(h\hat{x}) = 1$  thì  $h\hat{x} > \hat{x}$ , ngược lại nếu  $\text{sign}(h\hat{x}) = -1$  thì  $h\hat{x} < \hat{x}$ .

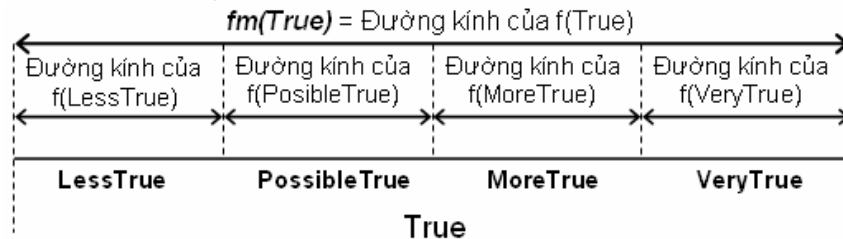
#### 2.2.1.2. Độ đo tính mờ, khoảng tính mờ, ánh xạ định lượng ngữ nghĩa

Trong phần này trên đã trình bày ngữ nghĩa về mặt định tính của các giá trị ngôn ngữ đại số gia tử. Phần này trình bày về mặt định lượng giá trị ngôn ngữ.

Xét một giá trị ngôn ngữ  $\hat{x}$ , ký hiệu  $H(\hat{x})$  là tập tất cả các khái niệm ngữ nghĩa được sinh ra từ  $\hat{x}$  nhờ việc thay đổi ngữ nghĩa của  $\hat{x}$  bằng các gia tử ngôn ngữ. Về mặt ngữ nghĩa, các khái niệm này đều mang nghĩa “gốc” của  $\hat{x}$  và do đó chúng góp phần tạo ra tính mờ của  $\hat{x}$ . Về mặt trực quan, kích cỡ của tập  $H(\hat{x})$  có liên quan đến tính mờ của khái niệm  $\hat{x}$  và có thể xem  $H(\hat{x})$  mô phỏng tính mờ của khái niệm này. Do vậy, xác định độ đo tính mờ của khái niệm  $\hat{x}$  có thể được định nghĩa dựa vào việc xác định kích thước của tập  $H(\hat{x})$ .

Xét một ánh xạ bảo toàn thứ tự  $f: AX \rightarrow [0,1]$ , độ đo tính mờ của  $\hat{x}$ , ký hiệu là  $fm(\hat{x})$ , được định nghĩa là đường kính của tập  $f(H(\hat{x}))$ . Trường hợp đặc biệt  $\hat{x}$  là phần tử sinh  $c$  có  $fm(c)$ .

Ví dụ về  $fm(\text{True})$  trong hình vẽ sau.



Hình 5. Độ đo tính mờ  $fm(\text{True})$

Với giả thuyết rằng tỷ số  $fm(h\hat{x})/fm(\hat{x})$  không phụ thuộc vào phần tử  $\hat{x}$  cụ thể, hay tỷ số này là hằng số với mọi  $\hat{x}$ , chỉ phụ thuộc gia tử  $h$ , do đó có thể được ký hiệu bằng  $\mu(h)$ , được gọi là *độ đo tính mờ của gia tử  $h$* .

Từ đó có thể tính được độ đo tính mờ của mọi giá trị ngôn ngữ trong đại số gia tử theo công thức:

$$fm(h_n h_{n-1} \dots h_1 c) = \mu(h_n) \times \mu(h_{n-1}) \times \dots \times \mu(h_1) \times fm(c)$$

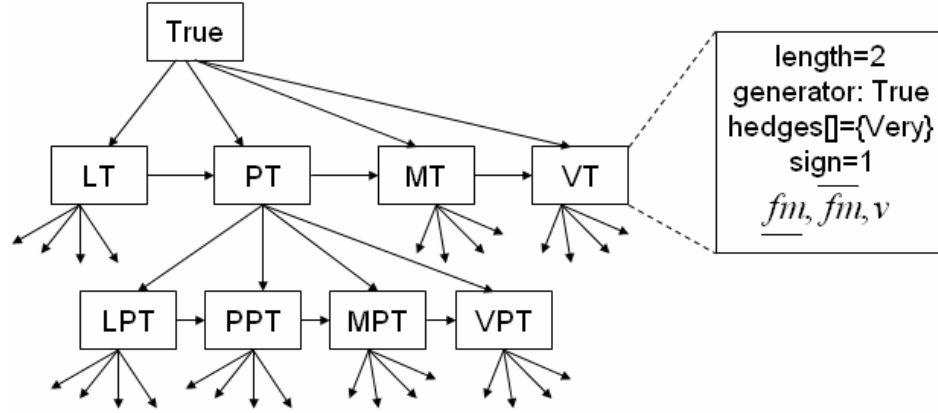
Đặt  $\alpha = \sum_{i=1}^{-q} \mu(h_i)$ ,  $\beta = \sum_{i=1}^p \mu(h_i)$ , với  $fm(c) \in [0,1]$ ,  $fm(h) \in [0,1]$ ,  $fm(c-) + fm(c+) = 1$  và  $\alpha + \beta = 1$  thì  $fm(\hat{x}) \in [0,1]$  với  $\forall \hat{x} \in AX$ .

Trên cơ sở độ đo tính mờ  $fm$  của các giá trị ngôn ngữ, khái niệm khoảng mờ liên kết với  $fm$  được xây dựng. Ký hiệu  $P([0,1])$  là tập tất cả các khoảng con trên đoạn  $[0,1]$ , ánh xạ  $J: AX \rightarrow P([0,1])$  được gọi là phép gán khoảng mờ dựa trên  $fm$  nếu với mỗi  $\hat{x} \in AX$  thì  $J(\hat{x})$  xác định một khoảng mờ  $[\underline{fm}(\hat{x}), \overline{fm}(\hat{x})] \subseteq P([0,1])$  có độ dài đúng bằng  $fm(\hat{x})$ , khi đó  $[\underline{fm}(\hat{x}), \overline{fm}(\hat{x})]$  được gọi là *khoảng tính mờ của  $\hat{x}$* .

Dựa vào khoảng tính mờ và độ đo tính mờ của các gia tử, việc định lượng ngữ nghĩa giá trị ngôn ngữ trong đại số gia tử tuyến tính đầy đủ hoàn toàn được xác định. Ta ký hiệu ánh xạ định lượng ngữ nghĩa này là:  $v: AX \rightarrow [0,1]$ .

Tóm lại, với mỗi giá trị ngôn ngữ  $\hat{x} \in AX$  trong đại số gia tử của biến chân lý ngôn ngữ TRUTH được đặc trưng bởi bộ ba giá trị  $\langle \underline{fm}(\hat{x}), v(\hat{x}), \overline{fm}(\hat{x}) \rangle$ , trong đó  $[\underline{fm}(\hat{x}), \overline{fm}(\hat{x})]$  thể hiện tính mờ còn  $v(\hat{x})$  là định lượng ngữ nghĩa của  $\hat{x}$ . Theo cách nhìn của tập mờ thì mỗi giá trị ngôn ngữ  $\hat{x}$  trong đại số gia tử có thể được biểu diễn ngữ nghĩa bằng một tập mờ với giá đỡ là  $[\underline{fm}(\hat{x}), \overline{fm}(\hat{x})]$  và có độ thuộc bằng 1 tại  $v(\hat{x})$ .

Toàn bộ giá trị ngôn ngữ trong đại số gia tử có thể mô hình hoá bằng cấu trúc cây tổng quát, với hai nút gốc là hai nút ứng với phần tử sinh (True và False). Mỗi một nút tương ứng với một giá trị ngôn ngữ, do đó bao gồm các thành phần: thành phần cấu tạo (độ dài ngôn ngữ, phần tử sinh, thứ tự gia tử và hàm dấu sign) và thành phần định lượng ngữ nghĩa (bộ ba giá trị  $\langle \underline{fm}(\hat{x}), v(\hat{x}), \overline{fm}(\hat{x}) \rangle$ ). Một nút cha có  $n$  nút con, trong đó  $n=p+q$  là tổng số gia tử. Nút con có tính chất kế thừa ngữ nghĩa từ nút cha. Các nút cùng mức có cùng độ dài ngôn ngữ, trong đó nếu cùng cha được gọi là các nút anh em. Một ví dụ về cây đại số gia tử cho dưới đây, trong đó có 4 gia tử là Less, Possibly, More, Very được viết tắt tương ứng là L,P,M,V.



Hình 6. Cây đại số gia tử với nút gốc là True

Ví dụ cây đại số gia tử như trên, với nút gốc là True, phần tử VeryTrue có các thành phần như sau:

Độ dài ngôn ngữ là 2.

Phần tử sinh là True.

Danh sách các gia tử: {Very}

Hàm dấu: có giá trị là 1.

## 2.2.2. Tập mờ loại II đại số gia tử

### 2.2.2.1. Định nghĩa

Xét đại số gia tử tuyến tính, đầy đủ  $(AX, G, H, \leq)$  với tập phần tử sinh  $G = \{\text{true}, \text{false}\}$ , tập các gia tử  $H = H^- \cup H^+$ . Tập mờ loại II đại số gia tử  $\hat{A}$  xác định trên không gian nền  $X$  là tập mờ, trong đó độ thuộc là các giá trị chân lý thuộc cấu trúc đại số gia tử trên, nghĩa là:

$$\hat{A} = \int_X \mu_{\hat{A}}(x) / x \text{ với } \mu_{\hat{A}}(x) \in AX$$

Ví dụ: Cho ĐSGT  $(AX, G, H, \leq)$ , với tập phần tử sinh  $G = \{\text{False}, \text{True}\}$ , tập các gia tử  $H = \{\text{Very}, \text{More}, \text{Possibly}, \text{Less}\}$  và  $U$  là không gian nền bao gồm các giá trị chiều cao của con người tính theo đơn vị lẫn chục cm. Khi đó tập mờ loại hai gia tử  $Tall$  có dạng như sau:

$$Tall = \frac{False}{160} + \frac{LessTrue}{170} + \frac{MoreTrue}{180} + \frac{VeryTrue}{190} + \frac{VeryVeryTrue}{200}$$

trong đó  $False, LessTrue, MoreTrue, VeryVeryTrue$  thuộc  $AX$ ; 160, 170, 180, 190, 200 là các giá trị trên  $U$ .

### 2.2.2.2. Biểu diễn tập mờ loại hai đại số gia tử

Phần này xây dựng cách biểu diễn mới cho tập mờ loại hai ĐSGT. Qua đó, cho phép chúng ta hiểu sâu hơn về dạng tập mờ loại hai này, đồng thời cách biểu diễn này cũng là tiền đề để xây dựng khái niệm trọng tâm của HAT2FS.



### Độ dài giá trị ngôn ngữ

Một giá trị ngôn ngữ trong ĐSGT tuyến tính đầy đủ  $(AX, G, H, \leq)$  có độ dài  $l$  được biểu diễn như sau  $h_{l-1} \dots h_2 h_1 c$ , với  $h_{l-1}, \dots, h_2, h_1$  là các gia tử,  $c \in \{true, false\}$ .

Ví dụ 2.2.2.1: *VeryVeryMoreTrue* là giá trị ngôn ngữ có độ dài  $l=4$ .

▪ **Nhận xét:** Khi giá trị chân lý ngôn ngữ  $\hat{x}$  trong ĐSGT có độ dài đủ lớn thì  $\underline{fm}(\hat{x}) \approx \overline{fm}(\hat{x}) \approx v(\hat{x})$ , trong đó  $\underline{fm}(\hat{x}), \overline{fm}(\hat{x})$  tương ứng là nút trái và nút phải của  $fm(\hat{x})$ . Mệnh đề sau cho thấy rõ hơn về nhận xét này.

▪ **Mệnh đề 1:** Cho một ĐSGT tuyến tính đầy đủ  $(AX, G, H, \leq)$ ;  $H = \{less, possibly, more, very\}$ ;  $G = \{true, false\}$ . Với  $\varepsilon > 0$  bé tùy ý cho trước ta luôn tìm được số nguyên dương  $k$  sao cho độ đo tính mờ của các giá trị chân lý ngôn ngữ có độ dài  $k$  nhỏ hơn  $\varepsilon$ .

### Chứng minh

Xét  $k = 1 + \lceil \log_{\lambda} (\varepsilon / \gamma) \rceil$ , trong đó  $\lambda = \max\{\mu(h_j) : j = 1, 2, 3, 4\}$  và  $\gamma = \max\{fm(c^-), fm(c^+)\}$ . Ta sẽ chứng minh độ đo tính mờ của các giá trị chân lý ngôn ngữ có độ dài  $k$  nhỏ hơn  $\varepsilon$ .

Thật vậy, do  $\hat{x} \in X_k$  nên  $\hat{x}$  được biểu diễn dưới dạng sau  $\hat{x} = h_{k-1} \dots h_1 c$ , trong đó  $c \in \{true, false\}$ . Ta có:

$$fm(\hat{x}) = \mu(h_{k-1})\mu(h_{k-2}) \dots \mu(h_1)fm(c) \leq \lambda^{k-1} \cdot \gamma = \lambda^{1 + \lceil \log_{\lambda} (\varepsilon / \gamma) \rceil - 1} \cdot \gamma \leq \lambda^{\log_{\lambda} (\varepsilon / \gamma)} \cdot \gamma = \varepsilon.$$

Vậy nếu  $\hat{x} \in AX_k$  thì  $fm(\hat{x}) \leq \varepsilon$ .

Trong chứng minh trên  $AX_k$  là kí hiệu tập giá trị chân lý ngôn ngữ có độ dài  $k$ .

### Chân đế của sự không chắc chắn của HA T2FS: $FOU_{HA}(\hat{A})$

Xét một HAT2FS,  $\hat{A} = \int_U \hat{x} / x$ ,  $\hat{x} \in AX$ . Mỗi  $\hat{x} \in AX$  được đặc trưng bởi một khoảng con ngữ nghĩa trên  $[0,1]$  với độ lớn  $fm(\hat{x})$  được xác định theo hàm độ đo tính mờ, ký hiệu khoảng con đó là  $[\underline{fm}_x, \overline{fm}_x]$ . Do đó, với  $\forall x \in U$  thì tập các  $[\underline{fm}_x, \overline{fm}_x]$  sẽ tạo ra một miền kín, và được gọi là chân đế của sự không chắc chắn của tập mờ loại hai ĐSGT  $\hat{A}$ , ký hiệu là  $FOU_{HA}(\hat{A})$ , có nghĩa là:

$$FOU_{HA}(\hat{A}) = \bigcup_{x \in U} [\underline{fm}_x, \overline{fm}_x]$$



### Kế thừa ngữ nghĩa giữa các giá trị chân lý

Cho ĐSGT tuyến tính đầy đủ  $(AX, G, H, \leq)$ ,  $G = \{True, False\}$ ,  $H = \{Very, More, Possibly, Less\} \cup \{Inf, Sup\}$ . Khi đó, với  $\hat{x} \in AX$  và  $h_i \in H$  thì ta gọi  $h_i\hat{x}$  là giá trị chân lý kế thừa ngữ nghĩa từ  $\hat{x}$ , ký hiệu là  $h_i\hat{x} \subseteq \hat{x}$ .

Ví dụ:  $VeryVeryMoreTrue \subseteq VeryMoreTrue \subseteq MoreTrue \subseteq True$

### Kết nhập ngữ nghĩa các giá trị chân lý

Cho ĐSGT tuyến tính đầy đủ  $(AX, H, G, \leq)$ ,  $G = \{True, False\}$ ,  $h_i \in H$ , ta luôn có

$\bigcup_{i=1}^{p+q} h_i\hat{x} = \hat{x}$ , khi đó  $\hat{x} \in AX$  được gọi là giá trị chân lý kết nhập ngữ nghĩa từ các  $h_i\hat{x}$ ,  $i = 1, \dots, p+q$ .

Ví dụ: Cho ĐSGT  $(AX, G, H, \leq)$ ;  $H = \{Less, Possibly, More, Very\}$ ;  $G = \{True, False\}$ . Để ngắn gọn ta ký hiệu  $V, M, L, P$  tương ứng với  $Very, More, Less, Possibly$ ; còn  $T$  và  $F$  là  $True$  và  $False$ . Khi đó ta luôn có:

$$MT = VMT + MMT + PMT + LMT$$

Phép “+” trong ví dụ 2.2.2.4 được hiểu là phép hợp các độ đo tính mờ của giá trị chân lý.

### 2.2.3. Các phép toán tập hợp

Với tập mờ loại I, các phép toán tập hợp được thực hiện trên độ thuộc là số thực của chúng, đó là hai toán tử t-norm và s-conorm tương ứng với phép giao và phép hợp. Từ đó, mở rộng với tập mờ loại II đại số gia tử, các phép toán tập hợp được đưa về thực hiện trên các giá trị ngôn ngữ. Sử dụng nguyên lý mở rộng với t-norm và s-conorm, ta đưa ra phép meet và join các giá trị chân lý ngôn ngữ.

Giả sử  $\hat{A} = \sum_{i=1}^N \mu_{\hat{A}}(x_i)/x_i$  và  $\hat{B} = \sum_{i=1}^N \mu_{\hat{B}}(x_i)/x_i$  là các tập mờ loại II đại số gia tử trên  $X$ . Khi đó, các phép toán tập hợp trên  $\hat{A}$  và  $\hat{B}$  được thực hiện trên các giá trị ngôn ngữ của chúng. Từ đó, ta định nghĩa *phép hợp* tương ứng với phép toán *join* trong đại số gia tử, và *phép giao* tương ứng với phép *meet* trong đại số gia tử.

#### 2.2.3.1. Phép hợp

$\hat{A} \cup \hat{B} = \sum_{i=1}^N (\mu_{\hat{A}}(x_i) \tilde{\vee} \mu_{\hat{B}}(x_i)) / x_i$ , trong đó  $\tilde{\vee}$  được ký hiệu là phép *join* hai giá trị ngôn ngữ.

