

République Algérienne Démocratique et Populaire
UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE
HOUARI BOUMEDIENNE



FACULTE DES SCIENCES MATHÉMATIQUES
PURES ET APPLIQUÉES

THESE

Présentée pour l'obtention du grade de :

Magister En Mathématiques

Spécialité : Analyse (Systèmes Dynamiques).

Par

HACHAMA MOHAMED

SUJET

**La Logique Floue : Théorie et Application à
la Modélisation des Systèmes Dynamiques**

Soutenu publiquement le : 05/03/2003, devant le jury composé de :

Mr. A. KESSI	Professeur (USTHB)	Président
Mr. R. BEBBOUCHI	Professeur (USTHB)	Directeur de la Thèse
Mr. M. AIDER	Maître de Conférences (USTHB)	Examineur
Mr. M. S. MAAMRA	Chargé de recherche (USTHB)	Examineur
Mr. K. BOUKHELATA	Maître de Conférences (USTHB)	Examineur

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et des Technologies de Bab el-zouar
des Sciences
Département des Mathématiques

جامعة البليدة Faculté
كلية الهندسة
دائرة الكيمياء الصناعية

MEMOIRE DE MAGISTER

Présenté par : **Mohamed HACHAMA**

En Vue de l'Obtention du Diplôme de Magister en Mathématiques
Option : Systèmes dynamiques

THEME

*La Logique Floue : Théorie et Application à la
Modélisation des Systèmes Dynamiques.*

Soutenu publiquement le 05 mars 2003 devant le Jury:

Président:	Mr. A. Kessy	Professeur (USTHB)
Examineurs:	Mr. B. MAAMRA Mr. M. AIDER	Maître de Conférences (USTHB) Maître de conférences (USTHB)
Rapporteur:	Mr. R. BEBBOUCHI	Professeur (USTHB)

Route de Soumaâ. B.P 270 09000 Blida (Algérie)

Tel/Fax : 025 43 36 31

DEDICACES

A mes parents

A mes frères et sœurs.

A ma grande famille

A mes amis

A tous ceux qui m'ont aidé, de près ou de loin .

REMERCIEMENTS

J'adresse mes sincères remerciements à Monsieur A. KESSI, Professeur à l'USTHB, qui m'a fait l'honneur de présider le jury de thèse, ainsi qu'à Messieurs M. S. MAAMRA, M. AIDER, et K. BOUKHETALA qui ont accepté d'en faire partie.

Je tiens aussi à exprimer ma profonde gratitude et mes remerciements les plus sincères à Monsieur R. BEBBOUCHI, Professeur à l'USTHB, pour m'avoir encadré, et pour la confiance qu'il m'a toujours témoignée. Son expérience de la recherche, ses conseils, ses encouragements et sa disponibilité m'ont été très précieux et m'ont permis de mener à bien ce travail.

Je remercie également Monsieur A. KHOUKHI, qui n'a pas ménagé ses efforts pour m'apporter son soutien et m'a aimablement assisté tout au long de mon travail.

Je ne saurais oublier de remercier vivement Madame et Monsieur MOUZALI pour m'avoir soutenu dans les moments difficiles. Qu'ils trouvent ici toutes les expressions de gratitude et de reconnaissance, pour leur dévouement et pour leur amitié éternelle.

Mes sincères remerciements vont aussi à tous ceux qui m'ont aidé, de près ou de loin. Je pense particulièrement à : Messieurs D. DUBOIS et H. PRADE, Madame B. B. MEUNIER, Monsieur J. JANTZEN, et Monsieur M. HADJILI. Ces chercheurs, que je n'ai pas eu le plaisir de rencontrer, ont contribué de façon efficace dans la réalisation de ce travail.

J'associe enfin à ces remerciements tous les membres du Département des Mathématiques, ma famille, ainsi que tous mes amis, avec une pensée particulière pour K. HACHAMA, M. RAHMANI, M. KADARI, M. IEKHLEF, K. SAHNOUNE, Y. KAIS, H. AMRI, F. REDAOUIA, ainsi que M. BELLALI, dont je garde un souvenir impérissable.

Table des Matières

Introduction générale 1

PARTIE I : LA LOGIQUE FLOUE

Chapitre I Théorie des Ensembles Flous

1. Introduction

- 1.1. Un peu d'histoire 4
- 1.2. Soyons logiques 5
- 1.3. Position de la théorie des ensembles flous 5

2. Les ensembles flous

- 2.1. Définitions 5
- 2.2. Opérations sur les sous-ensembles flous 8
- 2.3. Ensembles ordinaires associés à un sous-ensemble flou 9
- 2.4. Sous-ensemble flou convexe 11
- 2.5. Produit cartésien et projection de sous-ensembles flous 12
- 2.6. Principe d'extension 13
- 2.7. Spécificité et précision de sous-ensembles flous 14
- 2.8. Normes et conormes triangulaires 16

3. Relations et quantités floues

- 3.1. Relations floues 19
- 3.2. Quantités floues 22

Chapitre II Théorie des Possibilités et Raisonnement Approximatif

1. La théorie des possibilités

- 1.1. Mesures et distributions des possibilités 24
- 1.2. Dualité entre mesure de possibilité et de nécessité 26
- 1.3. Comparaison entre probabilité et possibilité, théorie de l'évidence 27
- 1.4. La théorie des ensembles flous comme base de celle des Probabilités 30
- 1.5. Principes possibilistes pour le Raisonnement Approximatif 30

2. Variables linguistiques et propositions floues

- 2.1. Variables linguistiques 32
- 2.2. Propositions floues 33

3. Raisonnement Approximatif

- 3.1. Implications floues 38
- 3.2. Modus Ponens généralisé 40
- 3.3. Comportement des implications floues 42

Chapitre III Mathématiques floues

1. Arithmétique floue :	
1.1. Quantités floues, intervalles flous, nombres flous	45
1.2. Opérations arithmétiques	46
1.2.1. Opérations unaires	46
1.2.2. Opérations binaires	47
1.3. Calcul pratique des quantités floues non - interactives	48
1.3.1. Représentation paramétrique	48
1.3.2. Utilisation des α - coupes	50
1.4. Comparaison des quantités floues	51
1.5. Distance entre nombres flous	52
1.6. Applications	52
2. Mesures et intégrales floues	
2.1. Mesures floues	53
2.2. Intégrales floues	54
2.2.1. Intégrale floue de Choquet	54
2.2.2. Intégrale floue de Sugeno	55
2.3. Complément : Probabilités d'événements flous	56
2.3.1. La probabilité d'un ensemble flou comme un scalaire	56
2.3.2. Probabilité d'un événement flou comme un ensemble flou	57
2.4. Applications	57
3. Programmation mathématique floue	
3.1. Programmation Linéaire Floue	58
3.1.1. PLF dans un environnement flou	59
3.1.2. PLF avec des coefficients flous	62
3.2. Programmation mathématique floue	63

PARTIE II : LA MODELISATION FLOUE

Chapitre IV Les systèmes flous

1. Introduction	64
2. La modélisation floue à base de connaissances	65
2.1. Structure interne d'un système d'inférence flou	66
2.1.1. La fuzzification	66
2.1.2. L'inférence	67
2.1.3. La défuzzification	68
2.2. Différents types des systèmes flous	70
2.2.1. Les modèles flous linguistiques	70
2.2.2. Modèles flous de Takagi-Sugeno (TSK)	73
2.2.3. Comparaison	76

3. Modélisation et identification des systèmes dynamiques	76
3.1 Approximation des fonctions statiques	76
3.2 La modélisation des systèmes dynamiques.....	77
3.2.1 Formulation discrète entrées-sorties	77
3.2.2 Formulation de l'espace d'état	78
4. Conclusion	80

Chapitre V Identification des systèmes flous

1. Introduction	81
2. Méthodes ad hoc	81
2.1. Méthode de Wang et Mendel	82
2.2. Avantages et inconvénients	83
3. Les méthodes basées sur la coalescence des données	83
3.1 Dérivation du modèle linguistique par coalition	84
3.2 Dérivation du modèle de TSK par coalescence	84
3.3 Algorithmes de coalescence	85
4. Méthodes neuro-floues	86
5. Méthodes génétiques-floues	88
6. Simulations	88
6.1. Approximation des fonctions statiques	89
6.2. Prédiction des séries chaotiques	92
7. Conclusion	95

Chapitre VI Application à la commande

1. Introduction	96
2. Les commandes PID floues	97
3. Commande floue multimodèle	99
4. Application pratique	100
5. Conclusion	105

CONCLUSION GENERALE..... 106

Annexe 1 Réseaux de Neurones	109
---	-----

Annexe 2 Algorithmes Génétiques	114
--	-----

Références Bibliographiques

Introduction Générale

" La précision n'est pas la vérité."

Henri Matisse

La modélisation est devenue une issue importante dans l'ingénierie et la science. Les approches traditionnelles de modélisation insistaient énormément sur la précision et la description exacte des systèmes. L'utilisation des outils mathématiques comme les équations différentielles ou aux différences, fonctions de transfert, ...etc. est appropriée et justifiée pour les systèmes bien définis. Mais, quand la complexité augmente, ces outils deviennent moins efficaces.

La complexité d'un système peut être caractérisée par plusieurs facteurs : le nombre d'éléments qui le composent, les relations entre ces éléments, leurs caractéristiques, la dynamique du système, ...etc. Lorsque la complexité augmente, les méthodes de modélisation traditionnelles s'avèrent souvent incapables de représenter le comportement global du système. L'impuissance des techniques quantitatives traditionnelles pour décrire correctement les systèmes complexes a été exprimée par Zadeh sous la forme du principe d'incompatibilité : *" Lorsque la complexité d'un système augmente, notre capacité à faire une description précise et significative de son comportement diminue jusqu'à un seuil au-delà duquel précision et signification (pertinence) deviennent des caractéristiques mutuellement exclusives "*.

Le traitement des systèmes complexes nécessite souvent la manipulation d'informations vagues, imprécises (il mesure entre 1.70m et 1.80m), incertaines (il est probable qu'il mesure 1.80m) ou à la fois imprécises et incertaines (il est probable qu'il mesure entre 1.70m et 1.80m). L'être humain est compétent dans la manipulation de tels systèmes de façon naturelle. Au lieu de raisonner en termes mathématiques, il décrit le comportement du système par des propositions linguistiques. Par exemple, un conducteur peut formuler une partie de sa connaissance par : *" Si la vitesse est élevée et qu'il y a un virage, alors freiner"*.

Afin de pouvoir représenter ce type d'informations, Zadeh a proposé de modéliser le mécanisme de la pensée humaine par un raisonnement approximatif basé sur des variables linguistiques. Dans le cas d'un système complexe, étant donné le grand volume d'informations nécessaires à sa description, la possibilité de résumer ces informations joue un rôle essentiel. Dans le cas de l'homme, la capacité de résumer se manifeste essentiellement dans l'utilisation de la langue naturelle. Un modèle linguistique formulé par un expert permet alors une représentation simplifiée du système traduisant la perception et la compréhension qu'en a l'expert.

La théorie des ensembles flous constitue une interface entre les mondes linguistiques et numériques. Les termes linguistiques sont représentés par des ensembles flous et donnent ainsi naissance à des concepts flous, et les règles linguistiques sont traduites en des règles floues.

La théorie des ensembles flous ou plus généralement la logique floue, est née en 1965 des travaux de Lotfi Zadeh, professeur à l'Université de Californie à Berkeley, reconnu pour ses travaux en automatique et théorie des systèmes [ZAD 65]. Dans un sens étroit, la logique floue est un système logique multivalent, qui est une extension de la logique classique bivalente, dans le but de raisonner sur des connaissances imparfaites (Raisonnement Approximatif). Mais, dans un sens plus large qui est prédominant aujourd'hui, la logique floue correspond à tous les développements issus de la théorie des ensembles flous [JAN 95].

La théorie des sous-ensembles flous et la logique floue peuvent être vues comme des théories mathématiques formelles pour la représentation des incertitudes. L'incertitude est cruciale pour la gestion des systèmes réels : si on veut garer sa voiture précisément dans une place, ca serait impossible. Par contre, on le fait avec, disons, 10 cm de tolérance. La présence de l'incertitude est le prix à payer pour pouvoir manipuler des systèmes complexes. La logique floue est une logique du flou, non pas une logique qui est elle-même floue. Au même titre que les lois des probabilités qui ne sont pas aléatoires, les lois traitant le flou ne sont pas floues. Il convient d'utiliser la logique floue lorsque des imprécisions entachent la connaissance dont nous disposons sur le système, lorsqu'une modélisation rigoureuse de celui-ci est difficile, lorsqu'il est très complexe, en exploitant des connaissances symboliques (souvent celles d'un expert), exprimées en langage naturel.

L'introduction des concepts flous dans la théorie des systèmes a impliqué la naissance d'une nouvelle classe de systèmes appelés " *systèmes flous* ". De tels systèmes sont capables de décrire des systèmes complexes en introduisant des informations vagues et en imitant le mécanisme de raisonnement approximatif utilisé chez l'homme.

Comme application, la modélisation floue est le processus par lequel un système dynamique est modélisé non dans la forme conventionnelle des équations différentielles ou aux différences, mais dans la forme d'un ensemble de règles floues et fonctions d'appartenance. C'est une approche qualitative permettant de modéliser des systèmes non linéaires, complexes.

De nos jours, les applications des systèmes flous se sont multipliées. Entre autres, on peut citer les contrôleurs flous, les modèles flous, les systèmes experts flous, les systèmes de diagnostics, de classification et de reconnaissance des formes, ...etc.

Organisation du mémoire :

Le but de ce mémoire est de montrer les bases théoriques sur lesquelles repose la logique floue, les outils qu'elle fournit ainsi que son application à la modélisation des systèmes dynamiques.

Le mémoire est subdivisé en deux parties de trois chapitres chacune.

Dans le *premier chapitre*, nous présentons la théorie des ensembles flous, sa position par rapport à la théorie classique des ensembles, ainsi que les concepts flous relatifs comme les relations floues, les quantités floues, ...etc.

Dans le *deuxième chapitre*, le raisonnement approximatif, alternative au raisonnement exacte rigide, est étudié dans le cadre de la théorie des possibilités.

Dans le *troisième chapitre*, nous exhibons quelques exemples de l'investigation de la logique floue dans les domaines des mathématiques classiques. Ainsi, nous présentons des versions floues de l'Arithmétique, la théorie des mesures et de l'intégration, et la programmation mathématique.

Dans le *quatrième chapitre*, les systèmes flous sont présentés, et, comme cas particulier, la modélisation floue des systèmes dynamiques.

Dans le *cinquième chapitre*, nous exposons les algorithmes d'identification des systèmes flous les plus connus. Des simulations sont réalisées pour tester l'efficacité de ces algorithmes, ainsi que la capacité des systèmes flous à représenter efficacement des systèmes dynamiques.

Dans le *sixième chapitre*, nous présentons une application des systèmes flous au contrôle d'un bras manipulateur.

Chapitre I

Théorie des ensembles flous

" À quoi bon soulever des montagnes quand il est si simple de passer par-dessus ?"

Boris Vian, écrivain français.

1. Introduction :

Avant d'introduire la théorie des ensembles flous, il est bon de remémorer le cheminement de la construction logique de la théorie des ensembles de Zemerlo et Fraenkel (voir [FUC 93], par exemple).

1.1. Un peu d'histoire :

La première construction de la théorie des ensembles revient à Cantor¹ (1872). La définition de Cantor concernant les ensembles est la suivante : " Un ensemble est la réunion en un tout d'objets de notre intuition ou de notre pensée, bien définis et différenciables les uns par rapport aux autres". On peut donc former des ensembles, selon cette définition, en considérant des objets bien spécifiés et tous distincts ; peu importe que ces objets soient concrets ou abstraits.

Ainsi, cette définition "intuitive" des ensembles selon Cantor conduit à des paradoxes, de type logique comme celui de Russel, émanant de la considération de l'ensemble de tous les ensembles, ou de type sémantique comme par exemple, l'énoncé : " Ce que je dis est faux " qui n'est ni vrai ni faux.

Pour éviter l'écueil de ces paradoxes, dû à l'usage incontrôlé de la langue et de l'imagination dans la construction des ensembles, on a établi d'autres fondements des mathématiques s'appuyant sur des concepts beaucoup moins intuitifs. La principale voie est celle de la logique formalisée, laquelle se ramène à l'emploi d'un langage formalisé convenable ou ce qu'on appelle " un système formel ".

¹ La théorie des ensembles telle que construite par Cantor est dite théorie naïve des ensembles.

1.2. Soyons logiques :

Un système formel ou une théorie comporte la donnée d'un langage, d'un système d'axiomes et des règles de déduction. La logique propositionnelle et celle du premier ordre (calcul des prédicats) constituent des exemples de systèmes formels. Toute théorie contenant au moins le calcul des prédicats (de nouveaux axiomes propres sont ajoutés) est appelée : Théorie du premier ordre.

La théorie formelle des ensembles construite par Zemerlo et Fraenkel au début du 20^{ème} siècle, constitue une théorie du premier ordre. Pièce angulaire de l'édifice mathématique, c'est la théorie mathématique en excellence, en ce sens qu'on a eu l'habitude de formuler toute autre théorie mathématique comme chapitre particulier [FUC 93].

1.3. Position de la théorie des ensembles flous :

Dans cette optique, on introduit la théorie des ensembles flous comme théorie mathématique basée sur la théorie des ensembles, mais qui présente l'intérêt de l'étendre, en modifiant la définition même d'un ensemble ou plus précisément celle d'un sous-ensemble car on considère toujours un ensemble ordinaire comme référence pour définir des ensembles flous.

A vrai dire, une théorie mathématique est définie comme étant un système formel contenant au moins la théorie des ensemble. Toutefois, on adoptera la définition suivante :

On appellera théorie mathématique la collection des résultats (théorèmes) relatifs à des objets bien spécifiés de la théorie des ensembles, introduits via de nouvelles définitions. En d'autres termes, une théorie mathématique est un *chapitre particulier* de la théorie des ensembles.

Ainsi, la théorie des ensembles flous est présentée comme l'ensemble des résultats relatifs à l'introduction de la notion nouvelle de sous-ensemble flou et les nouveaux concepts associés.

2. Les ensembles flous :

2.1. Définition de sous-ensembles flous :

Un ensemble flou est une classe d'objets caractérisée par une fonction qui assigne à chaque objet un degré d'appartenance à cette classe, appartenant à l'intervalle $[0,1]$.

Ainsi, plus le degré d'appartenance est proche de 1, plus l'appartenance est forte.

Le concept de sous-ensemble flou, extension de la notion de sous-ensemble classique, a été introduit [ZAD 65] pour éviter les passages brusques d'une classe à une autre (de la classe noire à la classe blanche par exemple) et autoriser des éléments à n'appartenir complètement ni à l'une ni à l'autre (à être gris, par exemple) ou encore à appartenir partiellement à chacune (avec un fort degré de la classe noire et un faible degré de la classe blanche dans le cas du gris foncé).

Les sous-ensembles flous servent donc à décrire des concepts vagues, imprécis, des propriétés graduelles ou des événements incertains. Les notions d'inclusion, de réunion, intersection, complément, relation, convexité, ...etc., sont étendues à de tels ensembles, et diverses propriétés de ces notions dans le contexte des ensembles flous sont établies.

A. Définition d'un sous-ensemble flou :

Soit X un ensemble de référence classique.

Définition 2.1.1 : Un *sous-ensemble flou* A de X est défini par une fonction d'appartenance qui associe à chaque élément x de X le degré $f_A(x)$, compris entre 0 et 1, avec lequel x appartient à A :

$$f_A : X \rightarrow [0, 1].$$

D'une façon formelle¹, on peut définir le sous-ensemble flou A par le couple (X, f_A) .

Dans le cas où f_A ne prend que des valeurs égales à 0 ou 1, le sous-ensemble flou A est un sous-ensemble classique de X . Un sous-ensemble classique est donc un cas particulier de sous-ensemble flou.

Notations : Soit A un sous-ensemble flou d'un ensemble ordinaire X . On note :

$$A = \frac{f_A(x_1)}{x_1} + \frac{f_A(x_2)}{x_2} + \dots + \frac{f_A(x_n)}{x_n} \quad \text{si } X = \{x_1, x_2, \dots, x_n\} \text{ fini,}$$

$\frac{f_A(x_i)}{x_i}$ exprime : l'élément x_i appartient à l'ensemble flou A avec un degré $f_A(x_i)$.

$f_A(x_i)$ est le degré de vérité de la proposition " $x_i \in A$ ".

Si X est dénombrable, on écrit :

$$A = \sum_{x \in X} \frac{f_A(x)}{x}.$$

¹Dans la définition d'un ensemble flou, on a préféré donner plus d'importance à l'aspect sémantique que l'aspect formel.