Thuật toán tính phép *join* hai giá trị ngôn ngữ được trình bày dưới đây:

**Input:**  $H^- = \{h_{-q}, \dots, h_{-1}\}$ ,  $H^+ = \{h_1, h_2, \dots, h_p\}$ ,  $SIG$  relation,  
 $\hat{x} = l_{g-1} \dots l_1 c \in AX_g$ ,  $\hat{y} \in AX_k$ ,  $g \leq k$ .

**Output:**  $\hat{z} = \hat{x} \tilde{\nabla} \hat{y}$

**BEGIN**

Step 1: **if**  $\hat{x}$  and  $\hat{y}$  do not have the semantically inheritance relationship  
**then return**  $\hat{z} = \hat{x} \nabla \hat{y}$   
**end if**

Step 2:

$\hat{z}^g \leftarrow \hat{x}$

**if**  $c = c^-$  **then**  $sig1 \leftarrow -1$  **else**  $sig1 \leftarrow 1$  **end if**

**if**  $g > 1$  **then**  $sig1 \leftarrow sig1 \times SIG(l_1, c)$  **end if**

**for**  $i = 2$  **to**  $g - 1$  **do**  $sig1 \leftarrow sig1 \times SIG(l_i, l_{i-1})$  **end for**

**for**  $i = g$  **to**  $k - 1$  **do**

**if**  $v(\hat{z}^i) \geq v(\hat{y})$  **then**

$\hat{z} \leftarrow \hat{z}^i$ , **break**

**end if**

**if**  $SIG(h_p, l_{i-1}) \times sig1 = 1$  **then**

$\hat{z}^{i+1} \leftarrow h_p \hat{z}^i$ ,  $l_i = h_p$

**else**

$\hat{z}^{i+1} \leftarrow h_{-q} \hat{z}^i$ ,  $sig1 \leftarrow -sig1$ ,  $l_i = h_{-q}$

**end if**

**end for**

**return**  $\hat{z}$

**END**

---

### 2.2.3.2. Phép giao

$\hat{A} \cap \hat{B} = \sum_{i=1}^N (\mu_{\hat{A}}(x_i) \tilde{\Delta} \mu_{\hat{B}}(x_i)) / x_i$ , trong đó  $\tilde{\Delta}$  được ký hiệu là phép *meet* hai giá trị ngôn ngữ.

Thuật toán tính phép *meet* hai giá trị ngôn ngữ được trình bày dưới đây.

---

**Input:**  $H^- = \{h_{-q}, \dots, h_{-1}\}$ ,  $H^+ = \{h_1, h_2, \dots, h_p\}$ , SIG relation,  
 $\hat{x} = l_{g-1} \dots l_1 c \in AX_g$ ,  $\hat{y} \in AX_k$ ,  $g \leq k$ .

**Output:**  $\hat{z} = \hat{x} \tilde{\Delta} \hat{y}$

**BEGIN**

Step 1: **if**  $\hat{x}$  and  $\hat{y}$  do not have the semantically inheritance relationship  
**then return**  $\hat{z} = \hat{x} \Delta \hat{y}$   
**end if**

Step 2:

$\hat{z}^g \leftarrow \hat{x}$

**if**  $c = c^-$  **then**  $sig1 \leftarrow -1$  **else**  $sig1 \leftarrow 1$  **end if**

**if**  $g > 1$  **then**  $sig1 \leftarrow sig1 \times SIG(l_1, c)$  **end if**

**for**  $i = 2$  **to**  $g - 1$  **do**  $sig1 \leftarrow sig1 \times SIG(l_i, l_{i-1})$  **end for**

**for**  $i = g$  **to**  $k - 1$  **do**

**if**  $v(\hat{z}^i) \leq v(\hat{y})$  **then**

$\hat{z} \leftarrow \hat{z}^i$ , **break**

**end if**

**if**  $SIG(h_p, l_{i-1}) \times sig1 = -1$  **then**

$\hat{z}^{i+1} \leftarrow h_p \hat{z}^i$ ,  $sig1 \leftarrow -sig1$ ,  $l_i = h_p$

**else**

$\hat{z}^{i+1} \leftarrow h_{-q} \hat{z}^i$ ,  $l_i = h_{-q}$

**end if**

**end for**

**return**  $\hat{z}$

**END**

---

### 2.2.3.3. Phần bù

$\neg \hat{A} = \sum_{i=1}^N \overline{\mu_{\hat{A}}(x_i)} / x_i$  trong đó  $\overline{\mu_{\hat{A}}(x_i)}$  là phép phủ định giá trị chân lý ngôn ngữ trong đại số gia tử, được thực hiện bằng cách giữ nguyên phần gia tử, thay đổi phần tử sinh True thành False, False thành True.

## 2.3. Hệ logic mờ loại II đại số gia tử

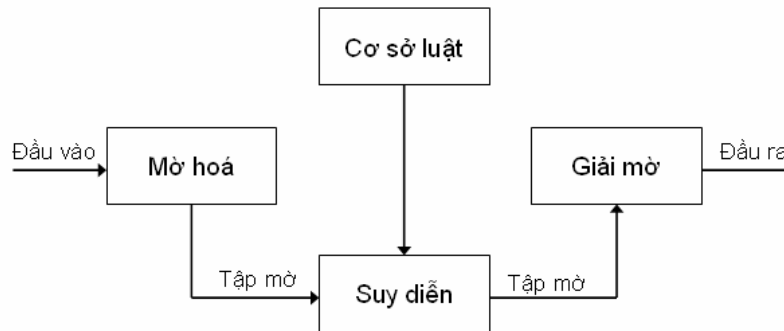
### 2.3.1. Giới thiệu về hệ logic mờ loại II đại số gia tử

Một mô hình hệ thống chính xác có ý nghĩa rất lớn trong thực tiễn và là mục tiêu cơ bản cần hướng tới của mọi quá trình mô hình hoá hệ thống. Tuy nhiên, để đạt được điều này là không đơn giản. Các hệ thống hiện nay, phần nhiều có đầu ra phụ thuộc vào rất nhiều yếu tố, mà trong số đó tồn tại những yếu tố khó có thể định lượng chính xác. Mặt khác, hàm ánh xạ giữa đầu vào - đầu ra rất phức tạp, khó có thể xây dựng tường minh được. Vấn đề mô phỏng chính xác một hệ thống như vậy là một việc không khả thi. Trong khi đó, biểu diễn các yếu tố không đầy đủ, không

rõ ràng, không chắc chắn lại là thế mạnh của logic mờ. Xuất phát từ thực tế đó, người ta đề xuất ra việc sử dụng hệ logic mờ thay thế mô hình vật lý đối với những hệ thống kể trên với một sai số chấp nhận được.

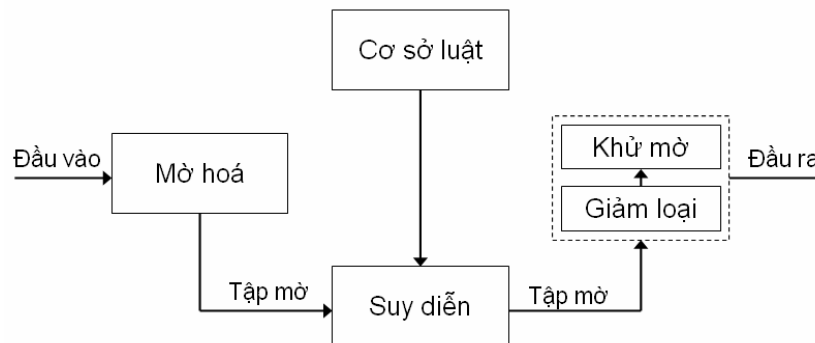
Hiện nay, các hệ logic mờ loại hai đang được ứng dụng trong các lĩnh vực như điều khiển học, xử lý tín hiệu... Các hệ này thường có khối lượng tính toán rất lớn. Trong phần này sẽ trình bày cách thức xây dựng hệ logic mờ loại hai đại số gia tử với ưu điểm là khối lượng tính toán không nhiều và hơn nữa duy trì được mối quan hệ giữa các giá trị chân lý ngôn ngữ.

Một hệ logic mờ bao gồm các thành phần tổng quát như sau:



Hình 7: Các thành phần tổng quát của một hệ logic mờ

Với hệ logic mờ loại I, các thành phần tương đương với hình vẽ trên. Với hệ logic mờ loại II, khối giải mờ bao gồm hai thành phần nhỏ, *giảm loại* thành tập mờ loại I và *khử mờ* thành số thực (hình ?). Quá trình suy diễn được thực hiện trên tập mờ.



Hình 8: Các thành phần của một hệ logic mờ loại II đại số gia tử

Hệ logic mờ loại hai đại số gia tử gồm 4 khối là: mờ hóa, cơ sở luật, mô tơ suy diễn và cuối cùng là khối xử lý đầu ra bao gồm giảm loại và khử mờ. Trong các phần tiếp theo của chương ba, 4 khối này lần lượt được trình bày.

### 2.3.2. Mờ hoá

Khối mờ hóa gia tử nhận đầu vào là các giá trị số thực, đầu ra là tập mờ loại hai đại số gia tử biểu diễn tương ứng với giá trị đầu vào đó. Tùy thuộc vào kiểu mờ hoá mà có dạng tập mờ đầu ra tương ứng, ở mức tổng quát, sau bước mờ hoá là tập mờ loại II.

### 2.3.3. Cơ sở luật

Cơ sở luật là tập các luật IF - THEN, đây là cơ sở tri thức của hệ logic mờ. Nói chung, trong một hệ logic mờ thường có  $p$  đầu vào  $x_1 \in X_1, x_2 \in X_2, \dots, x_p \in X_p$ , một đầu ra  $y \in Y$  và  $M$  luật, trong đó luật thứ  $l$  có dạng:

$$R^l : \text{IF } x_1 \text{ is } \hat{F}_1^l \text{ and...and IF } x_p \text{ is } \hat{F}_p^l \text{ THEN } y \text{ is } \hat{G}^l$$

Trong đó  $\hat{F}_1^l, \hat{F}_2^l, \dots, \hat{F}_p^l, \hat{G}^l$  là các tập mờ loại II đại số gia tử, xác định trên các không gian tương ứng.

Nếu mỗi luật được xem như một phép kéo theo thì luật  $R^l$  trên có thể được biểu diễn như một quan hệ mờ giữa không gian đầu vào  $X = X_1 \times X_2 \times \dots \times X_p$  và không gian đầu ra  $Y$ :

$$R^l : \hat{F}_1^l \times \dots \times \hat{F}_p^l \rightarrow \hat{G}^l = \hat{A}^l \rightarrow \hat{G}^l, \text{ với } \hat{F}_1^l \times \dots \times \hat{F}_p^l = \hat{A}^l$$

Khi đó, luật  $R^l$  được đặc trưng bởi quan hệ mờ  $\hat{R}_c^l$  với hàm thuộc:

$$\begin{aligned} \mu_{R^l}(x, y) &= \mu_{\hat{A}^l}(x) \tilde{\Delta} \mu_{\hat{G}^l}(y) \\ &= \mu_{\hat{F}_1^l}(x_1) \tilde{\Delta} \mu_{\hat{F}_2^l}(x_2) \tilde{\Delta} \dots \tilde{\Delta} \mu_{\hat{F}_p^l}(x_p) \tilde{\Delta} \mu_{\hat{G}^l}(y) \end{aligned}$$

### 2.3.4. Mô tơ suy diễn

Xét các giá trị đầu vào là  $(x'_1, x'_2, \dots, x'_p)$ . Tập mờ loại II đại số gia tử  $\hat{A}_0$  tương ứng là:

$$\mu_{\hat{A}_0}(x') = \mu_{X_1}(x'_1) \tilde{\Delta} \mu_{X_2}(x'_2) \tilde{\Delta} \dots \tilde{\Delta} \mu_{X_p}(x'_p)$$

Từ đó, đầu ra  $\hat{B}_0^l$  tương ứng với đầu vào  $\hat{A}_0$  được tính bằng phép hợp thành giữa tập mờ  $\hat{A}_0$  và quan hệ mờ  $\hat{R}_c^l$ :

$$\begin{aligned} \hat{B}_0^l &= \hat{A}_0 \circ \hat{R}_c^l \text{ hay ta có:} \\ \mu_{\hat{B}_0^l}(y) &= \tilde{\nabla}_x \left( \mu_{\hat{A}_0}(x') \tilde{\Delta} \mu_{\hat{R}_c^l}(x', y) \right) \end{aligned}$$

Cuối cùng, để xác định tập mờ đầu ra, ta đốt cháy  $\hat{A}_0$  bởi tất cả  $M$  luật có trong cơ sở luật, ta sử dụng phép hợp đầu ra của mỗi luật, tức là:

$$\hat{B}_0 = \bigcup_{l=1}^M \hat{B}_0^l, \text{ hay } \mu_{\hat{B}_0}(y) = \tilde{\nabla}_{l=1}^M \left[ \mu_{\hat{B}_0^l}(y) \right]$$

### 2.3.5. Bộ phận xử lý đầu ra

Bộ phận xử lý đầu ra trong hệ logic mờ loại hai đại số gia tử gồm hai phần: giảm loại và khử mờ.

Phép giảm loại tập mờ loại hai đại số gia tử  $\hat{A}$  thành tập mờ loại một  $A$  nhờ vào ánh xạ định lượng ngữ nghĩa  $v$ :  $\mu_A(x) = v_{\hat{A}}(x)$ .

Còn quá trình khử mờ, ta sử dụng phương pháp khử mờ trọng tâm: *Center Of Area*. Giá trị đầu ra của hệ thống được tính theo công thức:

$$y = \frac{\sum_{i=1}^N \mu_i c_i}{\sum_{i=1}^N \mu_i}$$

với  $c_i$  là trọng tâm của cụm đầu ra trong luật thứ  $i$ ,  $\mu_i$  là độ thuộc có được sau khi giảm loại ở trên.

## 2.4. Kết luận

Chương 2 đã trình bày toàn bộ cơ sở lý thuyết được sử dụng trong đồ án, bao gồm tập mờ, đại số gia tử, tập mờ loại II đại số gia tử và hệ logic mờ loại II đại số gia tử.

Một trong những ưu điểm của tập mờ loại hai đại số gia tử là duy trì được mối quan hệ giữa những giá trị chân lý ngôn ngữ. Hơn nữa, khối lượng tính toán của các phép toán hợp, giao và phần bù là không nhiều, điều này rất thích hợp khi sử dụng trong các hệ logic mờ. Hệ logic mờ loại II đại số gia tử thực hiện suy diễn trên tập mờ loại II đại số gia tử, và do đó ưu điểm của hệ logic mờ loại II đại số gia tử là xử lý thông tin không chắc chắn.

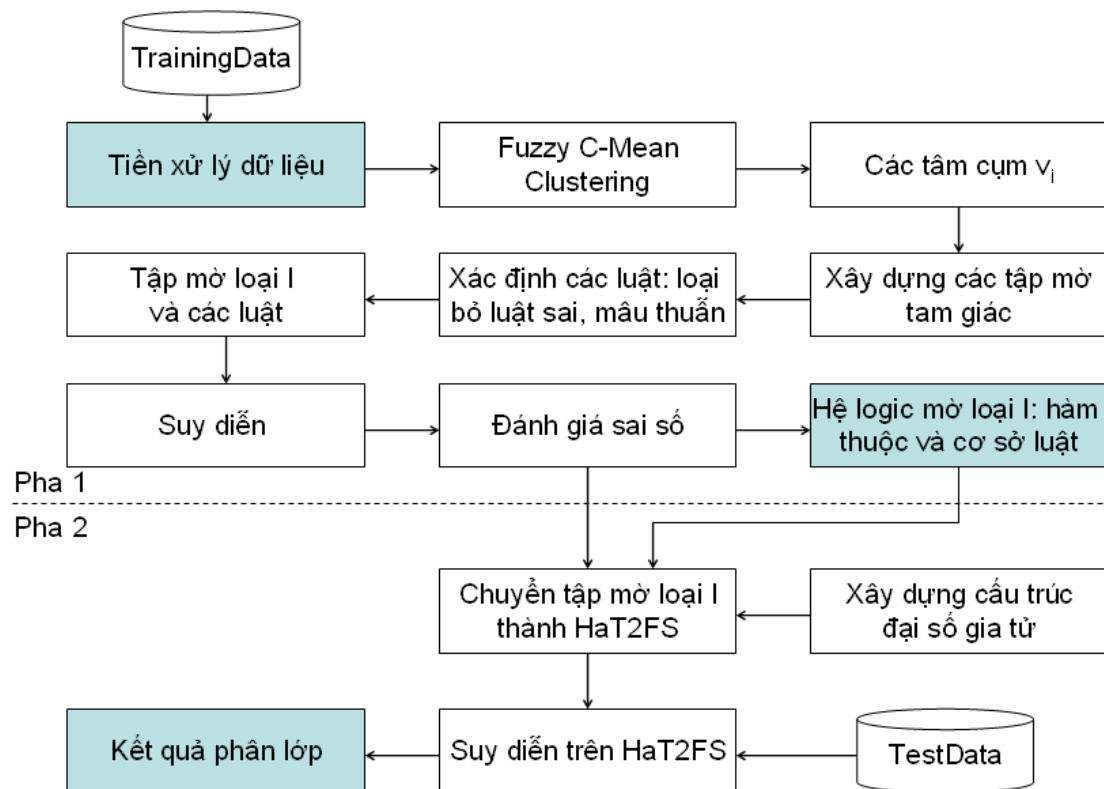
## CHƯƠNG 3

# MÔ HÌNH GIẢI BÀI TOÁN PHÂN LỚP SỬ DỤNG HỆ LOGIC MỜ LOẠI II ĐẠI SỐ GIA TỬ

Trong chương này sẽ trình bày chi tiết các bước giải quyết bài toán phân lớp sử dụng hệ logic mờ loại II đại số gia tử.

### 3.1. Sơ đồ khối của mô hình giải bài toán phân lớp sử dụng HaT2FLS

Mô hình tổng quan của hệ thống phân lớp sử dụng hệ logic mờ loại II đại số gia tử được trình bày trong hình vẽ dưới đây



Hình 9. Sơ đồ khối của hệ thống HaT2FLS giải bài toán phân lớp



Quá trình xây dựng bộ phân lớp được trải qua 2 pha chính:

Pha 1: Toàn bộ dữ liệu Training được tiền xử lý, chuẩn hoá và khởi tạo tham số cho thuật toán phân cụm mờ. Thuật toán phân cụm mờ Fuzzy C-mean Clustering cho kết quả là các tâm cụm để tạo tiền đề xây dựng hệ logic mờ loại I, hàm thuộc và cơ sở luật.

Pha 2: Xây dựng cấu trúc đại số gia tử. Sau đó xây dựng hệ logic mờ loại II đại số gia tử từ hệ logic mờ loại I, sai số của hệ logic mờ loại I và cấu trúc đại số gia tử đã được xây dựng. Sau đó là quá trình nhận dữ liệu Test, thực hiện quá trình suy diễn trên hệ logic mờ loại II đại số gia tử và cuối cùng thu được kết quả phân lớp.

### **Bài toán phân lớp ví dụ:**

Bài toán 1: Phân lớp mưa.

Xét các yếu tố ảnh hưởng đến thời tiết mưa hay không mưa của ngày tiếp theo. Các yếu tố đó có thể là nhiệt độ, độ ẩm, sức gió, lượng mây,... được coi là các thuộc tính. Ta chỉ xét kết quả mưa hay không mưa, vì vậy có thể coi như là 2 lớp, lớp “mưa” và lớp “không mưa”. Các kết quả từ những ngày hôm trước cho ta bộ dữ liệu training, với các yếu tố nhiệt độ, độ ẩm, sức gió, lượng mây và đã được phân lớp “mưa” hoặc “không mưa” như trong bảng sau:

Các thuộc tính					Phân lớp
Nhiệt độ	Độ ẩm	Sức gió	Lượng mây	Yếu tố khác	
31.5°C	80%	52km/h	70	...	Mưa
25°C	60%	80km/h	120	...	Mưa
33°C	50%	50km/h	60	...	Không mưa
...	...	...	...	...	...
<b>30.5°C</b>	<b>75%</b>	<b>60km/h</b>	<b>80</b>	<b>...</b>	<b>?</b>

Bảng 2: Bài toán phân lớp thời tiết mưa và không mưa

Với thông số thời tiết đo được của ngày hôm nay, chính là các giá trị của các thuộc tính, vậy ta có thể kết luận là thuộc lớp “mưa” hay “không mưa”. Rõ ràng đây chính là một ví dụ bài toán phân lớp rất thực tế và thú vị. Tuy nhiên, tập các dữ liệu thực tế do không có điều kiện thu thập, nên bài toán này sẽ chỉ sử dụng những lập luận để mô tả phương pháp.

### **Bài toán 2: Phân lớp Glass (dữ liệu kính)**

Dữ liệu Glass (Glass Dataset) được lấy từ kho dữ liệu chuẩn của UCI Machine Learning Repository. Dữ liệu kính (Glass) được nghiên cứu, thí nghiệm và tạo ra bởi B.German, làm việc tại một tổ chức tư pháp của Mỹ, Central Research



Establishment , USA Forensic Science Service. Nguyên nhân thúc đẩy việc nghiên cứu các thành phần của kính và phân lớp kính xuất phát từ nhu cầu thực tế trong việc phá các vụ án. Trong hầu hết các vụ án, những tấm kính còn sót lại có thể ẩn chứa điều gì đó và sẽ là một bằng chứng chính xác.

Các thuộc tính của dữ liệu Glass là:

1. Id number: 1 to 214
2. RI: refractive index
3. Na: Sodium
4. Mg: Magnesium
5. Al: Aluminum
6. Si: Silicon
7. K: Potassium
8. Ca: Calcium
9. Ba: Barium
10. Fe: Iron

Như vậy, không kể thuộc tính ID thì bộ dữ liệu Glass có 9 thuộc tính. Trong bộ dữ liệu này, kính được phân thành 6 lớp:

1. building\_windows\_float\_processed
2. building\_windows\_non\_float\_processed
3. vehicle\_windows\_float\_processed
4. containers
5. tableware
6. headlamps

Như vậy, dữ liệu Glass có 9 thuộc tính, được phân thành 6 lớp. Có tất cả 214 mẫu. Đồ án tốt nghiệp sử dụng bộ dữ liệu chuẩn Training và Test từ trang web [www.mlcomp.org](http://www.mlcomp.org) để mô tả các kết quả trong quá trình thực hiện. Bộ dữ liệu Training có 150 mẫu và bộ dữ liệu Test có 64 mẫu.

## 3.2. Các quá trình trong pha 1

### 3.2.1. Tiền xử lý dữ liệu

Quá trình này sẽ thực hiện chuẩn hoá miền dữ liệu, đồng thời tính một vài thông số của dữ liệu.

Từ dữ liệu Training đầu vào, giả sử giá trị  $a$  của thuộc tính  $i$  nằm trong dải  $[aMin..aMax]$ , sử dụng phép chuẩn hoá tuyến tính về miền  $[0..100]$ :

$$a \rightarrow \frac{a - aMin}{aMax - aMin} \times 100$$

Sau khi chuẩn hoá dữ liệu sẽ nằm trong dải  $[0..100]$ . Lưu ý là thông thường chuẩn hoá về miền  $[0..1]$ , tuy nhiên khi đó các phép toán thực hiện với số rất bé, do đó phát sinh sai số trong quá trình tính toán trên máy tính. Chuẩn hoá về miền  $[0..100]$  mang tính chất phóng đại dữ liệu so với miền  $[0..1]$ , giảm sai số trong quá trình tính toán trên máy tính.

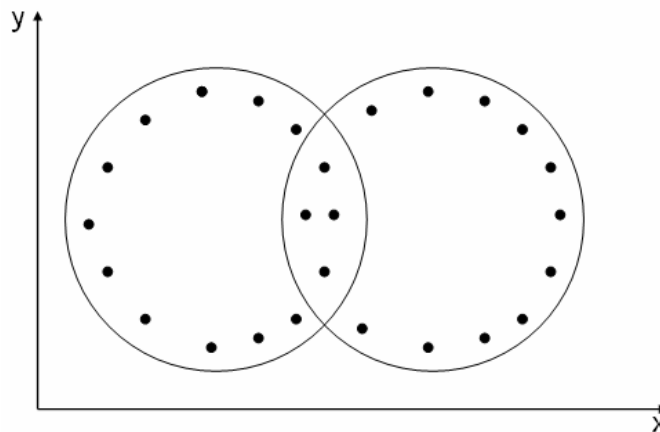
Đối với các thuộc tính có dữ liệu không phải số thực, ta ánh xạ bảo toàn thứ tự các giá trị đó lên miền  $[0..100]$  cách đều nhau. Ví dụ, thuộc tính “học lực” của một học sinh thông thường là “yếu”, “trung bình”, “khá”, “giỏi”, “xuất sắc”, sẽ được ánh xạ sang miền  $[0..100]$  tương ứng là: 0, 25, 50, 75, 100.

Quá trình này cũng thực hiện đếm số giá trị (khác nhau) của một thuộc tính. Ví dụ, với thuộc tính “học lực” trên ta có số giá trị khác nhau là 5. Khi đó, mặc dù bảng dữ liệu Training có rất nhiều mẫu, tuy nhiên với mỗi thuộc tính ta ghi lại số giá trị khác nhau này, kí hiệu với thuộc tính  $i$  là  $a_i$ .

### 3.2.2. Thuật toán phân cụm Fuzzy C-mean Clustering

Thuật toán Fuzzy C-mean Clustering (FCM) là một phương pháp phân cụm dữ liệu trong đó mỗi một phần tử dữ liệu có thể thuộc một hoặc nhiều cụm với các độ thuộc khác nhau.

Một ví dụ minh họa về phân cụm mờ được biểu diễn trong hình dưới đây, trong đó các phần tử nằm ở giữa có thể coi thuộc cả 2 cụm với độ thuộc là 0.5, các phần tử nằm ngoài cùng bên trái hoặc ngoài cùng bên phải chỉ thuộc một cụm với độ thuộc là 1.



Hình 10: Ví dụ phân cụm mờ.

Thuật toán này dựa trên việc tối thiểu hoá hàm mục tiêu:

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2$$

trong đó có  $N$  là số phần tử,  $C$  là số cụm,  $u_{ij}$  là độ thuộc của phần tử  $x_i$  trong cụm  $j$ ,  $x_i$  là phần tử dữ liệu thứ  $i$  và  $c_j$  là tâm cụm thứ  $j$ , kí hiệu  $\|.\|$  là hàm khoảng cách giữa hai phần tử, thường được sử dụng là hàm khoảng cách Ole.

Tham số  $m$  là một số thực bất kỳ lớn hơn 1, và thường được thiết lập bằng 2.0 theo Bezdek.[2]

Ta thực hiện phân cụm với từng thuộc tính riêng rẽ, khi đó thuật toán phân cụm được thực hiện trên dữ liệu một chiều, vì vậy hàm khoảng cách là hàm trị tuyệt đối.

**Thuật toán Fuzzy C-mean Clustering:**

**Input:** - Dữ liệu đã được chuẩn hoá về miền [0..100]  
- Số cụm đối với mỗi thuộc tính p là  $C_p$ ,  $i = 1, 2, \dots, n$  với n là số thuộc tính.

**Output:** - n tập hợp, mỗi tập hợp p có  $C_p$  tâm cụm đối với mỗi thuộc tính p  
 $\{c_1, c_2, \dots, c_{C1}\}, \{c_1, c_2, \dots, c_{C2}\}, \dots \{c_1, c_2, \dots, c_{Cn}\}$

**Bước duyệt toàn bộ n thuộc tính:**

**For p =1 to n do**

**Begin**

**Bước 1:** Khởi tạo  $C_p$  giá trị tâm cụm của thuộc tính p

**Bước 2:** Tính giá trị độ thuộc  $u_{ij}(k)$ :

$$u_{ij} = \frac{1}{\sum_{k=1}^C \left( \frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

**Bước 3:** Tính lại giá trị tâm cụm  $c_i(k)$

$$c_i = \frac{\sum_{j=1}^N u_{ij}^m . x_j}{\sum_{j=1}^N u_{ij}^m}$$

**Bước 4:** Kiểm tra  $|c_i(k) - c_i(k-1)| \leq \varepsilon$  ( $\varepsilon = 0.0001$ ) thì dừng  
Ngược lại quay lại bước 2.

**End For All properties**

---

Thuật toán phân cụm mờ cho kết quả các tâm cụm khá tốt và thời gian hội tụ rất nhanh. Tuy nhiên, đầu vào của thuật toán phải chỉ rõ dữ liệu được phân thành mấy cụm. Vì vậy, phương pháp hay được sử dụng là thuật giải di truyền kết hợp với thuật toán phân cụm để tối ưu số cụm sao cho hàm mục tiêu đạt giá trị nhỏ nhất. Tuy nhiên, thuật toán di truyền có một khuyết điểm khá lớn là thời gian chạy tương đối lâu. Vì vậy, đề xuất của đồ án phương pháp chọn số cụm với dữ liệu như sau:

Với từng thuộc tính, nếu số giá trị khác nhau  $a_i < 5$

Chọn số cụm bằng  $a_i$

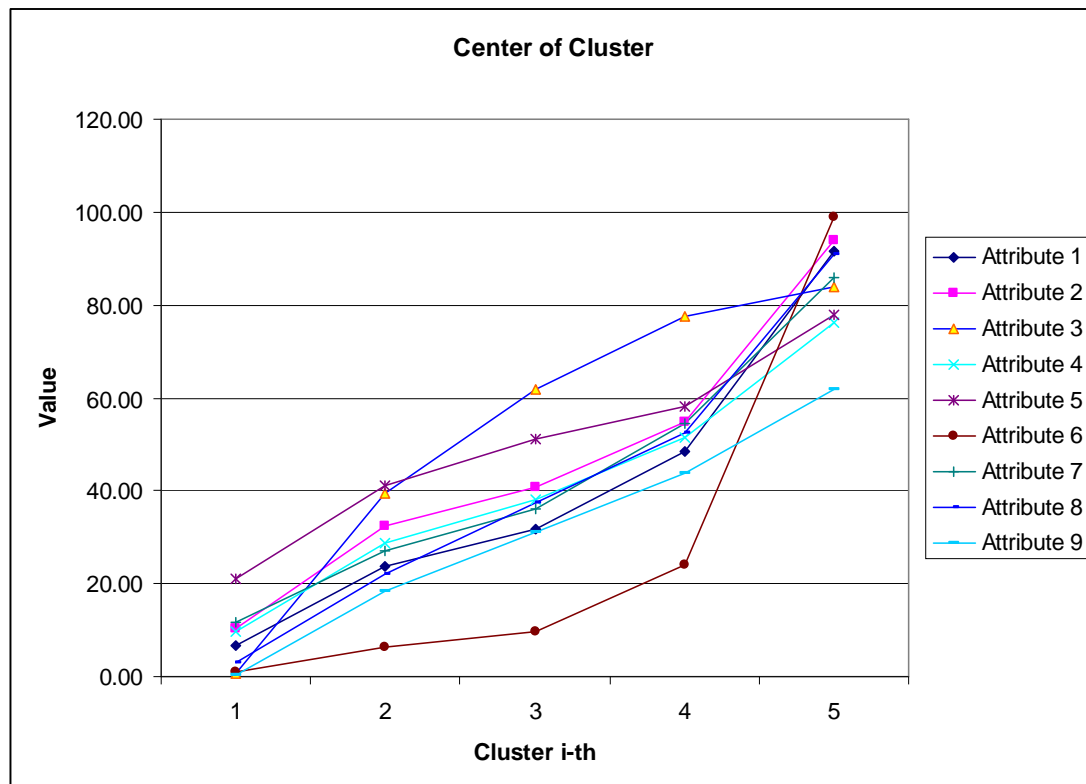
Ngược lại chọn số cụm bằng 5

Kết quả phân cụm với bộ dữ liệu Glass như sau:

Tất cả các thuộc tính đều được phân thành 5 cụm. Chú ý rằng sau quá trình chuẩn hoá giá trị về miền  $[0..100]$ , tất cả các tâm cụm là các số thực nằm trong khoảng này.

Thuộc tính	Cụm 1	Cụm 2	Cụm 3	Cụm 4	Cụm 5
1	6.58	23.72	31.81	48.48	91.74
2	10.49	32.30	40.78	54.66	94.01
3	0.51	39.52	61.75	77.49	83.87
4	9.83	28.69	38.26	51.46	76.27
5	21.16	41.06	51.18	58.18	77.94
6	0.88	6.43	9.79	24.08	99.01
7	11.72	27.01	36.22	54.36	85.94
8	3.13	21.94	37.33	52.49	90.81
9	0.20	18.29	31.02	43.70	61.81

Bảng 3: Kết quả phân cụm bài toán phân lớp Glass.



Hình 11: Biểu đồ các giá trị tâm cụm của bộ dữ liệu Glass.

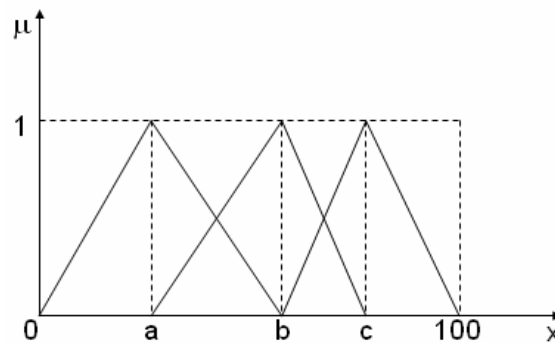
Một kỹ thuật để chọn số cụm tốt, đó là dựa vào đặc điểm thống kê của dữ liệu như kỳ vọng, phương sai, độ tự tương quan,... tuy nhiên cũng chưa thật sự có một phương pháp tối ưu. Đây chính là một hướng có thể tiếp tục được mở rộng của đề án.

### 3.2.3. Xây dựng các tập mờ tam giác

Sau khi tính được các giá trị tâm cụm ở bước trên, ta xây dựng các tập mờ tam giác từ các tâm cụm như sau:

- Sắp xếp các giá trị tâm cụm theo thứ tự tăng dần.
- Tập mờ tam giác có nút trái là tâm cụm trước, nút phải là tâm cụm sau, đỉnh là tâm cụm hiện tại.
- Riêng với tâm cụm giá trị nhỏ nhất thì nút trái là 0, với tâm cụm giá trị lớn nhất thì nút phải là 100.

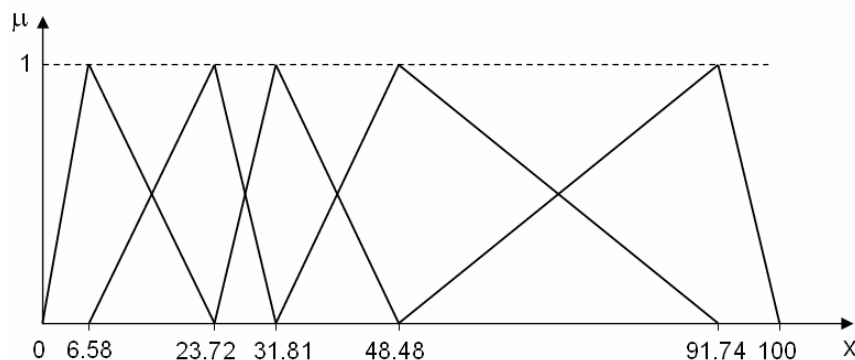
Ví dụ ta có 3 tâm cụm  $a, b, c$  đã sắp thứ tự. Ta xây dựng các tập mờ tam giác như sau:



Hình 12. Xây dựng tập mờ loại I từ các tâm cụm.