Dans le cas où X est continu, on écrit :

$$A = \int_X \frac{f_A(x)}{x} .$$

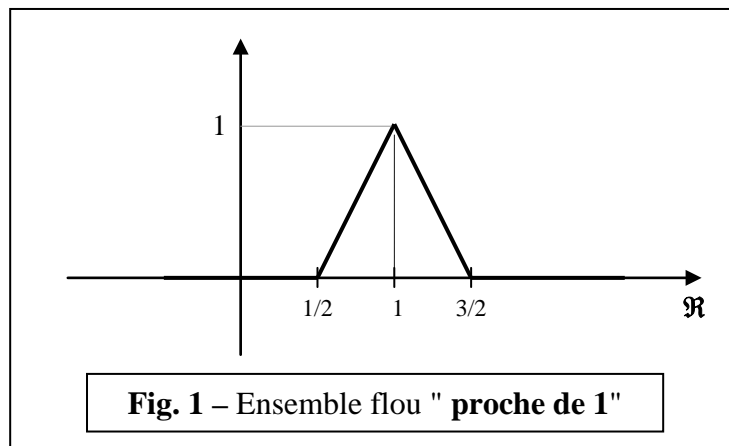
La famille de tous les sous-ensembles flous de X sera notée : $F(X)$.

On note $Prop(A)$, la proposition associée à A , permettant de définir ses éléments.

Exemple : Soit X l'ensemble des nombres réels \mathcal{R} , A un sous-ensemble flou contenant les nombres réels " proches de 1 " .

$$A = \int_{]-\infty, 1/2[} \frac{0}{x} + \int_{]1/2, 1[} \frac{2 \cdot x - 1}{x} + \int_{]1, 3/2]} \frac{-2 \cdot x + 3}{x} + \int_{]3/2, +\infty]} \frac{0}{x} ,$$

comme indiqué sur la figure 1.



B. Caractéristiques d'un sous-ensemble flou :

Définition 2.1.2 : Soit A un sous-ensemble flou d'un ensemble de référence X .

On définit pour A les caractéristiques suivantes :

1. Le support : $Supp(A) = \{x \in X / f_A(x) \neq 0\}$.

C'est l'ensemble des éléments de X qui appartiennent au moins un peu à A .

2. La hauteur : $h(A) = \sup_{x \in X} f_A(x)$.

Ca représente le plus fort degré avec lequel un élément de X peut appartenir à A .

3. Le noyau : $\text{noyau}(A) = \{x \in X / f_A(x) = 1\}$

C'est l'ensemble des éléments qui appartiennent complètement à A .

4. Normalité : A est dit normalisé si : $\exists x \in X : f_A(x) = 1$.

On utilise souvent des sous-ensembles flous normalisés, comme c'est le cas par exemple en théorie des possibilités.

2.2. Opérations sur les sous-ensembles flous :

La description d'un sous-ensemble flou A de l'ensemble de référence X correspond à l'identification des degrés avec lesquels une propriété $Prop(A)$ est satisfaite par les éléments de X . Comment peut-on construire le sous-ensemble flou défini par les degrés avec lesquels $Prop(A)$ n'est pas satisfaite ? Comment peut-on construire celui défini par les degrés avec lesquels deux propriétés $Prop(A)$ et $Prop(B)$ sont satisfaites simultanément ? ... Des opérations sur les sous-ensembles flous sont définies pour répondre à de telles questions.

Le concept de sous-ensemble flou de X étant une généralisation de la notion de sous-ensemble classique, ces opérations sont choisies de façon à être équivalentes aux opérations classiques lorsque les fonctions d'appartenance ne prennent que les valeurs 0 ou 1.

Définition 2.2 : Soit X un ensemble classique de référence, A et B deux sous-ensembles flous de X .

1. Egalité : $A = B$ si $\forall x \in X : f_A(x) = f_B(x)$.

2. Inclusion : $A \subset B$ si $\forall x \in X : f_A(x) \leq f_B(x)$.

3. Intersection : $A \cap B = C$ tel que : $\forall x \in X : f_C(x) = \min(f_A(x), f_B(x))$.

4. Union : $A \cup B = D$ tel que : $\forall x \in X : f_D(x) = \max(f_A(x), f_B(x))$.

5. Complément d'un sous-ensemble flou : $\forall x \in X : f_{A^c}(x) = 1 - f_A(x)$.

Comme dans la théorie des ensembles classique, les définitions que nous venons de donner conduisent aux propriétés suivantes :

Propriétés :

- \cup et \cap sont associatives,
- \cup et \cap sont commutatives,
- \cup est distributive par rapport à \cap , et \cap par rapport \cup ,

- $A \cap \emptyset = \emptyset$, $A \cap X = A$,
- $A \cup \emptyset = A$, et $A \cup X = X$,
- $(A^c)^c = A$,
- Lois de De Morgan : $(A \cap B)^c = A^c \cup B^c$, $(A \cup B)^c = A^c \cap B^c$,
- Les propriétés classiques de la non-contradiction ($A^c \cap A = \emptyset$) et du tiers exclu ($A^c \cup A = X$) ne sont pas satisfaites. Les seuls ensembles flous vérifiant ces deux propriétés sont les ensembles crisp (classiques).

Autres opérateurs pour définir les opérations ensemblistes :

Les opérateurs min, max et la complémentation à 1 ont été choisis pour définir respectivement l'intersection, l'union et le complément de sous-ensembles flous parce qu'ils préservent presque toutes les propriétés de ces opérations pour la théorie classique des ensembles.

Cependant, il existe d'autres opérateurs utilisables si l'on a des raisons spéciales de souhaiter d'autres comportements des opérations sur les sous-ensembles flous et si l'on préfère perdre d'autres propriétés que celles de la non-contradiction et du tiers exclu. Les opérateurs les plus connus sont les normes triangulaires pour l'intersection, les conormes triangulaires pour l'union et les négations pour le complément, sur lesquelles nous reviendrons dans le paragraphe 2.8.

Par exemple, si l'on définit respectivement l'intersection et l'union par :

$$\forall x \in X, f_{A \cap B}(x) = \max(f_A(x) + f_B(x) - 1, 0)$$

$$\forall x \in X, f_{A \cup B}(x) = \min(f_A(x) + f_B(x), 1)$$

alors les propriétés qui n'étaient pas vérifiées avec les choix précédents le sont et on retrouve en particulier :

$$(A^c \cap A = \emptyset) \text{ et } (A^c \cup A = X),$$

ce qui peut sembler satisfaisant, mais qui est compensé par le fait qu'on perd les deux propriétés d'idempotence :

$$A \cap A = A, \quad A \cup A = A.$$

2.3. Ensembles ordinaires associés à un sous-ensemble flou :

En présence de connaissances imprécises représentées par des sous-ensembles flous, plusieurs raisons conduisent à rechercher des sous-ensembles ordinaires qui leurs sont associés. On peut en citer [BOU 95] :

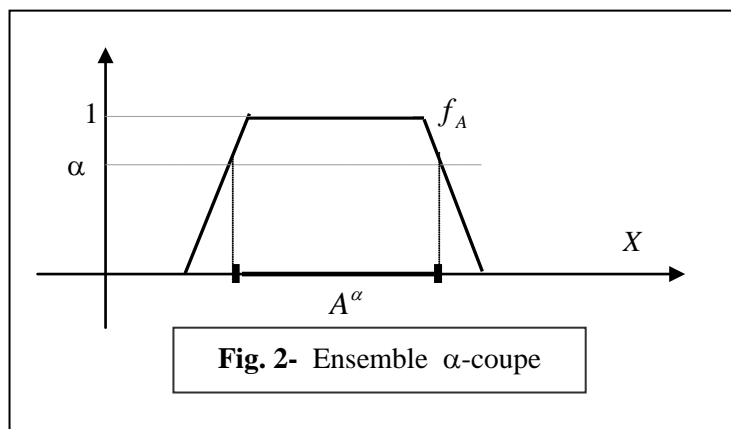
1. Evaluer à quel point ils diffèrent des ensembles ordinaires et calculer leurs degrés de flou, par exemple pour effectuer un choix entre plusieurs sous-ensembles flous en recherchant l'information la plus fiable possible,
2. Pouvoir utiliser des connaissances de la théorie classique des ensembles, en approchant (remplaçant) des sous-ensembles flous par des sous-ensembles ordinaires,
3. La recherche d'un sous-ensemble ordinaire aussi proche que possible du sous-ensemble flou disponible, dans le but de prendre une décision malgré l'imprécision des connaissances. Un cas particulier consiste en la recherche d'un seul élément de X qui représente au mieux le sous-ensemble flou. C'est ce qu'on appelle : la défuzzification.

A. Définition des α -coupes associées à un sous-ensemble flou :

Définition 2.3 : Pour un seuil donné α de $[0,1]$, on définit la **α -coupe** du sous-ensemble flou A de X (ou sous-ensemble ordinaire de niveau α associé à A) comme le sous-ensemble ordinaire :

$$A^\alpha = \{x \in X / f_A(x) \geq \alpha\}.$$

C'est le sous-ensemble des éléments de X qui satisfont $Prop(A)$ avec un degré au moins égal à α (figure 2).



B. Propriétés des α -coupes :

En choisissant un niveau α , on indique le seuil à partir duquel l'appartenance est considérée comme satisfaite. Plus on est exigeant sur la notion d'appartenance, plus on augmente le seuil α , et moins il existe d'éléments de X satisfaisant cette notion d'appartenance.

Les α -coupes des sous-ensembles flous jouissent des propriétés suivantes :

- $(A \cap B)^\alpha = A^\alpha \cap B^\alpha$,
- $(A \cup B)^\alpha = A^\alpha \cup B^\alpha$,
- Si $A \subseteq B$ alors $A^\alpha \subseteq B^\alpha$.

D'autres propriétés des α -coupes seront mises en évidence dans la suite, au cours de l'introduction de nouveaux concepts.

Remarque : On peut définir l'ensemble **α -coupe stricte** de A en considérant une inégalité stricte dans la définition 2.3.

C. Représentation d'un sous-ensemble flou à partir de ces α -coupes :

La famille de toutes les α -coupes d'un sous-ensemble flou A le représente parfaitement. L'ensemble flou A peut être reconstruit complètement à partir de ses α -coupes.

Théorème de décomposition 2.3 : Tout sous-ensemble flou A de l'ensemble de référence X est défini à partir de ses α -coupes par :

$$\forall x \in X, f_A(x) = \sup_{\alpha \in]0,1]} \alpha \cdot 1_{A^\alpha}(x).$$

2.4. Sous-ensembles flous convexes :

Les sous-ensembles flous les plus répandus sont ceux qui possèdent des fonctions d'appartenance " régulières ", c'est-à-dire sans ruptures brusques, représentant bien le caractère graduel et le passage progressif de la non-satisfaction de la propriété à laquelle ils sont associés, à sa satisfaction complète.

Lorsque X est l'ensemble des nombres réels, les sous-ensembles flous possédant une telle allure sont dits convexes [ZAD 65].