Chú ý rằng với tập mờ thứ nhất có đỉnh là  $a$ , nút trái là 0, nút phải là  $b$ . Với tập mờ thứ hai đỉnh là  $b$ , nút trái là  $a$ , nút phải là  $c$ . Với tập mờ thứ ba đỉnh là  $c$ , nút trái là  $b$ , nút phải là 100.

Kết quả xây dựng tập mờ tam giác đối với thuộc tính thứ nhất của bộ dữ liệu Glass như sau:



Hình 13: Xây dựng tập mờ tam giác cho thuộc tính thứ nhất của bộ dữ liệu Glass.

**Một số câu hỏi đặt ra trong vấn đề này:**

**Tại sao chọn tập mờ tam giác?** Tập mờ tam giác có một ưu điểm đặc trưng là giá trị độ thuộc tập trung ở giữa, giảm dần đều tuyến tính theo hai cạnh. Do đó, tập mờ tam giác rất hay được sử dụng trong các nghiên cứu về tập mờ.

**Tại sao các tập mờ tam giác lại chòm lên nhau?** Các tập mờ liên tiếp chòm lên nhau thể hiện một giá trị có thể xem là thuộc tập mờ này cũng có thể xem là thuộc tập mờ kia, một ưu điểm của hệ logic mờ giải quyết các bài toán với dữ liệu không chắc chắn, không chính xác.

**Tập mờ tam giác thể hiện điều gì?** Tập mờ tam giác thể hiện một “khoảng gần” trong suy diễn. Trong ví dụ tập mờ thuộc tính 1 của bộ dữ liệu Glass trên, ta có 5 “khoảng gần” là các số thực: 6.58, 23.72, 31.81, 48.48, 91.74. Giả sử với bài toán phân lớp mưa, giá trị “khoảng gần” mang ý nghĩa trong luật suy diễn như sau:

IF nhiệt độ “khoảng gần” 32°C AND sức gió “khoảng gần” 51km/h AND độ ẩm “khoảng gần” 80% THEN trời “mưa”

Rõ ràng, phương pháp tiếp cận tập mờ đã mang đến một cách tiếp cận rất hợp lý trong luật suy diễn.

### 3.2.4. Xác định các luật

Thuật toán xác định các luật như sau:

---

**Bước 1.** Duyệt qua tất cả các luật có thể:

For all rules:

*IF  $x_1$  is  $A_{i1}^1$  and  $x_2$  is  $A_{i2}^2$  and  $x_3$  is  $A_{i3}^3$  and...and  $x_n$  is  $A_{in}^n$  THEN  $y$  is  $k$*

với  $A_1^i, A_2^i, \dots, A_{Ci}^i$  là  $C_i$  tập mờ của thuộc tính  $i$ .

Đầu ra  $k = 1, 2, \dots, m$  với  $m$  là số lớp.

$i1$  là chỉ số tập mờ của thuộc tính 1,  $i2$  là chỉ số tập mờ của thuộc tính 2,...

và  $in$  là chỉ số tập mờ của thuộc tính  $n$ ,

**Bước 2:** Tính trọng số của các luật:

Đọc bản ghi thứ  $t$

Trọng số của luật  $j$  tương ứng với bản ghi này là:

$w_t = \text{Tích} \{ \text{độ thuộc ứng với từng thuộc tính trong tập mờ} \}$

Trọng số của luật bằng giá trị lớn nhất trong các giá trị  $w_t$

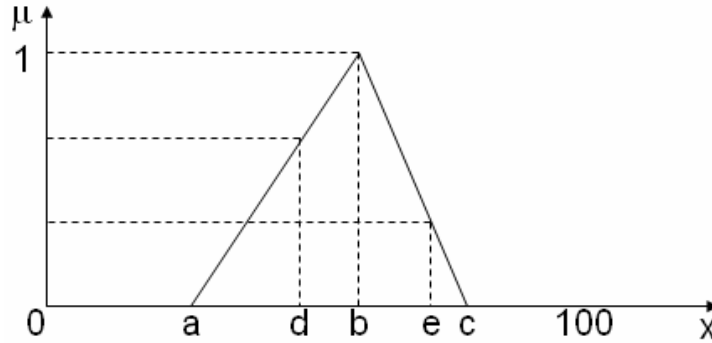
**Bước 3:** Loại bỏ luật nhỏ hơn ngưỡng cho trước.

**Bước 4:** Trong các luật còn lại, tồn tại những tập luật có cùng đầu vào và khác đầu ra, giữ lại luật có trọng số lớn nhất và loại bỏ những luật còn lại.

---

Ngưỡng được chọn từ đầu vào bài toán sao cho số luật thu được trong khoảng 200 - 300 luật. Với bài toán phân lớp Glass ngưỡng được chọn là 0.01.

Quá trình tính độ thuộc của từng thuộc tính trên từng tập mờ được minh họa trong hình vẽ sau:



Hình 14. Tính độ thuộc của một giá trị thuộc tính đối với tập mờ.

Giả sử ta có tập mờ tam giác như hình vẽ, có đỉnh tại b, mút trái tại a và mút phải tại c.

Nếu giá trị thuộc tính nhỏ hơn a hoặc lớn hơn c, trả lại giá trị 0.

Nếu giá trị thuộc tính d thỏa mãn:  $a < d < b$ , suy ra độ thuộc là: (theo định lý Talet về tính chất song song)

$$\frac{value}{1} = \frac{d - a}{b - a} \Rightarrow value = \frac{d - a}{b - a}$$

Nếu giá trị thực e thỏa mãn:  $b < e < c$ , suy ra độ thuộc là:

$$\frac{value}{1} = \frac{c - e}{c - b} \Rightarrow value = \frac{c - e}{c - b}$$

Việc lưu trữ các luật cũng là một vấn đề. Với số lượng thuộc tính ít, ví dụ 2 thuộc tính, ta có thể sử dụng mảng 2 chiều  $n_1 \times n_2$  với  $n_1$  là số tập mờ của thuộc tính 1,  $n_2$  là số tập mờ của thuộc tính 2, và giá trị trong mảng  $a[i][j]$  là trọng số của luật. Khi số lượng thuộc tính tăng lên, việc lưu trữ bằng mảng nhiều chiều tỏ ra không phù hợp, và sự dư thừa dữ liệu là vô cùng lớn. Trong trường hợp cần thực hiện nhiều thao tác tìm kiếm luật, việc lưu trữ luật bằng bảng băm tỏ ra là một cách lưu trữ hiệu quả. Tuy nhiên, toàn bộ chương trình chỉ thực hiện lấy từng luật ra và sử dụng nên cấu trúc lưu trữ bằng danh sách liên kết thông thường tỏ ra hiệu quả hơn cả.

### 3.2.5. Suy diễn trên hệ logic mờ loại I

Thuật toán suy diễn trên hệ logic mờ loại I:

---

**Input:** Tập các luật trên tập mờ loại I.  
Các bộ dữ liệu training.

---

**Output:** Số bộ dữ liệu training được test đúng

**Bước 1:** Đọc một bộ dữ liệu training.

**Bước 2:** value = 0; int result;  
float consider = 0; //hệ số xem xét

**Bước 3:** For tất cả các luật, với mỗi luật:  
    k = đầu ra của luật;  
    Tính độ thuộc của thuộc tính i:  $x_i$  trong tập mờ  $A_i$  là  $\mu_{A_i}(x_i)$   
     $w = \min\{\mu_{A_i}(x_i) \mid i = 1..n\}$   
     $consider\_k = \sum_{i=1}^n \mu_{A_i}(x_i)$

**Bước 4:** If (w > value)  
    value = w;  
    result = k;  
    consider = consider\_k  
Else if (w = value và consider\_k > consider)  
    value = w;  
    result = k;  
    consider = consider\_k  
End if  
End For

**Bước 5:** If (result = giá trị đầu ra đúng với bộ dữ liệu được test)  
    số trường hợp đúng ++

---

Đánh giá độ phức tạp thuật toán:  $O(nM)$  trong đó n là số thuộc tính và M là số luật. Trong thực tế thuật toán chạy rất hiệu quả và không mất nhiều thời gian.

**Trong thuật toán này có một số lưu ý:**

Trọng số của một bộ dữ liệu trên một luật được tính bằng phép min tất cả các độ thuộc của từng phần tử

Nếu trọng số đối với một luật là lớn nhất trong tất cả các luật, thì giá trị đầu ra bằng giá trị đầu ra của luật đó

Nếu có một vài luật có trọng số bằng nhau, ta xét đến hệ số xem xét consider, được tính bằng tổng tất cả các độ thuộc trong một luật. Chọn luật có trọng số lớn nhất và có giá trị consider lớn nhất.

Ví dụ. Giả sử với bộ dữ liệu Glass, ta có 2 luật:

IF  $x_1$  is  $A_1$  and ... and  $x_9$  is  $A_9$  THEN y is 2

IF  $x_1$  is  $A'_1$  and ... and  $x_9$  is  $A'_9$  THEN y is 3

Với một bộ dữ liệu, ta tính trọng số với luật thứ nhất và luật thứ hai bằng nhau:



$$w_1 = \min\{\mu_{A_i}(x_i) \mid i = 1..9\} = 0.2$$

$$w_2 = \min\{\mu_{A_i}(x_i) \mid i = 1..9\} = 0.2$$

thì ta tính đến hệ số consider =  $\sum_{i=1}^9 \mu_{A_i}(x_i)$  tức là tổng các độ thuộc

$$\text{consider}_1 = 2.43$$

$$\text{consider}_2 = 3.15$$

Ta sẽ theo luật có hệ số consider lớn hơn, tức là luật thứ hai, và kết luận bộ dữ liệu này thuộc lớp 3

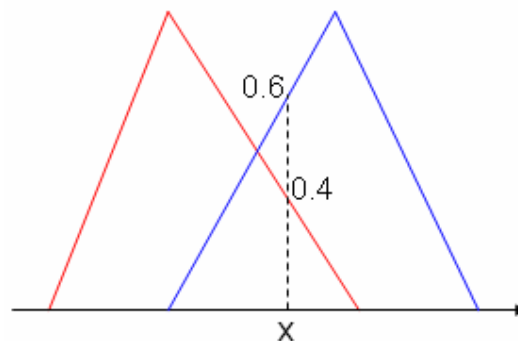
### 3.2.6. Đánh giá sai số hệ logic mờ loại I

**Nguyên nhân phân lớp sai trong tập mờ loại I:** đốt cháy bản ghi theo từng luật, chọn luật có giá trị đốt cháy lớn nhất. Phân lớp bản ghi theo luật chọn được. Do đó, nếu luật được chọn phân lớp sai thì bản ghi sẽ phân lớp sai. Ta gọi luật này là luật sai ứng với bản ghi đó.

**Tìm kiếm luật phân lớp đúng:** Trong quá trình training, trường hợp phân lớp sai, ta ghi lại luật sai đã chọn. Đốt cháy bản ghi trên tất cả các luật với đầu ra tương ứng với lớp đúng, chọn luật có giá trị đốt cháy lớn nhất, ta gọi là luật đúng.

Đây chính là khuyết điểm của tập mờ loại I: việc suy diễn tính toán hoàn toàn rõ, gây sai số trong nhiều trường hợp suy diễn với thông tin không chắc chắn có nhiều.

Một ví dụ minh họa sai số trong hệ logic mờ loại I được mô tả trong hình vẽ dưới đây:



Hình 15: Ví dụ về sai số trong hệ logic mờ loại I.

Giả sử ta có 2 tập mờ tam giác như hình vẽ. Một giá trị thực x được chiếu lên hai tập mờ ta có: độ thuộc trên tập mờ bên trái là 0.4 và độ thuộc trên tập mờ bên phải là 0.6. Do đó, theo ý nghĩa của tập mờ loại I, giá trị x được xem thuộc tập mờ bên phải “gần hơn” so với tập mờ bên trái.

Tuy nhiên, điều này chưa chắc chính xác trong nhiều trường hợp thực tế. Hơn nữa, tập mờ tam giác được tính ra trong các phần trước có thể có sai số, chưa chính xác. Ví dụ như khi nghiên cứu tuổi thọ của nhiều người cao tuổi, ta có thể thấy: khoảng cách từ 100 tuổi đến 110 tuổi là 10 năm, khoảng cách từ 110 tuổi đến 130 tuổi là 20 năm, do đó có thể kết luận 110 tuổi “gần” 100 tuổi hơn là 130 tuổi. Tuy nhiên, điều này có thể chưa chắc đúng. Vì 110 tuổi và 130 tuổi đều là những trường hợp rất ít xảy ra, ta có thể xem 110 tuổi và 130 tuổi là “gần nhau hơn” so với 110 tuổi và 100 tuổi.

### 3.2.7. Tổng kết pha I

Trong pha I, đồ án đã trình bày các bước để xây dựng hệ logic mờ loại I từ tập dữ liệu training đầu vào: xây dựng tập mờ loại I, xây dựng cơ sở luật, quá trình suy diễn trên tập mờ loại I và phát hiện nguyên nhân sai trong hệ logic mờ loại I. Kết thúc pha I ta thu được hệ logic mờ loại I hoàn chỉnh bao gồm: các tập mờ loại I và cơ sở luật trên tập mờ loại I. Trong các phần tiếp theo sẽ trình bày các quá trình trong pha II, trong đó sử dụng toàn bộ kết quả của pha I để xây dựng hệ logic mờ loại II đại số gia tử.

## 3.3. Các quá trình trong pha 2

### 3.3.1. Xây dựng cấu trúc đại số gia tử

Cấu trúc đại số gia tử đã được trình bày chi tiết trong Chương 2 phần 2. Toàn bộ các phần tử cấu trúc đại số gia tử có thể được biểu diễn bằng cây đại số gia tử. Cây đại số gia tử là cấu trúc cây tổng quát, có 2 gốc tách rời nhau ứng với phần tử sinh True và False, mỗi nút cha có  $n$  nút con trong đó  $n$  là tổng số gia tử,  $n = p+q$  với  $p$  là số phần tử sinh dương và  $q$  là số phần tử sinh âm.

Mỗi phần tử cấu trúc đại số gia tử được đặc trưng bởi hai tập giá trị:

- Giá trị biểu diễn:
  - length: độ dài của phần tử, được tính bằng tổng số gia tử và phần tử sinh.
  - generator: phần tử sinh, có giá trị True hoặc False.
  - hedge[]: mảng lưu trữ các gia tử liên tiếp nhau
  - sign: dấu của phần tử, được tính bằng tích các hàm SIG hai gia tử liên tiếp, gia tử cuối cùng với phần tử sinh và dấu của phần tử sinh.
- Giá trị định lượng ngữ nghĩa:
  - fm: độ đo tính mờ của giá trị ngữ nghĩa.

○  $\langle \underline{fm}(\hat{x}), v(\hat{x}), \overline{fm}(\hat{x}) \rangle$  : bộ ba giá trị định lượng ngữ nghĩa, trong đó  $\langle \underline{fm}(\hat{x}), \overline{fm}(\hat{x}) \rangle$  là khoảng mờ và  $\langle v(\hat{x}) \rangle$  là ánh xạ định lượng ngữ nghĩa bảo toàn thứ tự trên  $[0,1]$ .

Ta có thể đồng thời xây dựng cây đại số gia tử và tính hàm dấu sign và bộ ba giá trị định lượng ngữ nghĩa của tất cả các phần tử. Trong [1] có trình bày thuật toán tính các giá trị này, tuy nhiên chỉ tính cho một giá trị ngôn ngữ riêng lẻ. Phần này sẽ trình bày thuật toán cải tiến tính bộ ba giá trị định lượng ngữ nghĩa cho cả cây đại số gia tử.

---

**Đầu vào:**  $fm(c-) = \theta, H- = \{h_{-q}, \dots, h_{-1}\}, H+ = \{h_1, \dots, h_p\}, \mu(h_i)$ , quan hệ SIG, độ dài ngôn ngữ lớn nhất cho phép  $K$

**Đầu ra:**  $sign, \underline{fm}, v, \overline{fm}$  của toàn bộ các phần tử đại số gia tử.

**BEGIN**

$\alpha \leftarrow 0$

for  $i = -1$  downto  $-q$  do  $\alpha \leftarrow \alpha + \mu(h_i)$

Tree.CreateRoot(False)

$Root(False) : length \leftarrow 0, sign \leftarrow -1, \underline{fm} \leftarrow 0, \overline{fm} \leftarrow \theta, v \leftarrow (1 - \alpha) \times \theta$

Tree.CreateRoot(True)

$Root(True) : length \leftarrow 0, sign \leftarrow 1, \underline{fm} \leftarrow \theta, \overline{fm} \leftarrow 1, v \leftarrow (1 + \alpha) \times \theta$

$k \leftarrow 1$

while( $k < K$ )

begin

$k \leftarrow k + 1$

(\* tính toàn bộ nút mức  $k$  \*)

for(toàn bộ nút ở mức  $k-1$  :  $node$  )

begin

$fmP \leftarrow node.\overline{fm} - node.\underline{fm}$

$signP \leftarrow node.sign$

$leftP \leftarrow node.\underline{fm}$

$rightP \leftarrow node.\overline{fm}$

$hP \leftarrow node.hedges[node.length]$

for  $i = -q$  to  $p$  do

begin

(\* sinh nút con với gia tử tương ứng \*)

$node.CreateNode(childNode, h_i)$

$signC \leftarrow signP \times SIG(h_i, hP)$

if  $signC = 1$  then

```

    if  $i > 0$  then
         $right\_decrease \leftarrow 0$ 
        for  $j = i + 1$  to  $p$  do  $right\_decrease \leftarrow right\_decrease + \mu(h_j)$ 
    else
         $right\_decrease \leftarrow 0$ 
        for  $j = i - 1$  to  $-q$  do  $right\_decrease \leftarrow right\_decrease + \mu(h_j)$ 
    end if
     $childNode : sign \leftarrow signC$ ,
     $\overline{fm} \leftarrow rightP - right\_decrease \times fmP$ ,
     $\underline{fm} \leftarrow \overline{fm} - \mu(h_i) \times fmP$ 
else
    if  $i > 0$  then
         $left\_increase \leftarrow 0$ 
        for  $j = i + 1$  to  $p$  do  $left\_increase \leftarrow left\_increase + \mu(h_j)$ 
    else
         $left\_increase \leftarrow 0$ 
        for  $j = i - 1$  to  $-q$  do  $left\_increase \leftarrow left\_increase + \mu(h_j)$ 
    end if
     $childNode : sign \leftarrow signC$ ,
     $\underline{fm} \leftarrow leftP + left\_increase \times fmP$ ,
     $\overline{fm} \leftarrow \underline{fm} + \mu(h_i) \times fmP$ 
end if
if  $signC \times SIG(h_p, h_i) = -1$  then  $childNode : v \leftarrow \underline{fm} + (1 - \alpha) \times \mu(h_i) \times fmP$ 
else  $childNode : v \leftarrow \overline{fm} + \alpha \times \mu(h_i) \times fmP$ 
end for
end for
end while
END.

```

#### **Đánh giá độ phức tạp thuật toán:**

Sinh và tính các giá trị định lượng ngữ nghĩa cho nút mới:  $O(\max(p, q))$

Số vòng lặp for trong cùng được thực hiện chính bằng số nút mới được tạo ra.

Tổng số vòng lặp đã thực hiện (while và for) chính bằng tổng số nút trên cây. Từ đó độ phức tạp thuật toán bằng  $O(N \times \max\{p, q\})$ , với  $N$  là số nút trên cây.

Ký hiệu  $n = p + q$  là tổng số gia tử, số nút trên cây  $N$  có thể được tính như sau:

$$N = (1 + n + n^2 + \dots + n^K) \times 2 = \frac{2(n^{K+1} - 1)}{n - 1} \text{ với } K \text{ là độ cao của cây. Chú ý rằng ta}$$

có 2 cây với 2 nút gốc là True và False.

### 3.3.2. Xây dựng tập mờ loại hai đại số gia tử

Sau khi đã xây dựng hệ logic mờ loại I hoàn chỉnh trong pha 1, ta chuyển tập mờ loại I thành tập mờ loại II đại số gia tử đồng thời dựa trên sai số của hệ logic mờ loại I để xây dựng đa cấu trúc đại số gia tử trong hệ logic mờ loại II đại số gia tử.

Như đã trình bày trong phần nền tảng lý thuyết chương 2, tập mờ loại I có độ thuộc là các số thực và tập mờ loại II đại số gia tử có độ thuộc là giá trị ngôn ngữ. Vì vậy, quá trình chuyển từ tập mờ loại I thành tập mờ loại II đại số gia tử thực chất là chuyển từ số thực thành giá trị ngôn ngữ. Quá trình chuyển từ số thực thành giá trị ngôn ngữ tổng quát như sau:

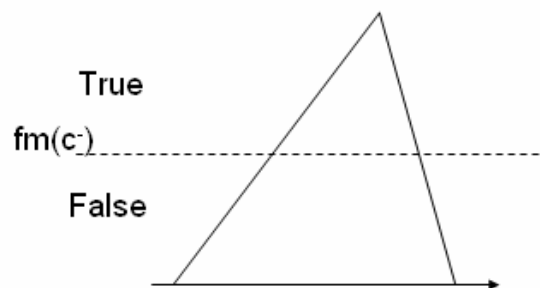
Với số thực  $x$ :

Tìm phần tử đại số gia tử  $\hat{x}$  sao cho giá trị  $v(\hat{x})$  gần với  $x$  nhất.

Trả về phần tử  $\hat{x}$

Với hệ logic mờ loại II đại số gia tử đã trình bày trong [1], hệ logic mờ loại II được xây dựng chỉ với một cấu trúc đại số gia tử. Điều này có nghĩa là, cấu trúc đại số gia tử đó hoàn toàn xác định duy nhất các tham số đầu vào:  $fm(c^-), \mu(h_i)$ .

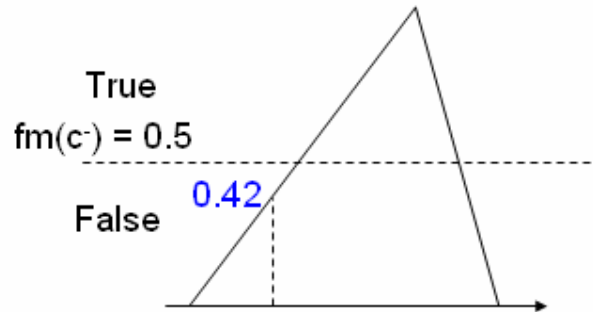
Giá trị  $fm(c^-)$  sẽ chia thành hai miền giá trị ngôn ngữ tương ứng với phần tử sinh là True và False



Hình 16.  $fm(c^-)$  chia khoảng  $[0,1]$  thành hai miền tương ứng với giá trị ngôn ngữ phần tử sinh là True và False.

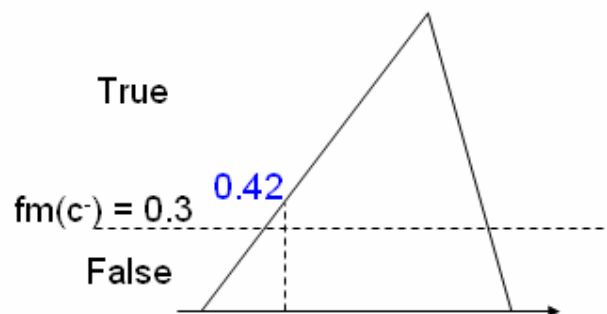
Khi  $fm(c^-)$  thay đổi, ví dụ  $fm(c^-)$  càng nhỏ, thì giá trị thực nhỏ lại có thể có ý nghĩa lớn. Ngược lại,  $fm(c^-)$  càng lớn, thì giá trị thực lớn lại có thể có ý nghĩa nhỏ.

Ví dụ: Với  $fm(c^-) = 0.5$



Hình 17. Với  $fm(c^-) = 0.5$ , khi đó giá trị 0.42 có thể xem là Less False

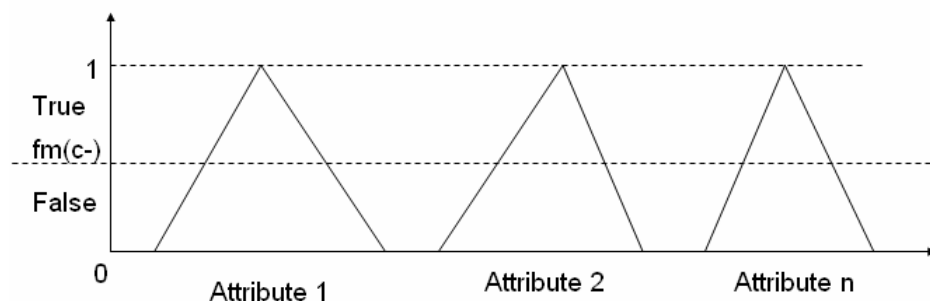
Với  $fm(c^-) = 0.3$



Hình 18. Với  $fm(c^-) = 0.3$ , khi đó giá trị 0.42 có thể xem là Less True

Đề tài nghiên cứu này sẽ xây dựng nhiều cấu trúc đại số gia tử trong các luật, và khi đó, các giá trị ngôn ngữ thực hiện các phép toán với nhau chỉ tác động lên thành phần biểu diễn của chúng (gia tử và phần tử sinh), không tác động lên các thành phần ngữ nghĩa của chúng  $\langle \underline{fm}(\hat{x}), v(\hat{x}), \overline{fm}(\hat{x}) \rangle$ .

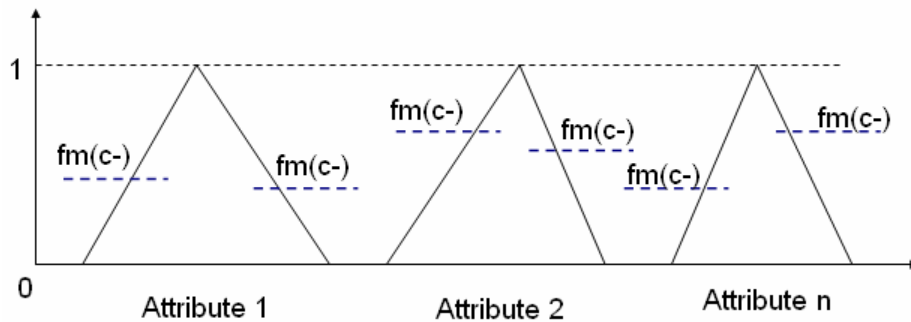
Đối với phương pháp xây dựng tập mờ loại II đại số gia tử từ tập mờ loại I đã xét trong [1], khi đó toàn bộ độ thuộc tập mờ được chuyển thành giá trị ngôn ngữ của một cấu trúc đại số gia tử. Cấu trúc đại số gia tử này được khởi tạo với các tham số  $fm(c^-), \mu(h_i)$ . Khi đó, toàn bộ các thuộc tính được nhìn nhận ranh giới giữa True và False như nhau.



Hình 19. Giá trị  $fm(c^-)$  phân cách True và False đối với mọi thuộc tính

Phương pháp này có nghĩa là, đối với mọi thuộc tính, mức độ gần với giá trị đỉnh tập mờ tam giác là gần như nhau. Tuy nhiên, điều này không hẳn đúng đối với nhiều trường hợp. Đối với vài thuộc tính, khoảng cách dù nhỏ nhưng có thể ta vẫn chưa xem là gần, ví dụ đối với lượng vi chất trong cơ thể người, thiếu vài chục mg canxi không được coi là thiếu canxi tuy nhiên thiếu vài chục mg iot có thể nói là thiếu iot trầm trọng.

Từ đó hướng cải tiến của đề án này so với [1] là: đối với mỗi thuộc tính ta có giá trị  $fm(c-)$  riêng, hay các thuộc tính khác nhau có cấu trúc đại số gia tử khác nhau. Toàn bộ các cấu trúc đại số gia tử này đều có cùng số lượng gia tử và giá trị các gia tử, tức là giống nhau về mặt biểu diễn ngôn ngữ nhưng khác nhau về các giá trị định lượng ngữ nghĩa. Đặc biệt, do đặc điểm của tập mờ tam giác, mỗi tam giác có 2 cạnh có thể coi là độc lập do đó mỗi tập mờ tam giác tương ứng với 2 giá trị  $fm(c-)$ , tức là 2 cấu trúc đại số gia tử.



Hình 20. Đa cấu trúc đại số gia tử đối với các thuộc tính.

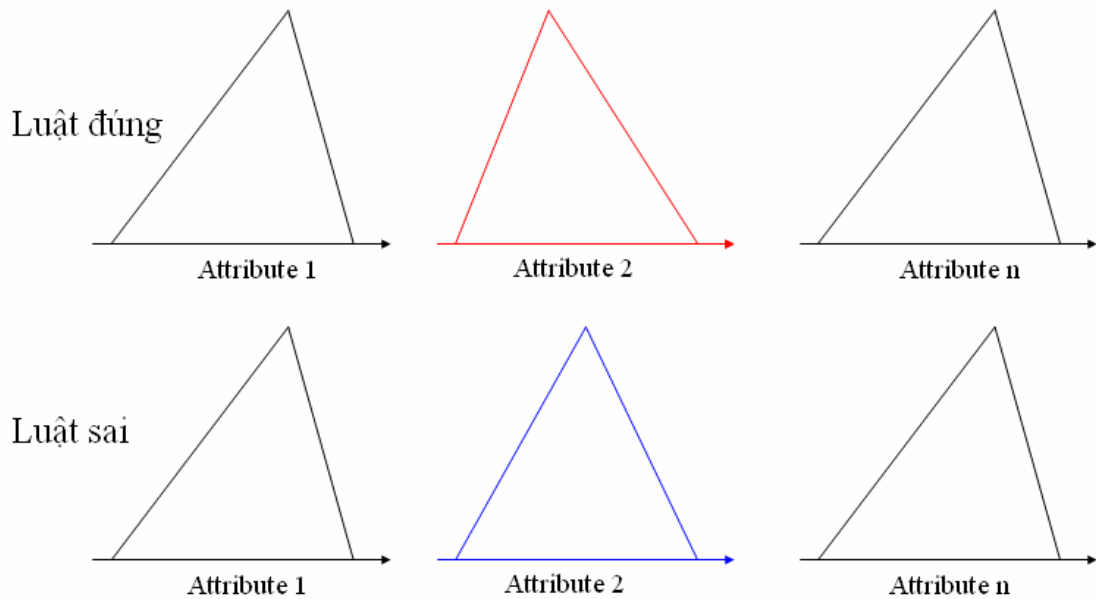
### Sử dụng sai số hệ logic mờ loại I xây dựng luật cho hệ logic mờ loại II đa cấu trúc đại số gia tử

Trong pha 1 đã trình bày phần trước, phần 6 của pha 1 ta đã có đánh giá về sai số của hệ logic mờ loại I. Trong đó, luật sai là luật gây ra suy diễn sai trên những bản ghi training, là luật có đầu ra sai và giá trị đốt cháy lớn nhất; luật đúng là luật có đầu ra đúng và giá trị đốt cháy lớn nhất.

Xét hai luật sai và luật đúng có dạng:

$$R^l : IF \ x_1 \text{ is } \hat{F}_1^l \text{ and...and } IF \ x_p \text{ is } \hat{F}_p^l \text{ THEN } y \text{ is } \hat{G}^l$$

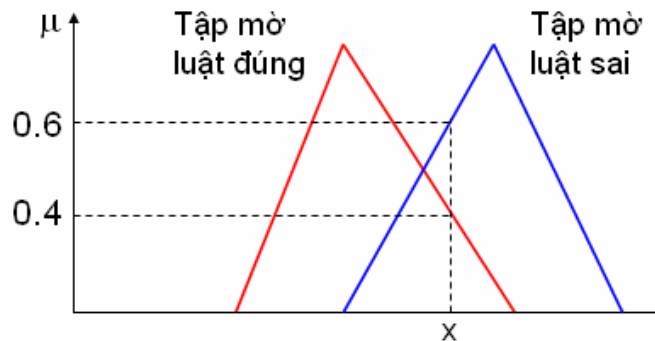
Hai luật này sẽ có những thuộc tính mà tập mờ tương ứng giống nhau, và những thuộc tính còn lại tập mờ tương ứng khác nhau, ví dụ trong hình vẽ sau:



Hình 21: Luật đúng và luật sai

Ví dụ trong hình vẽ trên, luật đúng và luật sai khác nhau ở tập mờ thuộc tính 2, các tập mờ còn lại của thuộc tính 1 và n giống nhau. Ta chỉ xét các tập mờ khác nhau trong 2 luật.

Vì đặc điểm các tập mờ tam giác được xây dựng có phần chòm lên nhau, nên các tập mờ khác nhau phải là các tập mờ liên tiếp, và tồn tại độ thuộc lên tập mờ của *luật sai* lớn hơn độ thuộc lên tập mờ của *luật đúng*. Trong ví dụ hình vẽ trên giả sử tập mờ luật đúng ở bên trái tập mờ luật sai như sau:



Hình 22. Độ thuộc trên tập mờ của luật đúng (bên trái) nhỏ hơn độ thuộc trên tập mờ của luật sai (bên phải)

Ta biến đổi  $f_m(c^-)$  của tập mờ luật đúng sao cho chuyển sang giá trị ngôn ngữ thì độ thuộc trên tập mờ luật đúng phải lớn hơn độ thuộc trên tập mờ luật sai.

Đối với ví dụ trên, giả sử ta có:

Với tập mờ của luật sai:  $f_{m_1}(c^-) = 0.5$  và  $\mu_1(x) = 0.6$

Với tập mờ của luật đúng:  $f_{m_2}(c^-) = 0.5$  và  $\mu_2(x) = 0.4$

Ta đưa vào một hằng số  $\varepsilon$  được khởi tạo trước ( $=0.01$ ). Cho hai tỷ số sau tỷ lệ thuận:

$f_{m_1}(c^-)$  và  $\mu_1(x)$



$fm'_2(c^-)$  và  $\mu_2(x) - \varepsilon$   
 Từ đó, ta tính lại  $fm'_2(c^-)$  như sau:

$$fm'_2(c^-) = \frac{\mu_2(x) - \varepsilon}{\mu_1(x)} \times fm_1(c^-)$$

Với ví dụ trên thì  $fm'_2(c^-) = 0.39 \times 0.5 / 0.6 = 0.325$ . Như vậy thực chất ta đã giảm  $fm(c^-)$  của tập mờ luật đúng, sao cho khi ánh xạ sang giá trị ngôn ngữ đại số gia tử thì độ thuộc trên tập mờ luật đúng lớn hơn trên tập mờ luật sai.

### 3.3.3. Suy diễn trên hệ logic mờ loại II đại số gia tử

Phần này nhận đầu vào là dữ liệu test TestData, và thực hiện suy diễn trên hệ logic mờ loại II đại số gia tử để cho ra kết quả đầu ra là phân lớp tương ứng với bản ghi test đó. Quá trình này gồm hai phần:

**Chuẩn hoá:** Bộ dữ liệu test có các dữ liệu là số thực, tương tự như trong phần tiền xử lý dữ liệu TrainingData, dữ liệu test cũng phải được chuẩn hoá về miền [0..100]. Giả sử giá trị  $a$  của thuộc tính  $i$  nằm trong dải  $[aMin..aMax]$ , sử dụng phép chuẩn hoá tuyến tính về miền  $[0..100]$ :

$$a \rightarrow \frac{a - aMin}{aMax - aMin} \times 100$$

Sau khi chuẩn hoá dữ liệu sẽ nằm trong dải  $[0..100]$ .