Définition 2.4 : Un sous-ensemble flou A de l'ensemble des nombres réels \mathfrak{R} est dit convexe si, pour tout couple (a, b) de points de \mathfrak{R} , et pour tout λ de $[0,1]$, la fonction d'appartenance de A vérifie :

$$f_A(\lambda.a + (1 - \lambda).b) \geq \min(f_A(a), f_A(b)),$$

ou bien d'une manière équivalente : $\forall \alpha \in [0,1], A^\alpha$ est un ensemble convexe¹ de \mathfrak{R} .

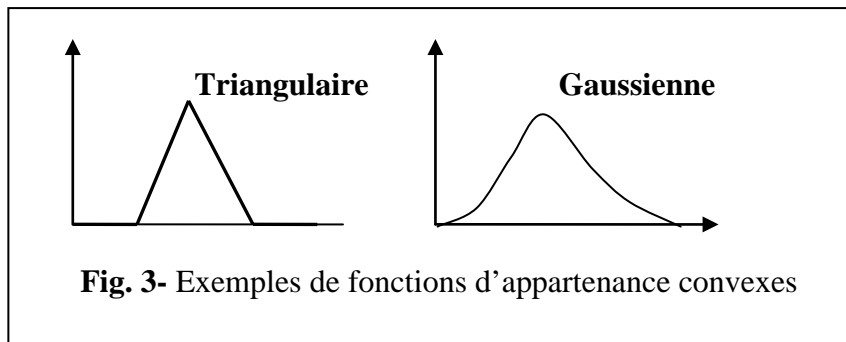


Fig. 3- Exemples de fonctions d'appartenance convexes

¹ Les ensembles convexes de \mathfrak{R} sont les intervalles.

2.5. Produit cartésien et projection de sous-ensembles flous :

A. Produit cartésien de sous-ensembles flous :

La description de tout système, même peu complexe, fait généralement intervenir plusieurs univers de référence. Par exemple, une prise de décision repose sur plusieurs critères, la reconnaissance d'une classe d'objets est basée sur plusieurs caractéristiques, la commande d'un processus prend en considération plusieurs variables. Lorsqu'on considère plusieurs ensembles de référence simultanément, on construit un univers global dont les diverses composantes sont les ensembles de référence initiaux.

On suppose que les ensembles de référence n'ont pas d'effets les uns sur les autres, c'est-à-dire que les observations faites sur chacun d'entre eux sont indépendantes les unes des autres. On dit alors que les sous-ensembles flous définis sur ces ensembles de référence sont *non-interactifs* (chapitre 2).

Soit donc X_1, \dots, X_n des ensembles de référence, leur produit cartésien est noté X . Soit A_1, A_2, \dots, A_n des sous-ensembles flous définis sur X_1, \dots, X_n respectivement.

Définition 2.5.1 : On définit leur *produit cartésien* $A = A_1 \times A_2 \times \dots \times A_n$, comme un sous-ensemble flou de X , de fonction d'appartenance :

$$\forall x = (x_1, \dots, x_n) \in X, f_A(x) = \min(f_{A_1}(x_1), \dots, f_{A_n}(x_n)).$$

B. Projection d'un sous-ensemble flou :

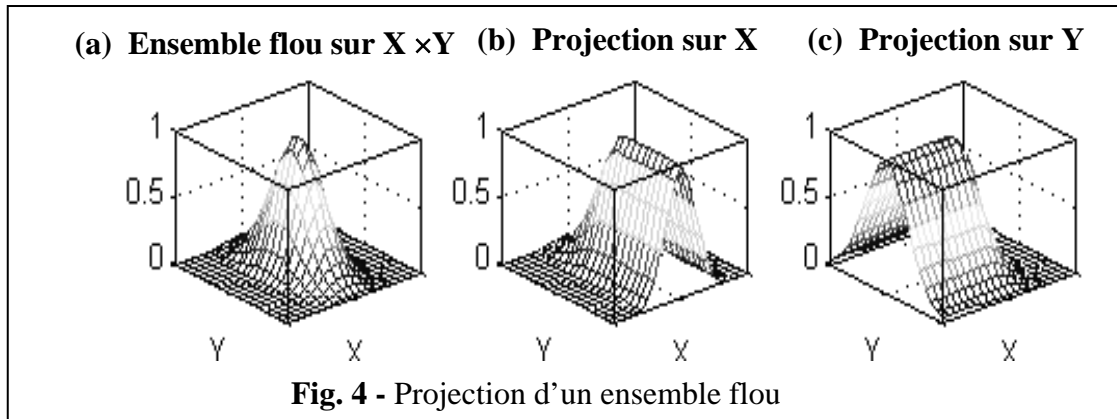
Inversement, la connaissance d'une caractérisation floue globale définie sur un univers complexe permet de donner des informations sur les différentes composantes de cet univers et définir des caractérisations floues sur chacune de ces composantes [ZAD 75].

Soit A un sous-ensemble flou défini sur l'ensemble $X_1 \times X_2$, produit cartésien de deux ensembles de référence X_1 et X_2 .

Définition 2.5.2 : *La projection* de A sur X_1 est le sous-ensemble flou $\text{Pr } oj_{X_1}(A)$, dont la fonction d'appartenance est définie par :

$$\forall x_1 \in X_1, f_{\text{Pr } oj_{X_1}(A)}(x_1) = \sup_{x_2 \in X_2} f_A((x_1, x_2)).$$

Propriété : Si $A = A_1 \times A_2$ et si A_2 est normalisé, alors $\text{Pr } oj_{X_1}(A) = A_1$.

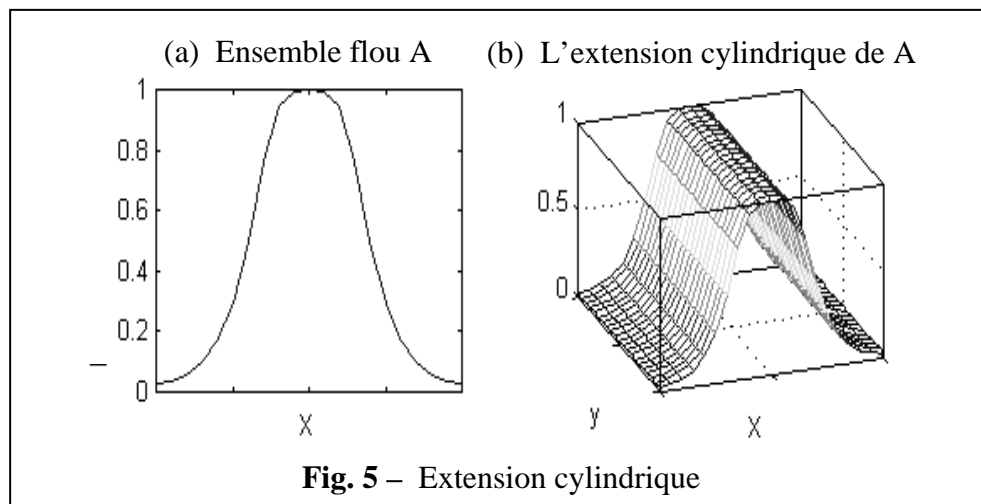


C. Extension cylindrique :

Définition 2.5.3 : L'extension cylindrique du sous-ensemble flou B de $Y = X_l \times \dots \times X_m$, avec $1 \leq l < \dots < m \leq n$, est le sous-ensemble flou B^e de $X = X_1 \times \dots \times X_n$ tel que :

$$\forall x = (x_1, \dots, x_l, \dots, x_m, \dots, x_n) \in X, f_{B^e}(x) = f_B(x_l, \dots, x_m).$$

Elle permet d'induire une connaissance sur toutes les composantes d'un univers complexe à partir d'une connaissance sur certaines d'entre elles seulement.



2.6. Principe d'extension :

L. A. Zadeh a introduit le principe d'extension [ZAD 75], l'un des plus importants de la théorie des sous-ensembles flous, pour permettre d'exploiter nos connaissances classiques dans le cas de données floues : Arithmétique floue, relations floues, ...

Soient X, Y deux ensembles de référence.

Définition 2.6 : Soit A un sous-ensemble flou de X et φ une application de X vers Y .

Le **principe d'extension** définit un sous-ensemble flou B de Y associé à A (image directe), par l'intermédiaire de φ , par :

$$\forall y \in Y, \quad f_B(y) = \sup_{\{x, \varphi(x)=y\}} f_A(x),$$

avec la convention : $\text{Sup}_{\emptyset} f_A(x) = 0$.

Exemples d'application :

- Moyennant des choix adéquats de φ , on peut retrouver à l'aide du principe d'extension, des concepts déjà vus : extension cylindrique, projection, ...
- Arithmétique floue : On peut généraliser grâce au principe d'extension les opérations arithmétiques classiques aux nombres flous : addition étendue, multiplication étendue, ...
- Application à **la règle compositionnelle** :

Etant donné trois ensembles de référence X_1, X_2 et X_3 , on considère les produits cartésiens $X = X_1 \times X_2, Y = X_2 \times X_3, Z = X_1 \times X_3$.

Soit l'application :

$$\begin{aligned} \varphi &: X \times Y && \rightarrow && Z \\ &((x_1, x_2), (x_2, x_3)) && \mapsto && (x_1, x_3) \end{aligned}$$

Etant donné deux sous-ensembles flous A de X et B de Y , le principe d'extension permet de leur associer un sous-ensemble flou C de Z , défini par :

$$f_C(x_1, x_3) = \text{Sup}_{x_2 \in X_2} \min [f_A(x_1, x_2), f_B(x_2, x_3)].$$

On obtient ainsi une règle générale importante [ZAD 79], qui sera utilisée dans tous les cas où l'on souhaite faire abstraction de certaines composantes d'un système.

2.7. Spécificité et précision d'un sous-ensemble flou :

Tous les sous-ensembles flous ne correspondent pas à un même degré d'imperfection dans les connaissances. La propriété à laquelle ils sont associés peut être plus ou moins imprécise.

Intuitivement, un sous-ensemble flou $A \in F(X)$ est dit spécifique lorsqu'il est proche de la détermination d'un élément unique (singleton). Il est dit précis lorsqu'il est proche de la détermination d'un ensemble ordinaire.

Il est utile d'évaluer le degré de spécificité ou de précision de sous-ensembles flous, pour être capable de choisir le moins flou lorsqu'on cherche à déterminer une caractérisation aussi nette que possible (pour une prise de décision par exemple) ou au contraire celui qui est le moins spécifique lorsque l'on veut conserver autant de souplesse qu'il est possible dans une description (pour l'agrégation d'informations différentes sur un même attribut, par exemple). Pour ce faire, plusieurs grandeurs mathématiques ont été introduites.

A. Degrés de flou :

Soit X un ensemble de référence, F et F' deux sous-ensembles flous de X .

Définition 2.7.1 : On définit la relation \leq_1 telle que $F' \leq_1 F$ si et seulement si :

- $f_{F'(x)} \leq f_F(x)$ pour tout x tel que $f_F(x) \leq 0.5$
- $f_{F'(x)} \geq f_F(x)$ pour tout x tel que $f_F(x) \geq 0.5$

C'est une relation d'ordre partiel.

Définition 2.7.2 : Un *degré de flou* d est une grandeur définie sur $F(X)$ et prenant ses valeurs dans l'ensemble des nombres réels, qui à tout $F \in F(X)$ fait correspondre $d(F)$ tel que :

- $d(F) = 0 \Leftrightarrow F$ est un ensemble ordinaire,
- $d(F)$ est maximale $\Leftrightarrow \forall x \in X : f_F(x) = 0.5$ (ensemble complètement flou),
- $d(F') \leq d(F) \Leftrightarrow F' \leq_1 F$.

Un exemple de degré de flou [KAU 73] est défini à partir de la distance entre F et le plus proche sous-ensemble non flou de X , au sens de la distance euclidienne, c'est-à-dire sa 1/2-coupe stricte $F^{1/2}$. Ce degré de flou peut être construit en utilisant, par exemple, la distance de Hamming ou la distance euclidienne :

$$d_1(F) = \sum_{x \in X} |f_F(x) - f_{F^{0.5}}(x)|, \quad d_2(F) = [\sum_{x \in X} (f_F(x) - f_{F^{0.5}}(x))^2]^{1/2}.$$

valables pour tout ensemble discret X .

B. Mesures de spécificité :

Une mesure de spécificité évalue dans quelle mesure la propriété observée $Prop(F)$ est proche de la détermination d'un élément unique de X .

Définition 2.7.3 : Une mesure de *spécificité* est une grandeur Sp définie sur $F(X)$, à valeurs dans $[0,1]$, telle que :

- $Sp(F) = 1 \Leftrightarrow F$ est un singleton,
- $F \subseteq F' \Rightarrow Sp(F) \geq Sp(F')$.

Exemple : Soit X un ensemble de référence fini.

On définit une mesure de spécificité sur $F(X)$ par :

$$Sp(F) = \sum_{i=1,2,\dots} [g_i - g_{i+1}] / i,$$

où g_1, g_2, g_3, \dots représentent les degrés d'appartenance des éléments de X rangés en ordre décroissant.

2.8. Normes et conormes triangulaires :

Les opérations d'intersection, d'union et de complémentation de sous-ensembles flous habituellement employées peuvent être remplacées par d'autres opérations construites à l'aide d'opérateurs différents du minimum, du maximum et de la complémentation à 1. On fait appel à eux lorsque les opérations habituelles ne s'avèrent pas satisfaisantes, par exemple en logique floue ou dans la commande de processus [BOU 95].

A. Définition et propriétés des normes et conormes triangulaires :

Définition 2.8.1 : Une *norme triangulaire* (*t-norme*) est une fonction T de $[0,1] \times [0,1]$ dans $[0,1]$, vérifiant :

1. Commutativité : $T(u, v) = T(v, u)$,
2. Associativité : $T(u, T(v, w)) = T(T(u, v), w)$,
3. Isotonie : $u \leq w, v \leq t \Rightarrow T(u, v) \leq T(w, t)$,
4. Élément neutre : $T(u, 1) = u$.

Cas particulier : L'opérateur $T = \min$ est une norme triangulaire.

Toute t-norme peut servir à définir l'intersection de deux sous-ensembles flous comme suit :

$$C = A \cap_T B, \text{ tel que : } \forall x \in X, f_C(x) = T(f_A(x), f_B(x)).$$

Cette opération est commutative, associative et monotone par rapport à l'inclusion. De plus, $A \cap X = A$.

Définition 2.8.2 : Une *conorme triangulaire* (*t-conorme*) est une fonction S de $[0,1] \times [0,1]$ dans $[0,1]$, vérifiant :

1. Commutativité : $S(u, v) = S(v, u)$;
2. Associativité : $S(u, S(v, w)) = S(S(u, v), w)$;
3. Isotonie : $u \leq w, v \leq t \Rightarrow S(u, v) \leq S(w, t)$;
4. Élément neutre : $S(u, 0) = u$.

Cas particulier : L'opérateur $S = \max$ est une conorme triangulaire.

Toute t-conorme peut servir à définir l'union de deux sous-ensembles flous comme suit :

$$D = A \cup_S B, \text{ tel que : } \forall x \in X, f_D(x) = S(f_A(x), f_B(x)).$$

Cette opération est commutative, associative et monotone par rapport à l'inclusion. De plus, $A \cup_S X = A$.

Remarque : Les t-normes et t-conormes peuvent être étendues à une dimension supérieure par associativité.

Propriété : Soit T une t-norme, S une t-conorme. Alors :

$$T_{\text{drastique}}(u, v) \leq T(u, v) \leq \min(u, v)$$

$$\max(u, v) \leq S(u, v) \leq S_{\text{drastique}}(u, v)$$

Voir le tableau 1 (page 18) pour les définition de $T_{\text{drastique}}$ et $S_{\text{drastique}}(u, v)$.

B. Négation et dualité entre opérateurs :

De même que l'intersection et l'union usuelles de sous-ensembles flous vérifient les lois de Morgan relativement au complément, comme nous l'avons indiqué, on peut associer une intersection \cap_T et une union \cup_S de sous-ensembles flous par l'intermédiaire d'un opérateur définissant le complément, de telle façon à vérifier les lois de Morgan, et que l'on appelle une négation.

Définition 2.8.3 : Une *négation* est une fonction $n : [0,1] \rightarrow [0,1]$, telle que :

- i) $n(0) = 1$ et $n(1) = 0$;
- ii) $u \geq v \Rightarrow n(u) \leq n(v)$.

Elle est dite *stricte* si $n(x)$ est continue et strictement monotone. Si, de plus, $n(n(u)) = u$ pour tout u dans $[0,1]$, elle est dite *involution*.

Cas particulier : La complémentation à 1 est une négation stricte et involutive.

Etant donné un sous-ensemble flou A de X , on peut utiliser une négation involutive pour définir son complément A^{cn} par :

$$\forall x \in X, f_{A^{cn}}(x) = n(f_A(x)).$$

Définition 2.8.4 : Une t-norme T et une t-conorme S sont dites duales pour la négation stricte n si :

$$- \quad n(T(u, v)) = S(n(u), n(v)),$$

- $n(S(u, v)) = T(n(u), n(v))$.

Propriété : Pour une t-norme et une t-conorme duales pour la négation stricte n , les lois de Morgan sont vérifiées :

$$(A \cap_T B)^{cn} = A^{cn} \cup_S B^{cn} \text{ et } (A \cup_S B)^{cn} = A^{cn} \cap_T B^{cn}.$$

Les t-normes et t-conormes les plus utilisées sont indiquées dans le Tableau 1 [GUP 91], [DPR 91].

Nom	t-norme	t-conorme	Négation
Zadeh	$\min(u, v)$	$\max(u, v)$	$1 - u$
Probabiliste	$u.v$	$u + v - u.v$	$1 - u$
Lukasiewicz	$\max(u + v - 1, 0)$	$\text{Min}(u + v, 1)$	$1 - u$
Hamacher ($\gamma > 0$)	$\frac{u.v}{\gamma + (1 - \gamma).(u + v - u.v)}$	$\frac{u + v + uv - (1 - \gamma)uv}{1 - (1 - \gamma)uv}$	$1 - u$
Yager ($p > 0$)	$\max(1 - ((1 - u)^p + (1 - v)^p)^{1/p}, 0)$	$\min((u^p + v^p)^{1/p}, 1)$	$1 - u$
Weber ($\lambda > -1$)	$\max((u + v - 1 + \lambda uv)/(1 + \lambda), 0)$	$\min(u + v + \lambda uv, 1)$	$(1 - u)/(1 + \lambda u)$
Drastique	u Si $v = 1$ v Si $u = 1$ 0 ; Sinon	u Si $v = 0$ v Si $u = 0$ 1 ; Sinon	$1 - u$

Tab. 1: Principales t-normes et t-conormes duales.

3. Relations et quantités floues :

Le concept de sous-ensemble flou constituant un assouplissement de celui de sous-ensemble ordinaire, en ce sens qu'il autorise à des éléments à n'appartenir que partiellement, nous avons montré que toutes les opérations classiques que l'on peut effectuer sur des sous-ensembles ou des éléments précisément connus peuvent être

étendues de façon à réaliser des opérations analogues lorsque des imperfections dans les connaissances obligent à utiliser des sous-ensembles flous. Tous les concepts mathématiques classiques qui sont utilisés lorsqu'on dispose de connaissances précises doivent également être adaptés en présence de connaissances imprécises. Par exemple, comment peut-on comparer deux valeurs imprécises ? Comment peut-on les combiner dans un calcul arithmétique ?

3.1. Relations floues :

Parmi les concepts flous les plus importants du point de vue des applications qu'ils peuvent avoir, les relations floues généralisent la notion de relation classique. Elles mettent en évidence des liaisons imprécises ou graduelles entre éléments d'un même ensemble [ZAD 71].

A. Définition de relations floues :

Définition 3.1.1 : Une *relation floue* R entre n ensembles de référence X_1, X_2, \dots, X_n est un sous-ensemble flou de $X_1 \times X_2 \times \dots \times X_n$, de fonction d'appartenance f_R .

Si $n = 2$, $X_1 = X_2 = X$; R est dite *relation binaire floue* sur X .

Cas particulier : Si X, Y sont deux ensembles finis, toute relation floue R sur $X \times Y$ peut être décrite par une matrice $M(R)$, ses éléments étant $f_R(x_i, y_j)$.

Remarque : Les relations floues étant des cas particuliers de sous-ensembles flous, toutes les propriétés et définitions qui concernent les sous-ensembles flous leurs sont applicables.

B. Composition de relations floues :

Soient trois ensembles de référence X, Y et Z . La connaissance de deux relations floues, l'une entre X et Y , l'autre entre Y et Z permet d'établir une relation floue entre X et Z , par composition, comme c'est le cas des relations classiques.

Définition 3.1.2 : La *composition* de deux relations floues R_1 sur $X \times Y$ et R_2 sur $Y \times Z$ définit une relation floue $R = R_1 \circ R_2$ sur $X \times Z$ de fonction d'appartenance définie par :

$$\forall (x, z) \in X \times Z, f_R(x, z) = \sup_{y \in Y} \min(f_{R_1}(x, y), f_{R_2}(y, z)).$$

Cette définition correspond à la composition sup-min. Il est cependant possible de remplacer l'opérateur min par un autre opérateur T , par exemple une norme triangulaire, pour définir la composition T-sup.