**Suy diễn:** Quá trình suy diễn này được thực hiện trên hệ logic mờ loại II đại số gia tử. Do đó, trong quá trình suy diễn, mỗi giá trị thực được chiếu lên lấy độ thuộc trong tập mờ và sau đó chuyển từ giá trị thực thành giá trị ngôn ngữ của đại số gia tử, dựa vào giá trị  $fm(c^-)$  của cấu trúc đại số gia tử tương ứng với tập mờ đó. Thuật toán suy diễn trên tập mờ loại II đại số gia tử như sau:

---

**Input:** Tập các luật trên tập mờ loại II HaT2FS

Các bộ dữ liệu test

**Output:** Số bộ dữ liệu test đúng

**Bước 1:** Đọc một bộ dữ liệu test

**Bước 2:** value = VVVF; int result;  
 float consider = 0; //hệ số xem xét

**Bước 3:** For tất cả các luật, với mỗi luật:

k = đầu ra của luật;

Với mỗi bộ  $(x_1, x_2, \dots, x_n)$  với  $n$  là số thuộc tính

$$\mu_t = meet(\mu_1(x_1), \mu_2(x_2), \dots, \mu_n(x_n))$$

$$consider\_k = \sum_{i=1}^n \mu_i(x_i)$$

**Bước 4:** If ( $v(w) > v(value)$ )

---

```

value = w;
result = k;
consider = consider_k
Else if (v(w) = v(value) và consider_k > consider)
value = w;
result = k;
consider = consider_k
End if
End For

```

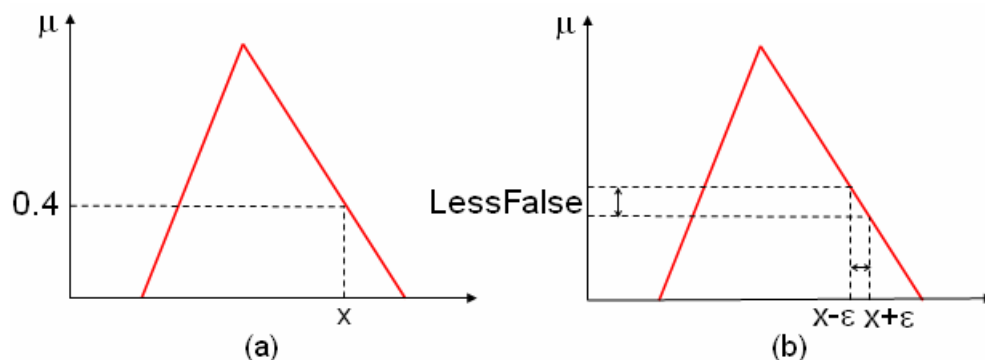
**Bước 5:** If (result = giá trị đầu ra đúng với bộ dữ liệu test)  
số trường hợp đúng ++

Một số câu hỏi trong phần này:

**Suy diễn trên tập mờ loại II đại số gia tử hơn gì suy diễn trên tập mờ loại I?**

Tập mờ loại II đại số gia tử đã cải thiện hai vấn đề sau:

**Thứ nhất**, sử dụng ngữ nghĩa thay cho số thực khiến cho việc suy diễn được thực hiện trên một lân cận nhỏ (khoảng nhỏ) thay vì trực tiếp một số thực rõ. Như ta đã biết trong các phần trước, tập mờ loại I có khuyết điểm là suy diễn trên tập mờ có độ thuộc của phần tử là số thực rõ. Tập mờ loại II đại số gia tử cải thiện vấn đề này và xem rằng trong một lân cận nhỏ từ các phần tử được xem là tương đương, được thể hiện rõ trong hình vẽ sau:



Hình 23: Với tập mờ loại I độ thuộc là một số thực rõ (hình a), với tập mờ loại II đại số gia tử là một giá trị ngôn ngữ (hình b), thể hiện một khoảng lân cận nhỏ trong quá trình suy diễn được xem là tương đương.

**Thứ hai**, như đã trình bày ở phần 6 của pha 1, sai số của hệ logic mờ loại I chính là do bản thân hình dạng tập mờ loại I đã quyết định luôn độ thuộc của phần tử, và do đó xuất hiện sai số do chính hình dạng của tập mờ loại I. Trong phần 1 của pha 2, sử dụng sai số của hệ logic mờ loại I để xây dựng cấu trúc đại số gia tử cho tập mờ loại II đại số gia tử đã thể hiện những ưu điểm trong xử lý thông tin đa dạng trong thực tế.

**Thứ ba**, hướng tiếp cận suy diễn trên hệ logic mờ loại II đại số gia tử chính là một hướng tiếp cận gần hơn với suy nghĩ của con người, vì vậy ngoài bài toán phân lớp, đây là một trong những hướng nghiên cứu quan trọng trong điều khiển tự động, robot. Để minh họa ta xét một ví dụ sau:

Ví dụ: **Phân lớp “phỏng vấn”**.

Một nhóm 40 người ứng viên nộp hồ sơ tuyển dụng và trải qua một bài kiểm tra (test). Dựa trên hồ sơ các ứng viên như bằng tốt nghiệp, thành tích học tập, điểm tiếng Anh, kinh nghiệm hoạt động xã hội, kinh nghiệm làm việc và kết quả bài test để đánh giá chọn ra 10 ứng viên xuất sắc nhất chọn vào vòng “phỏng vấn”. Đây cũng có thể coi là bài toán phân lớp, với 2 lớp là lớp “phỏng vấn” và lớp “không được phỏng vấn”. Các thuộc tính chính là các tiêu chí đánh giá ứng viên. Dữ liệu Training là các tiêu chí của giám đốc nhân sự, hay người quyết định tạo thành. Ta sẽ nghiên cứu cách thức phân lớp của con người.

- Suy nghĩ con người có cách tiếp cận như tập mờ loại I. Thật vậy, giả sử điểm tiếng Anh yêu cầu là trên 600 TOEIC, tuy nhiên người được 900 TOEIC được đánh giá cao hơn người được 600 TOEIC, đây chính là ý tưởng của tập mờ loại I. Thêm nữa, giả sử có người được 580 TOEIC, có lẽ không vì thế mà bị loại ngay, đó chính là ý tưởng của các tập mờ chớm lên nhau.
- Suy nghĩ con người lại không cứng nhắc như tập mờ loại I. Thật vậy, tập mờ loại I có độ thuộc bằng cách chiếu các phần tử lên tập mờ. Ví dụ 900 TOEIC có độ thuộc là 0.95, và 600 TOEIC có độ thuộc là 0.2. Tuy nhiên, suy nghĩ con người không cứng nhắc như vậy.
- Suy nghĩ con người có hướng tiếp cận là “rất được”, “gần được”, “quá ổn”, chính là hướng của đồ án tiếp cận đại số gia tử vào tập mờ. Ví dụ 900 TOEIC có thể giám đốc nhân sự sẽ đánh giá “rất tuyệt vời”, 600 TOEIC được đánh giá là “tạm được”. Quá trình tích hợp các thuộc tính chính là các phép toán trên các giá trị ngôn ngữ, như đã trình bày là phép *meet* và *join* các giá trị ngôn ngữ. Các phép toán này có thể tiếp tục được cải tiến trong tương lai.

### 3.3.4. Tổng kết pha 2

Trong toàn bộ pha 2, đồ án đã trình bày quá trình chuyển hệ logic mờ loại I thành hệ logic mờ loại II đại số gia tử, trong đó kế thừa toàn bộ cơ sở luật và sử dụng sai số của hệ logic mờ loại I. Quá trình suy diễn trên hệ logic mờ loại II đại số gia tử tương tự như quá trình suy diễn trên hệ logic mờ loại I, chỉ khác là độ thuộc được thực hiện trên các giá trị ngôn ngữ, điển hình là phép *meet* hai giá trị ngôn ngữ. Phép *join* không có trong phần suy diễn vì bài toán phân lớp với đầu ra là các giá trị số nguyên rõ, do đó không cần *join* các kết quả từ các luật lại mà chỉ chọn ra kết quả có giá trị lớn nhất. Một số ưu điểm của hệ logic mờ loại II đại số gia tử so với hệ logic mờ loại I cũng đã được trình bày ở phần 2 pha 2.

## CHƯƠNG 4

# CHƯƠNG TRÌNH GIẢI BÀI TOÁN PHÂN LỚP

### 4.1. Bộ dữ liệu thử nghiệm

Đồ án sử dụng hai bộ dữ liệu chuẩn của UCI Machine Learning Repository là bộ dữ liệu Glass và Ecoli. Đặc điểm của bộ dữ liệu Glass đã được trình bày trong phần 1 chương 3. Phần này sẽ nhắc lại những đặc điểm quan trọng nhất.

Bộ dữ liệu Glass (Glass Dataset) [4] là bộ dữ liệu về các thành phần trong kính và phân lớp kính. Dữ liệu có 9 thuộc tính, các thuộc tính này là lượng các thành phần một số đơn chất trong kính (RI, Na, Mg, Al, Si, K, Ca, Ba, Fe), và được phân thành 6 lớp được đánh số từ 1 đến 6. Toàn bộ dữ liệu có 214 mẫu.

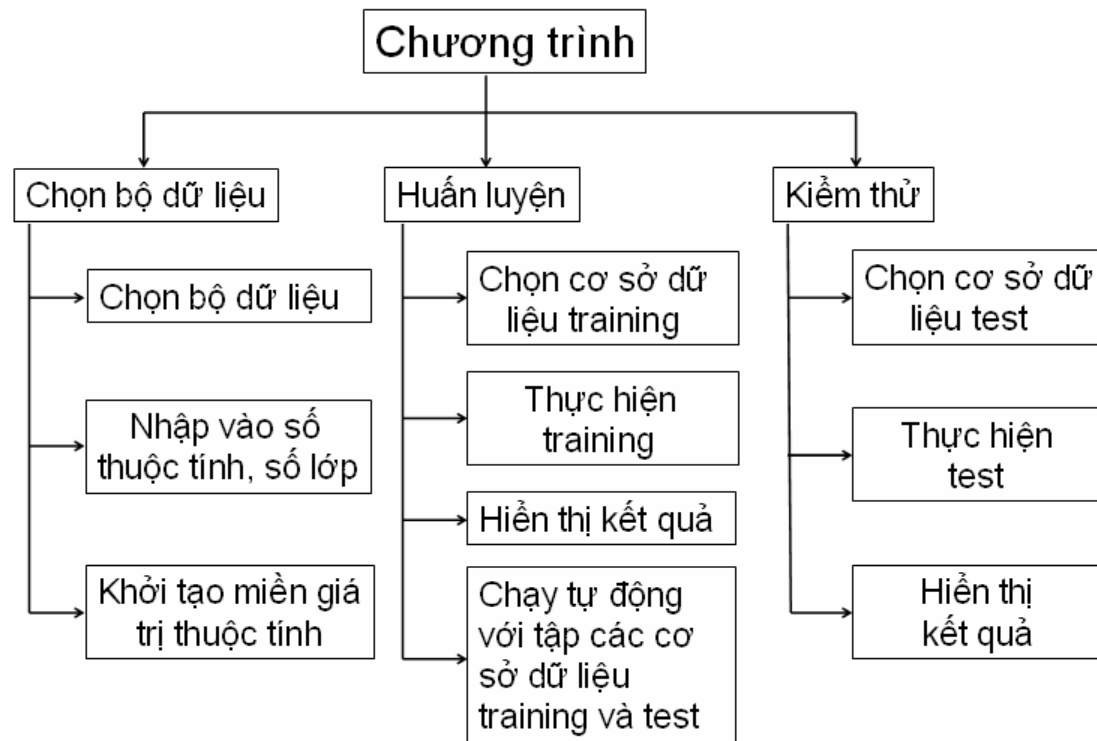
Bộ dữ liệu Ecoli (Ecoli Dataset) [5] là bộ dữ liệu nghiên cứu về các thành phần protein trong tế bào vi khuẩn. Dữ liệu có 7 thuộc tính, các thuộc tính này là lượng các thành phần sinh học và hoá học (mcg, gvh, lip, chg, aac, alm1, alm2), và được phân thành 8 lớp (cp, im, pp, imU, om, omL, imL, imS). Toàn bộ dữ liệu có 336 mẫu.

### 4.2. Phân tích thiết kế ứng dụng

#### 4.2.1. Biểu đồ phân cấp chức năng

Chương trình gồm các chức năng chính:

- Chức năng lựa chọn bộ dữ liệu (dataset) cho bài toán phân lớp.
- Chức năng huấn luyện bộ dữ liệu training.
- Chức năng kiểm thử với bộ dữ liệu test.



Hình 24. Biểu đồ phân cấp chức năng hệ thống

### Chức năng chọn bộ dữ liệu

Chức năng này cho phép người dùng chọn bộ dữ liệu (dataset) cần phân lớp. Từ đó xác định số thuộc tính, số lớp tương ứng với bộ dữ liệu đó. Và cuối cùng người dùng cần khởi tạo miền giá trị cho từng thuộc tính trong bộ dữ liệu.

Một lưu ý trong chức năng này đó là, mỗi bộ dữ liệu có cấu trúc định nghĩa khác nhau, tên lớp khác nhau, file dữ liệu chuẩn có định dạng khác nhau. Vì vậy, thực tế chương trình phải được định nghĩa một danh sách các bộ dữ liệu được chọn.

### Chức năng huấn luyện

Chức năng này cho phép người dùng chọn file dữ liệu training và bắt đầu tiến hành quá trình training. Kết quả sau khi training bao gồm số cụm, số luật, thời gian chạy, bảng kết quả các cụm, bảng kết quả các luật.

**Chức năng chọn cơ sở dữ liệu training.** Cơ sở dữ liệu này được lưu trong file text. Định dạng file cơ sở dữ liệu phải tuân theo định dạng chuẩn của UCI Machine Learning Repository. Người dùng chỉ cần thao tác browse đến file training. Định nghĩa bộ dữ liệu đã được chọn từ chức năng chọn bộ dữ liệu, vì vậy chương trình có thể đọc được chính xác cơ sở dữ liệu tương ứng.

**Chức năng thực hiện training.** Phần này sẽ thực hiện toàn bộ quá trình training như đã trình bày trong pha 1 của chương 3. Người dùng chỉ cần thao tác nhấn nút

Start Training trên giao diện chương trình. Một lưu ý là ngưỡng loại bỏ luật có thể được thay đổi bởi người dùng.

**Chức năng hiển thị kết quả.** Kết quả của quá trình training được hiển thị trên giao diện chương trình, bao gồm số cụm dữ liệu ứng với từng thuộc tính, số luật, thời gian training; ngoài ra kết quả còn có thể được lưu ra riêng các file Excel (.xls) với các dữ liệu có số lượng lớn như bảng kết quả các cụm dữ liệu, bảng kết quả các luật được tạo thành.

**Chức năng chạy tự động với tập các cơ sở dữ liệu training và test.** Mỗi lần chạy thử một bộ dữ liệu mất khá nhiều thao tác của người dùng và mất thời gian. Vì vậy, để chạy thử vài trăm lần có lẽ là một phức tạp với việc thao tác lặp lại vài trăm lần của người dùng. Chức năng này cho phép người sử dụng đặt vào hàng trăm file cơ sở dữ liệu TrainingData và TestData ở cùng thư mục với chương trình, chọn 2 số nguyên a và b tương ứng với chỉ số file đầu tiên và chỉ số file cuối cùng và nhấn nút chạy tự động. Ví dụ, người dùng nhập vào số 1 và số 300, chương trình sẽ tự động tìm và chạy các file Training1, Test1, Training2, Test2, ..., Training300, Test300 trong cùng thư mục với chương trình. Khi xong một bộ test chương trình sẽ ghi ra kết quả ra file có tên bao gồm tên file đầu vào tương ứng và thêm phần “\_Result” vào cuối.

### **Chức năng kiểm thử**

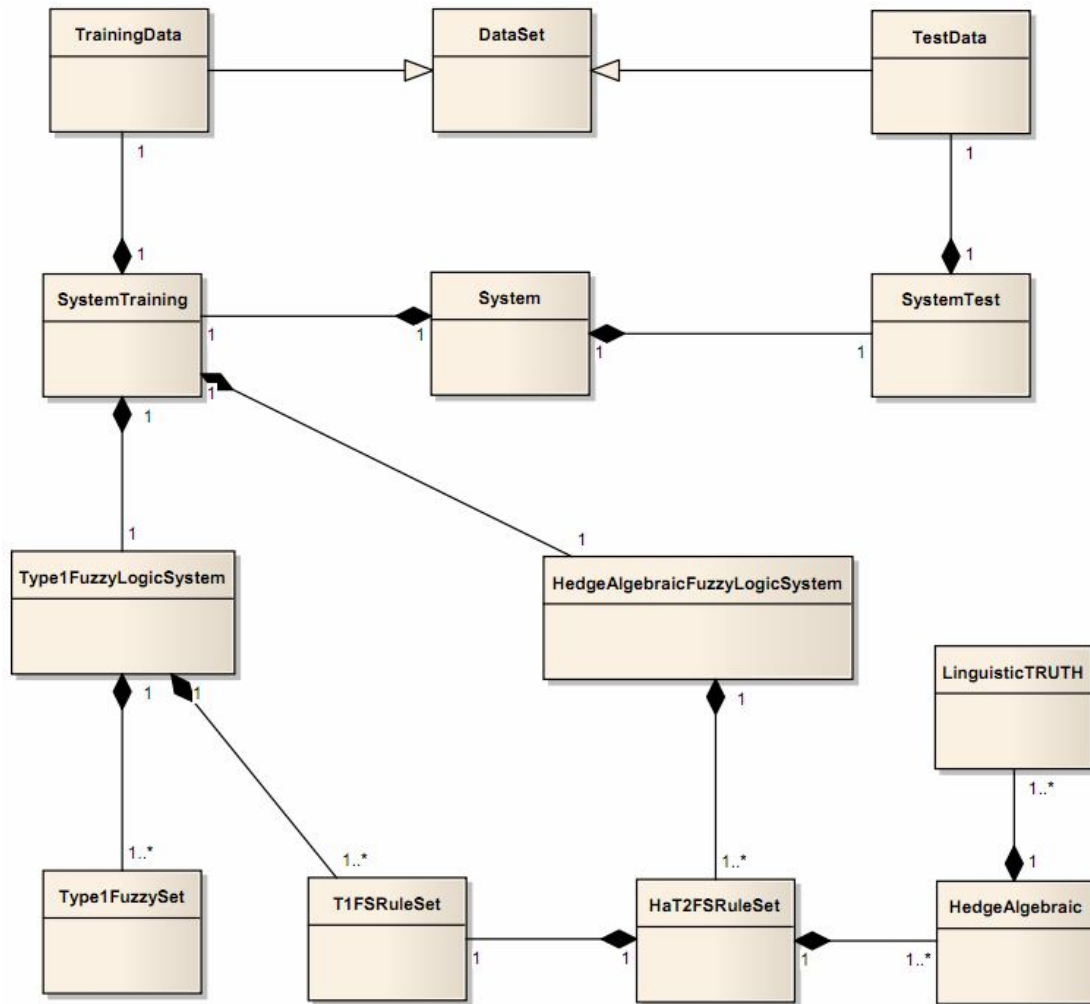
Chức năng này cho phép người dùng chọn file dữ liệu test và bắt đầu tiến hành quá trình test. Kết quả sau khi test bao gồm số bản ghi phân lớp đúng và danh sách từng bản ghi được phân lớp với cả hệ logic mờ loại I và hệ logic mờ loại II đại số gia tử, thời gian chạy quá trình test đối với cả hệ logic mờ loại I và hệ logic mờ loại II đại số gia tử.

**Chức năng chọn cơ sở dữ liệu test.** Người dùng browse đến file cơ sở dữ liệu Test. Tương tự như file cơ sở dữ liệu Training, file cơ sở dữ liệu Test cần phải đúng định dạng.

**Chức năng thực hiện Test.** Người dùng chỉ cần nhấn nút Start Testing, toàn bộ quá trình được thực hiện như đã trình bày trong pha 2 của chương 3, quá trình suy diễn được thực hiện trên cả hệ logic mờ loại I và hệ logic mờ loại II đại số gia tử và cho ra kết quả phân lớp với từng bản ghi.

**Chức năng hiển thị kết quả.** Toàn bộ kết quả phân lớp và thời gian chạy ứng với hệ logic mờ loại I và hệ logic mờ loại II đại số gia tử được hiển thị trên giao diện chương trình hoặc cũng có thể ghi ra riêng file Excel (.xls)

## **4.2.2. Biểu đồ lớp của chương trình**



Hình 25. Biểu đồ lớp của hệ thống

Biểu đồ lớp của hệ thống bao gồm các thành phần chính sau:

- Lớp Dataset: Tập dữ liệu đầu vào cho chương trình, gồm tập dữ liệu Training và tập dữ liệu Test.
- Lớp System: Đây là lớp thể hiện cho toàn bộ hệ thống chương trình, bao gồm nhiều thành phần con là các con trỏ của các lớp nhỏ hơn.
- Lớp Type1FuzzyLogicSystem: lớp hệ logic mờ loại I, là lớp có các phương thức thực thi toàn bộ các quá trình trong pha 1 của hệ thống.
- Lớp Type1FuzzySet: lớp tập mờ loại I, trong này định nghĩa các thành phần của tập mờ tam giác loại I như nút trái, nút phải, đỉnh.
- Lớp T1FSRuleSet: lớp lưu trữ các luật của hệ logic mờ loại I, chính là các luật với tập mờ tam giác loại I.
- Lớp HedgeAlgebraicFuzzyLogicSystem: lớp hệ logic mờ loại II đại số gia tử, là lớp có các phương thức thực thi toàn bộ các quá trình trong pha 2 của hệ thống.



- Lớp HedgeAlgebraic: lớp đại số gia tử, lớp đặc tả toàn bộ các đặc tính của cấu trúc đại số gia tử như  $\text{fm}(\text{True})$ ,  $\text{fm}(\text{False})$ ,  $\mu(h_i)$  với  $h_i$  là gia tử, lớp này có nhiệm vụ xây dựng cây đại số gia tử.
- Lớp HaT2FSRuleSet: lớp luật của hệ logic mờ loại II đại số gia tử.
- Lớp LinguisticTRUTH: lớp ngôn ngữ chân lý TRUTH.

#### 4.2.3. Một số lớp chính của hệ thống

**Lớp Dataset:** cung cấp các việc cài đặt cho bộ dữ liệu đầu vào của hệ thống. Lớp này chứa các thuộc tính numProperty (số thuộc tính), numClass (số lớp), filepath (đường dẫn đến file dữ liệu), enum Problem (tên bộ dữ liệu, được khai báo trong enum), phương thức standardize (chuẩn hoá).

Dataset
numProperty: int numClass: int filePath: string nameProblem: enum PROBLEM
standardize(): void

Lớp này có thể được kế thừa bởi hai lớp con TrainingData và TestData.

**Lớp System:** lớp đặc trưng cho toàn bộ hệ thống, tương tác với giao diện chương trình. lớp này có các thành phần là các con trỏ của các lớp nhỏ hơn: systemTraining (hệ thống huấn luyện), systemTest (hệ thống test), hai phương thức là training (gọi thực thi các phương thức của systemTraining) và test (gọi thực thi các phương thức của systemTest).

System
systemTraining: pointer of SystemTraining systemTest: pointer of SystemTest
training(): void test(): void

**Lớp SystemTraining:** thực thi việc training của hệ thống, với các thành phần là các con trỏ của các lớp nhỏ hơn: trainingData (dữ liệu Training), t1fls (hệ logic mờ loại I), hat2fls (hệ logic mờ loại II đại số gia tử), các phương thức là inputTraining (nhập dữ liệu Training), startTraining (thực thi quá trình training).

SystemTraining
trainingData: pointer of TrainingData t1fls: pointer of Type1FuzzyLogicSystem



hat2fls: pointer of HaT2FLS
inputTraining(): void
startTraining(): void

**Lớp Type1FuzzyLogicSystem:** lớp mô hình hệ logic mờ loại I, chứa các thành phần centerpoint (tâm cụm dữ liệu), t1fuzzysset (tập các tập mờ tam giác loại I), t1ruleset (tập các luật), các phương thức FuzzyCmean (phân cụm dữ liệu mờ), constructRule (xây dựng luật), infer (suy diễn)

Type1FuzzyLogicSystem
centerpoint: float[]
t1fuzzysset: Type1FuzzySet[]
t1ruleset: T1FSRuleSet[]
FuzzyCmean(): void
constructRule(): void
infer(): void

**Lớp Type1FuzzySet:** lớp xây dựng cấu trúc tập mờ tam giác, chứa các thành phần left (phần tử nút trái), right (phần tử nút phải), center (phần tử đỉnh tam giác), phương thức là calculate\_u (tính độ thuộc của một phần tử)

Type1FuzzySet
left: float
right: float
center: float
calculate_u(float): float

**Lớp T1FSRuleSet:** lớp xây dựng cấu trúc tập luật cho hệ logic mờ loại I, chứa các thành phần valueAttribute (mảng các thuộc tính đầu vào), valueClass (tên lớp đầu ra), valueRule (trọng số của luật), phương thức là calculate\_burn (đốt cháy dữ liệu từ đầu vào)

T1FSRuleSet
valueAttribute: int[]
valueClass: int[]
valueRule: float
calculate_burn(float[]): float

**Lớp HedgeAlgebraicFuzzyLogicSystem:** lớp xây dựng mô hình hệ logic mờ loại II đại số gia tử, gồm các thành phần hat2fsRuleSet (tập các luật của hệ logic mờ loại II đại số gia tử) hedgeAlgebra (cấu trúc đại số gia tử), các phương thức là constructHA (xây dựng cấu trúc đại số gia tử), convertFromT1FS (xây dựng luật cho hệ logic mờ loại II đại số gia tử dựa trên tập luật của hệ logic mờ loại I), infer (suy diễn)

HedgeAlgebraicFuzzyLogicSystem
hat2fsRuleSet: HaT2FSRuleSet[]
hedgeAlgebra: HedgeAlgebraic
constructHA(): void
convertFromT1FS(): void
infer(): int

**Lớp LinguisticTRUTH:** lớp ngôn ngữ chân lý TRUTH, xây dựng cấu trúc một phần tử ngôn ngữ TRUTH, gồm các thành phần length (độ dài ngôn ngữ), generator (phần tử sinh), hedges[] (mảng các gia tử), sign (dấu), fm (độ đo tính mờ),  $\underline{fm}$ ,  $\overline{fm}$  (khoảng mờ ứng với giá trị ngôn ngữ),  $v$  (ánh xạ định lượng ngữ nghĩa).

LinguisticTRUTH
length: int
generator: bool
hedges[]: int
sign: bool
fm: float
$\underline{fm}$ : float
$\overline{fm}$ : float
$v$ : float

**Lớp HedgeAlgebraic:** lớp xây dựng mô hình cấu trúc đại số gia tử, gồm các thành phần fm(True), fm(False), numHedge (số lượng gia tử), u[numHedge] (mảng giá trị độ đo tính mờ của gia tử), tree (cây cấu trúc đại số gia tử), các phương thức là constructTree (xây dựng cây đại số gia tử), getHA (lấy phần tử đại số gia tử tương đương với một giá trị thực), meet (phép meet hai giá trị ngôn ngữ), join (phép join hai giá trị ngôn ngữ)

HedgeAlgebraic
fm(True): float
fm(False): float
numHedge: int
u[]: float
tree: Tree
constructTree(): void
getHA(float): LinguisticTRUTH
meet(LinguisticTRUTH,LinguisticTRUTH): LinguisticTRUTH
join(LinguisticTRUTH,LinguisticTRUTH): LinguisticTRUTH

**Lớp HaT2FSRuleSet:** lớp xây dựng tập luật cho hệ logic mờ loại II đại số gia tử, có các thành phần t1fsRuleSet (kế thừa giá trị từ T1FSRuleSet), fmFalse (giá trị fm(c-) trong đa cấu trúc đại số gia tử trên từng tập mờ), phương thức calculate\_burn (đốt cháy tập dữ liệu đầu vào)

HaT2FSRuleSet
t1fsRuleSet: T1FSRuleSet
fmFalse: float
calculate_burn(float[]): LinguisticTRUTH

### 4.3. Môi trường và công cụ phát triển

Môi trường .Net và ngôn ngữ lập trình Visual C++ được lựa chọn để cài đặt chương trình. Ngôn ngữ C++ trên môi trường .Net có nhiều ưu điểm:

- Phát triển ứng dụng nhanh.
- C++ là ngôn ngữ mạnh với ưu điểm con trỏ (sử dụng con trỏ tối ưu trong thiết kế)
- Tốc độ chương trình chạy nhanh.
- Quản lý tài nguyên an toàn.

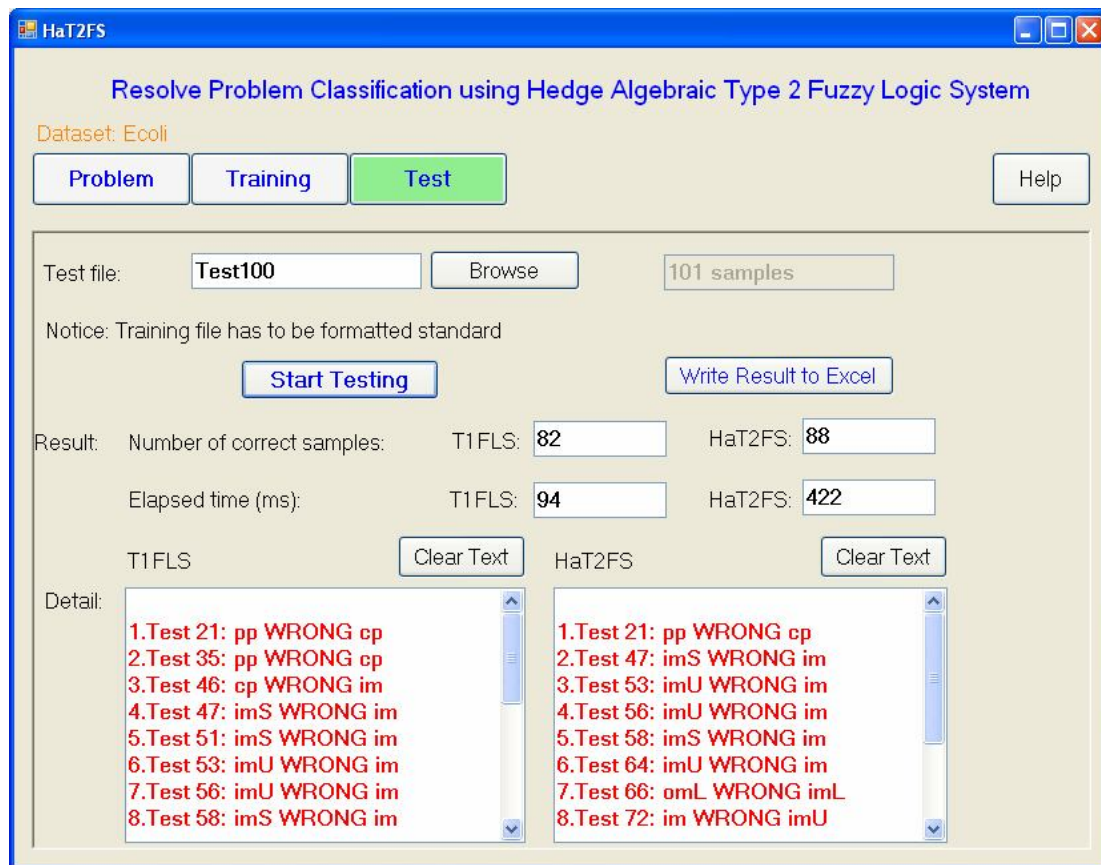
Mã nguồn chương trình được cài đặt trên bộ công cụ Visual Studio 2008 và bản đóng gói chương trình được chạy trên nền .Net Framework 3.0.

## 4.4. Chương trình

### 4.4.1. Giao diện chính

Giao diện chính của chương trình được chia làm 2 phần chính:

- Hệ thống các tab chức năng (menu)
- Vùng hiển thị thông tin.



Hình 26. Giao diện chính

### 4.4.2. Hệ thống tab chức năng của chương trình

Bao gồm các tab tương ứng với sơ đồ phân cấp chức năng.

- Problem: chọn bộ dữ liệu cho bài toán.
- Training: thực hiện chức năng huấn luyện hệ thống, đồng thời hiển thị kết quả training.
- Test: thực hiện kiểm thử hệ thống, đồng thời hiển thị kết quả test.

#### 4.4.2.1 Tab Problem

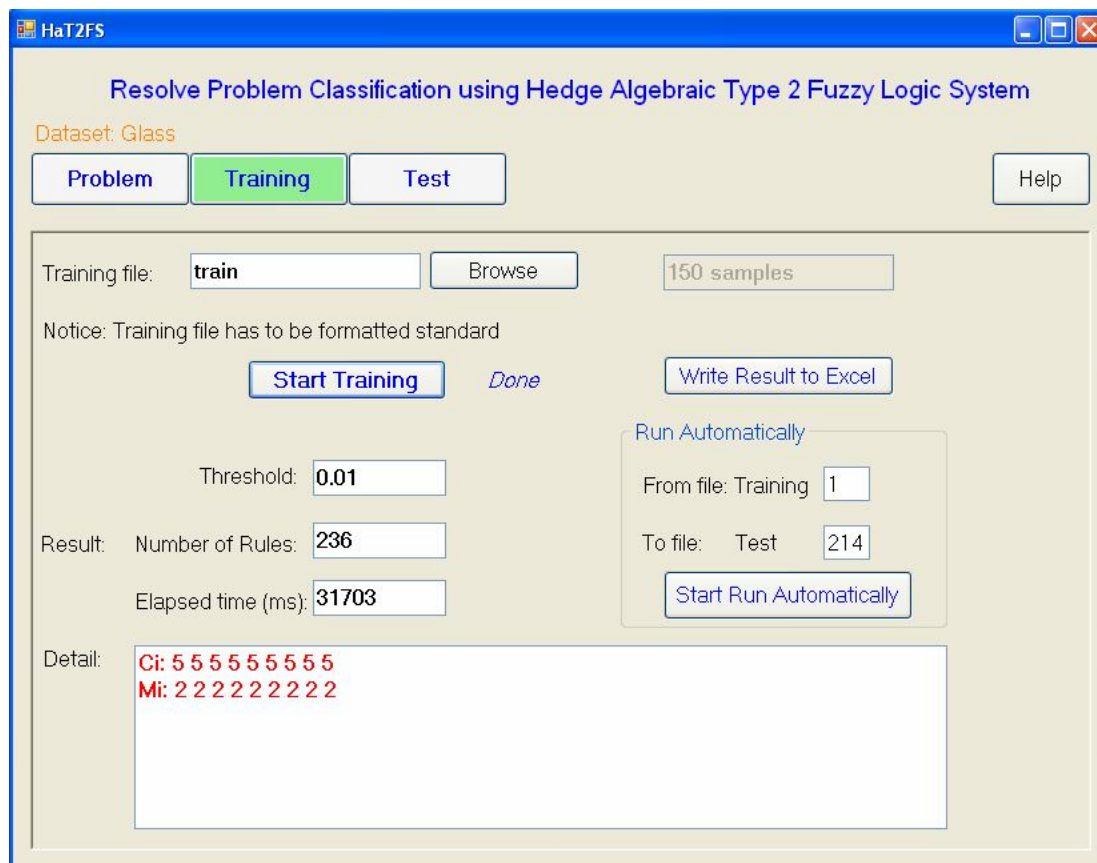
Chọn bộ dữ liệu cho bài toán, đồng thời nhập vào thông số của bộ dữ liệu (số thuộc tính, số lớp, miền dữ liệu của từng thuộc tính)

	Min	Max
Property 1:	1.5112	1.5339
Property 2:	10.73	17.38
Property 3:	0	4.49
Property 4:	0.29	3.5
Property 5:	69.81	75.41
Property 6:	0	6.21
Property 7:	5.43	16.19
Property 8:	0	3.15
Property 9:	0	0.51

Hình 27. Giao diện tab problem.