C. Propriétés particulières des relations binaires floues :

Parmi les relations binaires floues, celles qui servent à décrire des ressemblances " à peu près égal " ou des antériorités " beaucoup plus petit " sont les plus fréquentes. On essaie donc de les caractériser en étendant les notions classiques de relation d'équivalence et de relation d'ordre. Dans ce but, on définit des propriétés particulières des relations floues définies sur $X \times X$ [ZAD 71].

Définition 3.1.3 : La relation binaire floue R sur X est dite :

- i) **Réflexive** si : $\forall x \in X : f_R(x, x) = 1$,
- ii) **Symétrique** si : $\forall (x, y) \in X \times X : f_R(x, y) = f_R(y, x)$,
- iii) **Transitive** si : $R \circ R \subseteq R$,
- iv) **Antisymétrique** si : $\forall (x, y) \in X \times X : f_R(x, y) > 0, f_R(y, x) > 0 \Rightarrow x = y$.

Remarque : Ces définitions généralisent les définitions classiques.

Pour mettre en évidence la compatibilité entre les définitions de propriétés particulières de relations floues qui ont été présentées et les propriétés analogues pour les relations classiques, on considère pour tout niveau $\alpha \in]0, 1]$ la α -coupe R^α d'une relation floue R , définie comme celle d'un ensemble flou quelconque.

Définition 3.1.4 : Etant donné un seuil α dans $[0, 1]$, la relation de niveau α associée à R est la relation ordinaire R^α telle que : $x R^\alpha y \Leftrightarrow f_R(x, y) \geq \alpha$.

Propriétés :

- R est réflexive $\Leftrightarrow \forall \alpha \in]0, 1] : R^\alpha$ est réflexive ;
- R est symétrique $\Leftrightarrow \forall \alpha \in]0, 1] : R^\alpha$ est symétrique ;
- R est antisymétrique $\Leftrightarrow \forall \alpha \in]0, 1] : R^\alpha$ est antisymétrique ;
- R est transitive $\Leftrightarrow \forall \alpha \in]0, 1] : R^\alpha$ est transitive.

On va maintenant généraliser les notions classiques de relations d'équivalence et relations d'ordre.

- Relations de similarité :

Pour représenter l'idée de ressemblance ou de proximité, on utilise une relation floue qui étend la notion d'équivalence.

Définition 3.1.5 : Une *relation de similarité* est une relation binaire floue réflexive, symétrique et transitive.

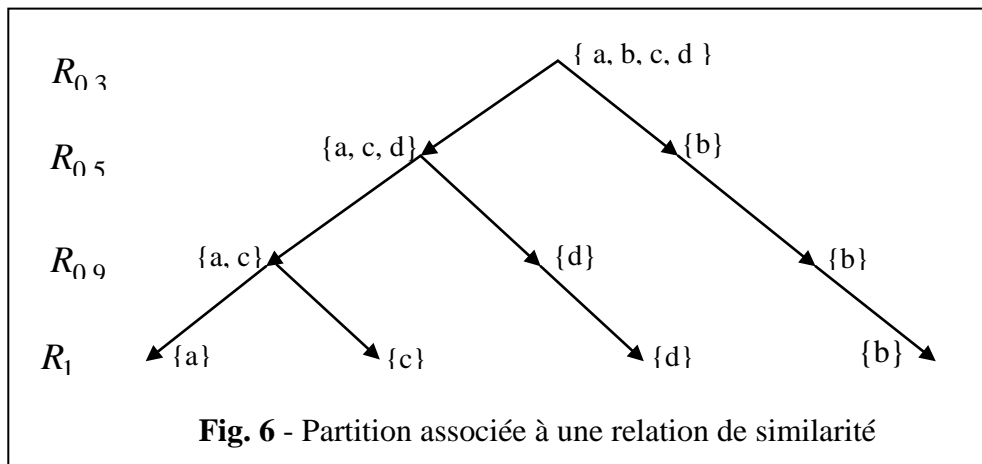
Propriété : R est une relation de similarité $\Leftrightarrow \forall \alpha \in]0,1] : R^\alpha$ est une relation d'équivalence.

Les éléments satisfaisant une relation d'équivalence classique peuvent être regroupés en classes d'équivalence. De façon analogue, on réunit dans une même classe les éléments de X dont la ressemblance est mise en évidence par la relation floue R . Si l'on considère R elle-même, tous les couples de X la satisfont, avec des degrés variables. On fait alors appel aux relations de niveau α associées à R .

Exemple : Soit l'ensemble de référence $X = \{a, b, c, d\}$, et une relation binaire de similarité R , dont la fonction d'appartenance est définie par la matrice suivante :

	A	B	C	D
A	1	0.3	0.9	0.5
B	0.3	1	0.3	0.3
C	0.9	0.3	1	0.5
D	0.5	0.3	0.5	1

On obtient 4 relations différentes de niveau α associées à R , dont on vérifie que ce sont des relations d'équivalence (voir figure 6).



Les relations de similarité interviennent par exemple dans les problèmes de classification et de reconnaissance des formes.

- Relations d'ordre floues :

L'autre grande classe de relations floues possédant des propriétés spécifiques constitue une extension des relations d'ordre ordinaires. Elles correspondent à l'idée de préférence ou d'antériorité.

Définition 3.1.6 : Une *relation d'ordre floue* est une relation binaire floue réflexive, antisymétrique et transitive.

Propriété : R est une relation d'ordre floue $\Leftrightarrow \forall \alpha \in]0,1] : R^\alpha$ est une relation d'ordre.

3.2. Quantités floues :

A. Définition des quantités floues :

Dans les applications, une proportion importante de sous-ensembles flous représente des propriétés de variables prenant leurs valeurs dans l'ensemble \mathfrak{R} des nombres réels, comme le prix, la taille, le poids, la pression, l'âge, le taux de croissance, etc. Ils correspondent à des mesures dont la précision n'est pas absolue ou à des estimations imprécises recueillies auprès d'observateurs ou d'experts. L'univers de référence X est alors \mathfrak{R} et les sous-ensembles flous sont généralement normalisés, appelés des quantités floues.

Définition 3.2 : Une *quantité floue* est un sous-ensemble flou normalisé de \mathfrak{R} .

On traite les quantités floues à l'aide de l'arithmétique floue (chapitre 3), et des relations floues. Ils correspondent à l'idée de voisinage d'une valeur précise ou intervalle de valeurs aux bornes mal déterminées.

B. Opérations arithmétiques floues :

Le principe d'extension permet de faire toutes les opérations arithmétiques classiques avec des nombres flous, en choisissant à chaque fois la fonction φ de la définition 2.6. convenablement.

Soient A et B deux quantités floues.

En choisissant $\varphi(x, y) = x + y$, par exemple, on définit l'addition floue par :

$$f_{A+B}(z) = \sup_{(x+y=z)} (\min(f_A(x), f_B(y))) ,$$

On reviendra sur l'arithmétique floue dans le chapitre 3.

C. Relations floues entre quantités floues :

Des grandeurs réelles peuvent être comparées à l'aide de relations floues, qui assouplissent l'égalité et la relation d'ordre classique par exemple.

Si l'une de ces grandeurs a pour valeurs une quantité floue N , la relation R qui la lie à la seconde variable permet de donner la valeur floue de celle-ci comme une quantité floue M de fonction d'appartenance :

$$\forall x \in \mathfrak{R} \quad f_M(x) = \sup_{y \in \mathfrak{R}} \min(f_R(x, y), f_N(y)),$$

en application de la règle compositionnelle.

Exemple : Soit R la relation " plus petit que ", M et N deux quantités floues.

Sachant que M est plus petit que N , on peut déduire la fonction d'appartenance de M à partir de celle de N par composition de R et la fonction d'appartenance de N .

Chapitre II

Théorie des Possibilités et Raisonnement Approximatif

" Le vrai danger, ce n'est pas quand les ordinateurs penseront comme les hommes, c'est quand les hommes penseront comme les ordinateurs."

Sydney J. Harris.

1. La théorie des possibilités

La théorie des sous-ensembles flous a été introduite pour répondre aux besoins de la représentation des connaissances exprimées symboliquement en langage naturel, soumises à des imprécisions ou présentant un caractère vague. Néanmoins, les incertitudes sur la véracité d'une affirmation ne sont pas traitées. Or, incertitudes et imprécisions sont étroitement liées. Ceci a amené L. Zadeh à introduire la théorie des possibilités en 1978, pour permettre de manipuler des incertitudes de nature non probabiliste, et pour constituer un cadre dans lequel connaissances imprécises et connaissances incertaines peuvent coexister et être traitées conjointement [ZAD 78].

La théorie des possibilités est une théorie mathématique basée sur la théorie classique des ensembles, où les concepts flous ne sont pas requis à priori, et qui correspond à l'introduction de la nouvelle notion de distribution de possibilité (voir 1.3, chapitre 1).

1.1. Mesures et distributions de possibilité :

La théorie des possibilités fournit une méthode pour formaliser les incertitudes subjectives sur des événements, c'est-à-dire un moyen de dire dans quelle mesure la réalisation d'un événement est possible et à quel point on en est certain sans, toutefois, avoir à sa disposition l'évaluation de sa probabilité.

A. Mesure de possibilité :

Etant donné un ensemble de référence X , on attribue à chaque événement de X , c'est-à-dire à toute partie de X , un coefficient compris entre 0 et 1, évaluant à quel point cet événement est possible.

Définition 1.1.1 : Une *mesure de possibilité* π est une fonction $\pi : P(X) \rightarrow [0,1]$, vérifiant :

- i) $\pi(\phi) = 0, \pi(X) = 1,$
- ii) $\forall (A_i)_{i \in I} \in P(X) : \pi(\bigcup_i A_i) = \text{Sup}_i \pi(A_i),$ où I est un ensemble d'indices quelconque.

Donc une mesure de possibilité n'est pas additive.

Un événement A est dit tout à fait possible si $\pi(A) = 1$, impossible si $\pi(A) = 0$.

Propriétés :

- i) $\pi(A \cup B) = \text{Max}(\pi(A), \pi(B)),$
- ii) $\pi(A \cap B) \leq \text{Min}(\pi(A), \pi(B)),$
- iii) $A \subseteq B \Rightarrow \pi(A) \leq \pi(B),$
- iv) $\text{Max}(\pi(A), \pi(\overline{A})) = 1.$

B. Distribution de possibilité :

Une mesure de possibilité est totalement définie si l'on attribue un coefficient à toute partie de X ou bien, d'une façon équivalente, à toute partie élémentaire (singleton).

Définition 1.1.2 : Une *distribution de possibilité* π est une fonction $\pi : X \rightarrow [0,1]$,

vérifiant la condition de normalité : $\text{Sup}_{x \in X} \pi(x) = 1.$

La mesure de possibilité associée est définie par : $\pi(A) = \text{Sup}_{x \in A} \pi(x).$

C. Distributions de possibilité conjointes

Soient deux ensembles de référence X et Y .