#### 4.4.2.2. Tab Training

Thực hiện quá trình huấn luyện hệ thống (người dùng browse đến file dữ liệu và nhấn nút Start Training), sau đó hiển thị kết quả training.



Hình 28: Giao diện tab Training.

#### 4.4.2.3. Tab Test

Chức năng kiểm thử một bộ dữ liệu nhập vào từ người dùng và hiển thị kết quả.

HaT2FS

Resolve Problem Classification using Hedge Algebraic Type 2 Fuzzy Logic System

Dataset: Glass

Problem Training Test Help

Test file: test Browse 64 samples

Notice: Training file has to be formatted standard

Start Testing Write Result to Excel

Result: Number of correct samples: T1 FLS: 44 HaT2FS: 48

Elapsed time (ms): T1 FLS: 108 HaT2FS: 172

T1 FLS Clear Text HaT2FS Clear Text

Detail:

T1 FLS:

- 1.Test 11: 2 WRONG 1
- 2.Test 12: 2 WRONG 1
- 3.Test 14: 2 WRONG 1
- 4.Test 17: 2 WRONG 1
- 5.Test 18: 2 WRONG 1
- 6.Test 28: 3 WRONG 2
- 7.Test 29: 1 WRONG 2
- 8.Test 35: 6 WRONG 2

HaT2FS:

- 1.Test 11: 2 WRONG 1
- 2.Test 12: 3 WRONG 1
- 3.Test 17: 2 WRONG 1
- 4.Test 18: 2 WRONG 1
- 5.Test 28: 3 WRONG 2
- 6.Test 35: 6 WRONG 2
- 7.Test 40: 1 WRONG 2
- 8.Test 41: 1 WRONG 3

Hình 29. Giao diện tab Test.

## 4.5. Kết luận

Chương này đã trình bày chi tiết quá trình thiết kế lớp trong chương trình. Giao diện chương trình cũng được thiết kế sao cho người dùng dễ thao tác nhất và cung cấp nhiều thông tin nhất trong quá trình thử nghiệm. Phần tiếp theo trình bày kết quả thực nghiệm bài toán.



## CHƯƠNG 5

# ĐÁNH GIÁ KẾT QUẢ VÀ BÀN LUẬN

### 5.1. Dữ liệu và cách kích bản thử nghiệm

Bộ dữ liệu Glass [4] gồm 214 mẫu, 9 thuộc tính, 6 lớp.

Bộ dữ liệu Ecoli [5] gồm 336 mẫu, 7 thuộc tính, 8 lớp.

Các kích bản thử nghiệm phổ biến :

*10-fold cross validation* (ký hiệu 10-fold): Dữ liệu được chia ngẫu nhiên thành 10 phần, lấy 9 phần training, 1 phần test, lặp lại phần test với cả 10 phần.

*5-fold cross validation* (ký hiệu 5-fold): Dữ liệu được chia ngẫu nhiên thành 5 phần, 4 phần training, 1 phần test lặp lại với cả 5 phần.

*2-fold cross validation* (ký hiệu 2-fold): Dữ liệu được chia ngẫu nhiên thành 2 phần, 1 phần training, 1 phần test lặp lại với cả 2 phần.

*Leave One Out*: Dữ liệu có tổng cộng N mẫu, lấy N-1 mẫu làm training, 1 mẫu làm test. Lặp lại N lần với N mẫu test.

*Full Training, Full Test*: Toàn bộ dữ liệu được lấy làm training, sau đó test cũng chính toàn bộ dữ liệu đó.

Các phương pháp khác được chọn để so sánh:

- Hierarchical Fuzzy Rule [8], 2010.
- Degree of ignorance and lateral position [7], 2011.
- Supervised and Multivariate Discretization [6], 2010.
- Novel Fuzzy Decision Tree [9], 2009.
- Multiobjective Genetic Fuzzy Rule [10], 2009.
- Với phương pháp của đồ án, ký hiệu là HaT2FLS.
- Với phương pháp tập mờ loại I, ký hiệu là T1FLS.

### 5.2. Các kết quả thử nghiệm

Toàn bộ các kích bản đều được thử nghiệm ít nhất 3 lần độc lập. Ký hiệu TestAc là Test Accuracy, là độ chính xác phân lớp của bộ dữ liệu Test. Ký hiệu TrainAc là Training Accuracy, là độ chính xác phân lớp chính bộ Training. Các kết quả đều dưới dạng phần trăm (%).

Thời gian chạy với bộ dữ liệu Glass là khoảng 18s - 30s tùy từng kích bản.



Kết quả với bộ dữ liệu Glass như sau:

Phương pháp	TestAc	TrainAc	Kịch bản
Hierarchical Fuzzy Rule	61.73	82.05	10-fold
Multiobjective Genetic Fuzzy Rule	68.29	83.97	10-fold
T1FLS	66.67	86.59	10-fold
HaT2FLS (đề tài)	70.41	95.97	10-fold
Novel Fuzzy Decision Tree	72.43	72.43	Full Train Full Test
T1FLS	84.58	84.58	Full Train Full Test
HaT2FLS (đề tài)	94.86	94.86	Full Train Full Test
Novel Fuzzy Decision Tree	67.19	-	Leave One Out
T1FLS	62.62	-	Leave One Out
HaT2FLS (đề tài)	64.95	-	Leave One Out
Degree of ignorance and lateral position	61.21	77.79	5-fold
T1FLS	65.03	88.61	5-fold
HaT2FLS (đề tài)	69.16	95.55	5-fold
Supervised and Multivariate Discretization	68.22	-	2-fold
T1FLS	66.12	89.72	2-fold
HaT2FLS (đề tài)	70.79	97.43	2-fold
T1FLS	68.75	90.00	<a href="http://www.mlcomp.org">www.mlcomp.org</a>
HaT2FLS (đề tài)	75.00	98.00	<a href="http://www.mlcomp.org">www.mlcomp.org</a>

Bảng 4. Kết quả thử nghiệm bộ dữ liệu Glass

Riêng với dữ liệu Glass chuẩn từ [www.mlcomp.org](http://www.mlcomp.org), TrainingData có 150 mẫu, TestData có 64 mẫu, kết quả với T1FLS là 44/64 với TestAc, 135/150 với TrainAc, kết quả với HaT2FLS là 48/64 với TestAc, 147/150 với TrainAc.

Thời gian chạy với bộ dữ liệu Ecoli là khoảng 1s - 2s tùy từng kịch bản. Kết quả với bộ dữ liệu Ecoli như sau:

Phương pháp	TestAc	TrainAc	Kịch bản
Hierarchical Fuzzy Rule	87.19	93.26	10-fold
T1FLS	79.94	91.45	10-fold
HaT2FLS (đề tài)	82,32	99.50	10-fold
Degree of ignorance and lateral position	79.76	85.59	5-fold
T1FLS	80.90	91.74	5-fold
HaT2FLS (đề tài)	82.89	99.52	5-fold
T1FLS	90.77	90.77	Full Train Full Test
HaT2FLS (đề tài)	99.40	99.40	Full Train Full Test
T1FLS	80.65	91.20	Leave One Out
HaT2FLS (đề tài)	82.44	99.44	Leave One Out

Bảng 5. Kết quả thử nghiệm bộ dữ liệu Ecoli

### **5.3. Đánh giá kết quả**

Nhìn chung tất cả các kịch bản, phương pháp hệ logic mờ loại II đại số gia tử đạt được kết quả cải tiến khá đáng kể so với phương pháp hệ logic mờ loại I và một số phương pháp khác. Một điều đặc biệt nữa là độ chính xác Training (Training Accuracy) rất cao, với bộ dữ liệu Glass thường đạt được khoảng 95%, với bộ dữ liệu Ecoli đạt được hơn 99%, chính xác hơn nhiều so với hệ logic mờ loại I, điều này đạt được bởi sự cải tiến xây dựng hệ logic mờ loại II đại số gia tử dựa vào sai số của hệ logic mờ loại I. Kết quả này hoàn toàn có thể làm tốt hơn, vì với cả hai bộ dữ liệu ngưỡng chọn để loại bỏ luật là 0.01. Dựa vào từng đặc điểm của các bộ dữ liệu để chọn ngưỡng loại bỏ luật tương ứng sẽ đưa đến kết quả tốt hơn.

## CHƯƠNG 6

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 6.1. Kết luận

Bài toán phân lớp đã có nhiều hướng giải quyết của nhiều nhà khoa học trên thế giới. Hầu hết các hướng giải quyết đều cố gắng tìm ra những đặc trưng nào đó trong quá trình xử lý dữ liệu chứa nhiều có nhiễu. Đồ án đã xây dựng một phương pháp giải quyết bài toán phân lớp dựa trên mô hình hệ logic mờ loại II đại số gia tử, một phương pháp đã áp dụng thành công vào một vài bài toán xử lý thông tin không chắc chắn. Mô hình phương pháp đồ án đề xuất cũng có nhiều cải tiến so với các nghiên cứu trước đó.

Đầu tiên, đồ án đã xây dựng hoàn chỉnh hệ logic mờ loại I từ dữ liệu TrainingData đầu vào. Quá trình này gồm thuật toán phân cụm mờ Fuzzy C-mean với đầu ra là các tâm cụm, xây dựng tập mờ tam giác dựa trên các tâm cụm, xác định tập luật dựa trên quá trình sinh luật và đốt cháy trên tất cả các bản ghi, thực hiện suy diễn trên hệ logic mờ loại I và ghi nhận sai số. Hệ logic mờ loại I được xây dựng hoàn chỉnh là tiền đề để xây dựng hệ logic mờ loại II đại số gia tử.

Tiếp theo, từ tập luật và sai số của hệ logic mờ loại I cùng với quá trình xây dựng cây đại số gia tử, toàn bộ luật được chuyển từ tập mờ loại I thành tập mờ loại II đại số gia tử. Quá trình suy diễn với đầu vào là dữ liệu Test (TestData) được thực hiện trên hệ logic mờ loại II đại số gia tử qua các phép toán trên giá trị ngôn ngữ là phép meet thu được kết quả phân lớp.

Kết quả thử nghiệm trên hai bộ dữ liệu của UCI Machine Learning Repository và so sánh với nhiều phương pháp khác trên thế giới vài năm trở lại đây cho thấy phương pháp của đồ án có những kết quả rất khả quan. Mặc dù không phải kết quả thử nghiệm nào cũng tốt hơn các phương pháp khác nhưng hầu hết các kết quả tốt hơn cho thấy khả năng ứng dụng hệ logic mờ loại II đại số gia tử trong khai phá dữ liệu mà điển hình là bài toán phân lớp rất hiệu quả.

Mô hình hệ logic mờ loại II đại số gia tử đồ án đã trình bày vẫn còn tồn tại một số vấn đề sau :

- Thứ nhất, đầu vào cho thuật toán phân cụm mờ Fuzzy C-mean là khởi tạo trước số cụm và tham số mờ  $m$ , điều này vẫn còn đang được nghiên cứu trên thế giới. Trong các nghiên cứu trước đây về hệ logic mờ loại II đại số gia tử, thuật giải di truyền được áp dụng để tối ưu tham số hệ thống. Tuy nhiên, việc cài đặt thuật giải di truyền chạy thời gian tương đối lâu, nên tác giả đồ án chọn tham số mờ  $m$  được khởi tạo theo Bezdek [2], và tham số số cụm được khởi là 5.

- Thứ hai, ngưỡng loại bỏ luật được chọn là 0.01 đối với cả hai bộ dữ liệu và với mọi kịch bản dữ liệu đầu vào. Điều này chưa hẳn đã tốt và đôi khi bộ luật thu được chưa phải là bộ luật tối ưu tốt nhất. Điều này gây ra một vài lần thử nghiệm kết quả thu được chưa thật sự tốt.
- Thứ ba, toàn bộ sai số của hệ logic mờ loại I được lấy làm cơ sở để xây dựng đa cấu trúc đại số gia tử trên các tập mờ. Điều này đã cải thiện một cách đáng kể kết quả kiểm thử chính bộ Training (95% với bộ dữ liệu Glass và hơn 99% với bộ dữ liệu Ecoli) và cải thiện khoảng 2 - 6% với TestData. Điều này dường như có thể đã quá khớp. Nguyên nhân do toàn bộ sai số được sử dụng để thay đổi toàn bộ các tập mờ. Hướng cải thiện vấn đề này được trình bày dưới đây.

## **6.2. Hướng phát triển trong tương lai**

Trong tương lai các hướng phát triển tập trung vào những vấn đề còn tồn tại ở trên. Các hướng phát triển như sau:

- Phát hiện ra tính chất thống kê của bộ dữ liệu. Trong phần tiền xử lý dữ liệu, một phương pháp phân tích thống kê nào đó đưa ra một đặc điểm về dữ liệu, dựa trên cơ sở đó khởi tạo đầu vào là số cụm cho giải thuật phân cụm mờ Fuzzy C-mean. Đạt được điều này thì các tập mờ tam giác trong hệ thống thu được sẽ là rất tốt.
- Một phương pháp xác định luật nhanh hơn là phương pháp duyệt qua tất cả các luật có thể rồi tính trọng số. Việc duyệt qua tất cả các luật có thể sẽ rất khó khăn khi số thuộc tính tăng lên. Thêm nữa là ngưỡng loại bỏ luật được chọn tốt hơn dựa trên tham số thống kê nào đó của bộ dữ liệu, khi đó bộ luật thu được sẽ tốt và có ý nghĩa.
- Lấy một phần thay vì toàn bộ các sai số của hệ logic mờ loại I để xây dựng cấu trúc đại số gia tử trên tập mờ. Phần còn lại sẽ được test lại một lần nữa trong quá trình suy diễn trên hệ logic mờ loại II đại số gia tử. Hướng phát triển này sẽ làm giảm sự quá khớp với dữ liệu Training và đồng thời đưa vào quá trình suy diễn của hệ logic mờ loại II đại số gia tử ngay trong quá trình training. Hướng phát triển này hứa hẹn một kết quả ý nghĩa tốt hơn nhiều trong tương lai.

## TỔNG KẾT

Với ưu điểm xử lý thông tin không chắc chắn, mô hình giải bài toán phân lớp sử dụng hệ logic mờ loại II đại số gia tử đã thể hiện một phương pháp tiếp cận mới dựa trên suy diễn với các giá trị ngữ nghĩa. Với các kết quả đạt được, mô hình hệ logic mờ loại II đại số gia tử thể hiện ưu điểm trong xử lý thông tin và có thể được áp dụng trong các bài toán dự báo, điều khiển tự động và ra quyết định. Bài toán phân lớp là một kiểu bài toán dự báo đã cho thấy kết quả rất khả quan.

Với 6 chương, đồ án đã trình bày chi tiết mô hình phương pháp và các kết quả đạt được. Chương 1 nêu ra bài toán cần giải quyết và giới thiệu phương pháp. Chương 2 trình bày cơ sở lý thuyết bao gồm các khái niệm cơ bản về tập mờ và hệ logic mờ loại II đại số gia tử. Chương 3 trình bày phương pháp giải bài toán phân lớp sử dụng hệ logic mờ loại II đại số gia tử và chi tiết từng pha, ý nghĩa từng quá trình. Chương 4 trình bày phân tích thiết kế chương trình và chương 5 trình bày kết quả thực nghiệm với những bộ dữ liệu chuẩn. Qua những kết quả đạt được, kết luận về phương pháp và hướng phát triển được trình bày trong chương 6.

Đồ án đã nêu bật được phương pháp sử dụng hệ logic mờ loại II đại số gia tử để giải bài toán phân lớp và ý nghĩa phương pháp. Một số vấn đề còn có thể phát triển thêm của đồ án, tác giả đồ án rất mong được các thầy cô, các bạn sinh viên và các bạn đọc tiếp tục phát triển phương pháp để đạt được những kết quả tốt hơn.

## TÀI LIỆU THAM KHẢO

- [1] Tran Dinh Khang, Dinh Khac Dong, Phan Anh Phong, The hedge algebraic type-2 fuzzy sets, submitted to Elsevier, 2011.
- [2] Mohanad Alata, Mohammad Molhim, and Abdullah Ramini, *Optimizing of Fuzzy C-Means Clustering Algorithm Using GA*, World Academy of Science, Engineering and Technology, 2008.
- [3] G. J. Klir and T. A. Folger, Fuzzy Sets, Uncertainty, and Information. Prentice-Hall International, 1992.
- [4] <http://archive.ics.uci.edu/ml/datasets/Glass+Identification>
- [5] <http://archive.ics.uci.edu/ml/datasets/Ecoli>
- [6] Feng Jiang, ZhixiZhao and YanGe1, A Supervised and Multivariate Discretization Algorithm for Rough Sets, Springer-Verlag Berlin Heidelberg, 2010.
- [7] J. Sanz, A. Fernández, H. Bustince, F.Herrera, A genetic tuning to improve the performance of Fuzzy Rule-Based Classification Systems with Interval-Valued Fuzzy Sets: Degree of ignorance and lateral position, 2011 Elsevier.
- [8] A. Fernandez, M. J. del Jesus, F. Herrera, Analysing the Hierarchical Fuzzy Rule Based Classification Systems with Genetic Rule Selection, 2010 IEEE.
- [9] E. Ahmadi, M. Taheri, N. Mirshekari, Sattar Hashemi, Ashkan Sami, Ali Hamze, Cooperative Fuzzy Rulebase Construction based on a Novel Fuzzy Decision Tree, 2009 IEEE.
- [10] Rafael Alcala, Yusuke Nojima, Francisco Herrera, Hisao Ishibuchi, Generating Single Granularity-Based Fuzzy Classification Rules for Multiobjective Genetic Fuzzy Rule Selection, 2009 IEEE.
- [11] Hồ Thuần, Đặng Thanh Hà, Logic mờ và ứng dụng, Nhà xuất bản Đại Học Quốc Gia Hà Nội, 2007.