Définition 1.1.3 : On appelle distribution de possibilité *conjointe* $\pi(x, y)$, une distribution de possibilité définie sur le produit $X \times Y$.

Cette connaissance globale π sur $X \times Y$ fournit une information sur chacun des ensembles de référence X et Y , pris individuellement, sous forme de distributions marginales.

Définition 1.1.4 : Etant donné une distribution de possibilité conjointe π sur $X \times Y$, on définit les distributions de possibilité *marginales* par :

$$\pi_X(x) = \sup_{y \in Y} \pi(x, y) \quad , \quad \pi_Y(y) = \sup_{x \in X} \pi(x, y).$$

Définition 1.1.5 : Etant donné deux distributions de possibilité π_X, π_Y sur X et Y , les ensembles de références sont dits *non-interactifs* si : $\pi(x, y) = \text{Min}(\pi_X(x), \pi_Y(y))$.

L'influence réciproque des ensembles de référence peut être étudiée à l'aide des distributions de possibilités conditionnelles :

Définition 1.1.6 : La distribution de possibilité *conditionnelle* $\pi_{Y/X}$ est l'ensemble des degrés avec lesquels les éléments y de Y sont possibles sachant qu'un élément x de X est possible, de telle sorte que :

$$\pi(x, y) = \pi_{Y/X}(x, y) * \pi_X(x),$$

où $*$ est un opérateur de combinaison qui est généralement le minimum ou le produit.

1.2. Dualité entre mesure de possibilité et de nécessité :

A. Mesure de nécessité :

Une mesure de possibilité donne une information sur l'occurrence d'un événement A relatif à un ensemble de référence X , mais elle ne suffit pas pour décrire l'incertitude existant sur cet événement ; en particulier, on peut avoir $\pi(A) = 1$, et $\pi(A^c) = 1$.

On note $P(X)$ l'ensemble des parties de X .

Définition 1.2 : Une *mesure de nécessité* est une fonction $N : P(X) \rightarrow [0,1]$, vérifiant :

- i) $N(\phi) = 0, N(X) = 1.$
- ii) $N(\bigcap_i A_i) = \inf N(A_i)$, où I un ensemble d'indices quelconque.

Propriétés :

- i) $N(A \cap B) = \text{Min}(N(A), N(B)),$
- ii) $N(A \cup B) \geq \text{Max}(N(A), N(B)),$
- iii) $A \subseteq B \Rightarrow N(A) \leq N(B),$
- iv) $\text{Min}(N(A), N(A^c)) = 0.$

B. Relation entre mesure de nécessité et mesure de possibilité :

Les axiomes définissant les mesures de possibilité et de nécessité sont duaux. En particulier, une mesure de nécessité peut être obtenue à partir de la donnée d'une mesure de possibilité par la formule :

$$N(A) = 1 - \pi(A^c).$$

N est dite : mesure de nécessité duale de π .

Propriétés : Soit π une mesure de possibilité, N la mesure de nécessité duale associée. Alors :

- $N(A) \leq \pi(A)$,
- $\pi(A) \neq 1 \Rightarrow N(A) = 0$.

En l'absence de $P(A)$, l'évaluation de la probabilité de l'événement A , on se sert du couple $(N(A), \pi(A))$ pour représenter l'incertitude sur l'occurrence de l'événement A .

1.3. Comparaison entre Probabilité et Possibilité, théorie de

l'Evidence :

On peut établir un parallélisme entre théorie des probabilités et théorie des possibilités. Pour ce faire, on se place dans un cadre plus général, celui des mesures floues. On en présente un cas particulier important, celui des mesures de croyance, introduites dans la théorie de l'évidence, qui présente l'intérêt d'apparaître comme plus générale que la théorie des possibilités et la théorie des probabilités [KOH 94].

A. Mesures floues :

Une mesure floue est une fonction qui attribue à tout événement de X un coefficient compris entre 0 et 1, indiquant dans quelle mesure on peut penser que l'événement se réalisera, en se basant sur des connaissances plus ou moins subjectives.

Définition 1.3.1 : Une *mesure floue* est une fonction $g : P(X) \rightarrow [0,1]$, vérifiant :

- i) $g(\phi) = 0, g(X) = 1$,
- ii) $A \subseteq B \Rightarrow g(A) \leq g(B)$,

Propriétés :

- i) $g(A \cup B) \geq \text{Max}(g(A), g(B))$;
- ii) $g(A \cap B) \leq \text{Min}(g(A), g(B))$;

Exemples : Les mesures de possibilités, de nécessités et de probabilités sont des mesures floues.

B. Fonctions de croyance et de plausibilité :

Pour modéliser la crédibilité attribuée à des faits dont on ne connaît pas la probabilité, on utilise des fonctions de croyance. On se place dans le cadre de la théorie de l'évidence Dempster-Shafer [KOH 94]. On présente l'approche de Shafer (1976), qui possède le mérite de permettre de mettre en évidence le parallélisme entre possibilité et probabilité.

On considère un ensemble de référence X que l'on suppose fini pour simplifier. On répartit une masse de croyance égale à 1 entre tous les événements possibles de X .

Définition 1.3.2 : On appelle *masse de croyance* une fonction $m : P(X) \rightarrow [0,1]$, vérifiant :

- i) $m(\emptyset) = 0$,
- ii) $\sum_{A \in P(X)} m(A) = 1$.

On appelle *élément focal* toute partie non vide E de X , telle que $m(E) \neq 0$.

Définition 1.3.3 : On appelle *fonction de croyance* la fonction $Bel : P(X) \rightarrow [0,1]$, définie par :

$$Bel(A) = \sum_{E \subseteq A} m(E).$$

Définition 1.3.4 : On appelle *fonction de plausibilité* la fonction $Pl : P(X) \rightarrow [0,1]$, définie par :

$$Pl(A) = \sum_{E \cap A \neq \emptyset} m(E), \text{ ou encore } Pl(A) = 1 - Bel(A^c).$$

Propriété : Les fonctions de croyance et de plausibilité sont des mesures floues.

Exemple :

Supposons qu'on dispose d'une urne dans laquelle on place 10 boules, dont 5 rouges, 2 bleues, 2 vertes et une noire. L'ensemble de référence est noté $X = \{r, b, v, n\}$. On tire au hasard une des boules.

Alors : $P(r) = 0,5$, $P(b) = 0,2$, $P(v) = 0,2$, $P(n) = 0,1$.

Nous disposons ainsi de toute la connaissance possible sur l'occurrence des événements.

Maintenant, supposons que l'on mette 4 boules rouges, 2 bleues, 2 vertes et une noire dans l'urne, puis on ajoute une dont ne nous indique pas la couleur. Nous pouvons alors répartir notre masse de croyance entre les différentes éventualités :

$$m(r) = 0,4, \quad m(b) = 0,2, \quad m(v) = 0,2, \quad m(n) = 0,1,$$

et il reste une portion de 0,1 que nous ne savons pas localiser et que nous attribuons donc à X tout entier ($m(X) = 0,1$).

Nous pouvons alors calculer la plausibilité et la croyance attribuées à chacun des événements, comme indiqué sur le tableau ci-dessous. La dernière boule est de couleur inconnue ; si l'on fait quatre hypothèses sur cette couleur, chacune conduit à la détermination d'une distribution de probabilités : f_r si la boule est rouge, f_b si elle est bleue, f_v si elle est verte, et f_n si elle est noire.

<i>Événements</i>	$\{r\}$	$\{b\}$	$\{v\}$	$\{n\}$	$\{r, b, v, n\}$	$\{r, b\}$
<i>Bel / Pl</i>	0,4 / 0,5	0,2 / 0,3	0,2 / 0,3	0,1 / 0,2	1 / 1	0,6 / 0,7
f_r	0,5	0,2	0,2	0,1	1	0,7
f_b	0,4	0,3	0,2	0,1	1	0,7
f_v	0,4	0,2	0,3	0,1	1	0,6
f_n	0,4	0,2	0,2	0,2	1	0,6

Tab. 1 : Exemple des croyances attribuées à la sortie de boules d'une urne.

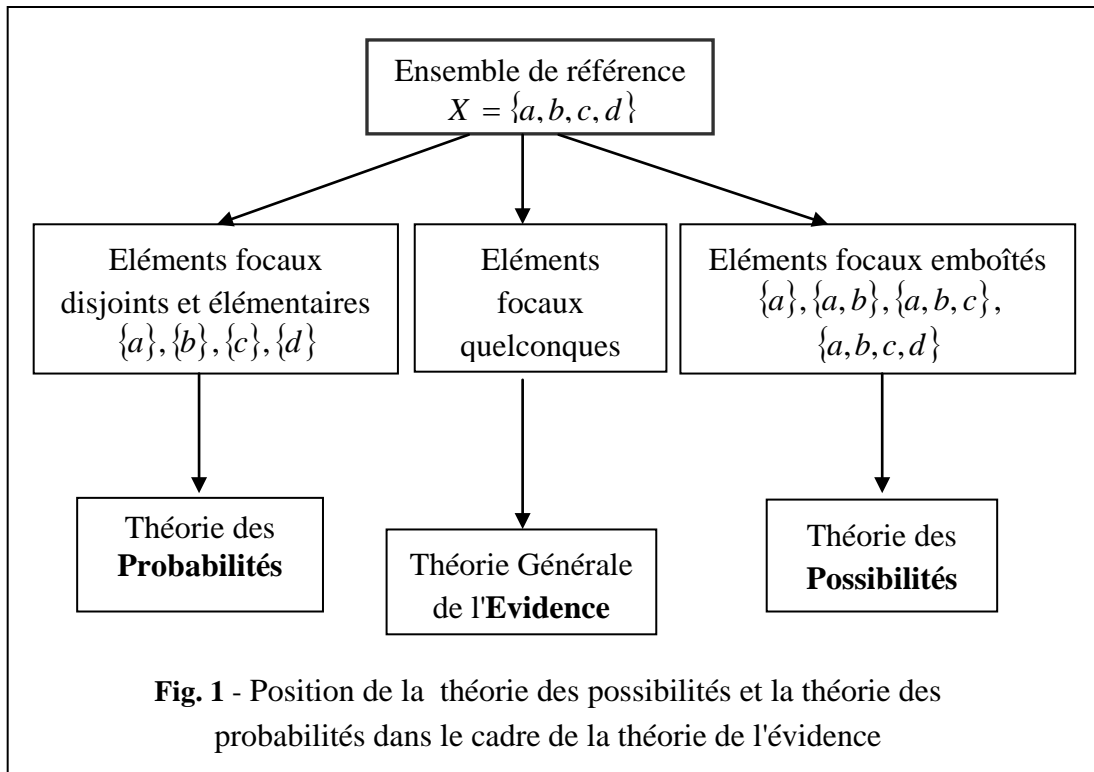
On remarque que dans tous les cas de figure, croyance et plausibilité encadrent la probabilité.

Propriété : La croyance et la plausibilité d'un événement quelconque encadrent sa probabilité inconnue :

$$Bel(A) \leq P(A) \leq Pl(A).$$

C. Cas particuliers :

La théorie de l'évidence accepte toutes les répartitions possibles de la masse initiale de croyance entre les divers événements. Deux situations sont particulièrement remarquables, et conduisent à la mise en évidence des théories des possibilités et des probabilités comme cas particuliers de la théorie de l'évidence (fig. 1) [BOU 95].



1.4. La théorie des ensembles flous comme base de celle des possibilités :

La théorie des ensembles flous fournit une base naturelle à la théorie des possibilités [ZAD 78].

Dans cette perspective, une contrainte souple (floue) sur les valeurs que peut prendre une variable x comme par exemple "proche de 1" ou "beaucoup plus grand que 2" induit une distribution de possibilités sur les valeurs que peut prendre cette variable. On associe donc à une variable dont les valeurs sont floues une distribution de possibilité de la même manière qu'on associe à une variable aléatoire dont les valeurs ont stochastiques une distribution de probabilités.

Par ailleurs, on peut interpréter toute distribution de possibilité comme une restriction floue élastique sur les valeurs que peut prendre une variable. La distribution de possibilité devient alors une fonction d'appartenance d'un ensemble flou représentant cette contrainte floue. Cela nous permet d'utiliser les outils de la théorie des ensembles flous comme par exemple la projection ou l'extension cylindrique pour induire de nouvelles distributions de possibilités.

1.5. Principes possibilistes pour le Raisonnement Approximatif :

On va énoncer dans ce qui suit, deux principes de la théorie des possibilités très importants dans la théorie du raisonnement approximatif. Cette dernière peut être vue comme une application de la théorie des possibilités [DPR 91].

A. Principe de la spécificité minimale :

Soit U le domaine de variation d'une variable x .

Supposons que différentes sources d'informations fournissent n contraintes sur les valeurs que peut prendre x , sous forme de distributions de possibilités π_i , $i = 1, \dots, n$.

Le principe de spécificité minimale associe à x la distribution de possibilités :

$$\pi = \min_i \pi_i,$$

Cette distribution de possibilité agrège les différentes distributions issues de sources différentes, c'est à dire, elle permet de prendre en compte simultanément toutes les contraintes représentées par les différentes distributions de possibilités sous forme d'une seule distribution.

Notons que l'extension cylindrique et la projection sont en accord avec le principe de spécificité minimale, car elles associent la distribution de possibilités la moins spécifique.

B. Principe de combinaison/projection

Soient $x_i / i = 1, \dots, n$ des variables ayant leurs valeurs dans des ensembles de référence U_i .

A partir d'une connaissance globale sur l'espace produit $U_1 \times U_2 \times \dots \times U_n$, sous forme de distribution de possibilités conjointe, on peut déduire une distribution de possibilité pour chaque x_i par projection :

$$\pi_i(x_i) = \sup_{u_i = x_i} \pi(u_1, u_2, \dots, u_n)$$

Possédant des distributions de possibilités sur plusieurs variables interactives ou non, le principe de combinaison/projection permet de tirer la distribution de possibilité de chaque variable, en suivant les deux étapes suivantes :

- a. Combiner les différentes distributions de possibilités en une seule, définie sur l'espace produit, en utilisant l'extension cylindrique, suivie du principe de la spécificité minimale pour l'agrégation des différentes distributions en une seule,
- b. Tirer la distribution de chaque variable par projection.

2. Variables linguistiques et propositions floues

On définit dans ce qui suit, la notion de variables linguistiques, introduite par Zadeh en 1975, et on introduit le concept de proposition floue qui est utilisé en logique floue. Les fonctions d'appartenance des caractérisations floues des variables conduisent à la définition de distributions de possibilités qui permettent de traiter les incertitudes engendrées par ces caractérisations floues [BOU 95].

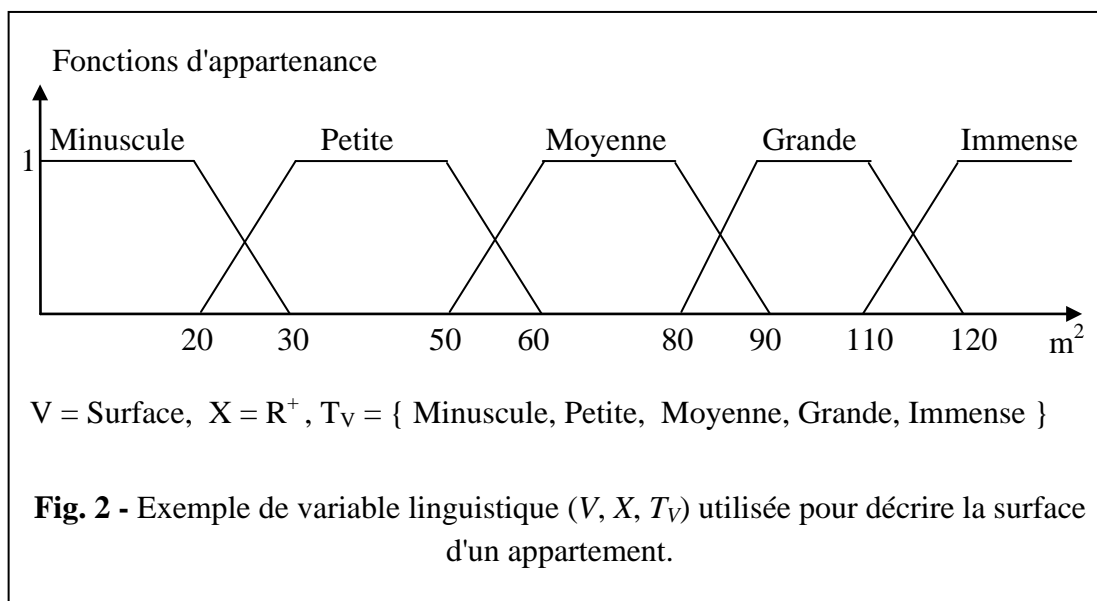
2.1. Variables linguistiques :

La notion de variable linguistique a été introduite par Zadeh [ZAD 75]. Une variable linguistique est une variable prenant ses valeurs dans un ensemble de "mots" symboliques (sous-ensembles flous), définissant certaines catégories d'un ensemble de référence.

Définition 2.1 : On appelle *variable linguistique* un triplet (V, X, T_V) , tel que :

- i) X : un ensemble de référence,
- ii) V : une variable symbolique définie sur X ,
- iii) $T_V : T_V = \{A_1, A_2, \dots\}$ ensemble fini ou dénombrable de sous-ensembles flous normalisés, utilisés pour caractériser V .

Exemple : Considérons la surface d'un appartement comme variable V , définie sur l'ensemble X des nombres réels positifs. Un exemple d'ensemble T_V est donné dans la figure 2.



Remarque : T_V correspond à une description qualitative du domaine numérique X . Pour avoir une description exhaustive de X , les ensembles flous A_i doivent former une partition floue. Cela veut dire qu'au moins les deux propriétés suivantes doivent être vérifiées :

- $\forall x \in X : \max A_i(x) > 0$,
- $\forall x \in X, i \neq j : \min (A_i(x), A_j(x)) < 1$.

Souvent une partition floue est définie par une condition plus restrictive [DPR 91] :

$$\forall x \in X : \sum_i A_i(x) = 1.$$

2.2. Propositions floues :

A. Définition de proposition floue :

Définition 2.2.1 : On appelle *modificateur linguistique* un opérateur mod qui permet à partir de toute caractérisation floue A de V , de produire une nouvelle caractérisation floue $\text{mod}(A)$. Si la fonction d'appartenance de A est f_A , celle de $\text{mod}(A)$ est $f_{\text{mod}(A)} = t_{\text{mod}(A)}(f_A)$, obtenue par l'intermédiaire d'une transformation mathématique.

On note $M(T_V)$ l'ensemble des caractérisations obtenues à partir de T_V , en utilisant des modificateurs linguistiques.

Exemples : Dans le tableau ci-dessous, on donne quelques exemples de modificateurs linguistiques. On a choisi trois exemples représentant trois types différents, selon qu'ils renforcent la caractérisation, qu'ils l'affaiblissent ou encore qu'ils réduisent le flou.

Désignation de l'opérateur	Transformation	Terme linguistique	Visualisation de l'opérateur
Compression	$f_{\text{très}(A)} = (f_A)^2$	Très	
Dilatation	$f_{\text{peu}(A)} = \sqrt{f_A}$	Peu	
Réduction du Flou	$\begin{cases} 2.(f_A)^2 & \text{si } f_A \leq 1/2 \\ 1 - 2.(1 - f_A)^2 & \text{si } 1/2 \leq f_A \end{cases}$	Plus près de	

Tab. 2 - Exemples de modificateurs linguistiques

Définitions 2.2.2 :

- Une **proposition floue élémentaire** est définie à partir d'une variable linguistique (V, X, T_V) par la qualification " V est A ", avec : $A \in T_V$ ou $A \in M(T_V)$. La valeur de vérité de la proposition floue élémentaire " V est A " est définie par la fonction d'appartenance f_A de A .
- L'utilisation conjointe de propositions floues élémentaires " V est A ", " W est B ",...pour des variables $V, W \dots$, supposées non-interactives, en utilisant des connecteurs comme la conjonction, la disjonction et l'implication, forme une **proposition floue**.
- Une **proposition floue générale** est obtenue par conjonction, disjonction, négation et implication de propositions floues quelconques.
- Une **règle floue** est une proposition floue de la forme " Si p alors q " utilisant une implication entre deux propositions floues quelconques p et q .

B. Distribution de possibilités associée à une proposition floue :

En introduisant la notion de variable linguistique et de proposition floue, on a étendu une forme d'énoncé naturel en logique et on a fait donc un pas vers le raisonnement en environnement flou. Les fonctions d'appartenance utilisées jusque là pour caractériser des imprécisions où des concepts vagues, permettent de gérer les incertitudes en les interprétant comme des distributions de possibilités.

Définition 2.2.3 : Etant donnée une proposition floue élémentaire " V est A ", une distribution de possibilités lui est *associée*, donnée par : $\pi_{v,A}(x) = f_A(x)$.

La distribution de possibilités associée à une proposition floue générale peut être calculée en appliquant les principes possibilistes énoncés dans le paragraphe 1.5. et des opérations ensemblistes de la théorie des ensembles flous, les distributions de possibilités étant considérées comme des fonctions d'appartenance. La conjonction, la disjonction et l'implication sont modélisées par des relations floues.

Souvent, dans le langage naturel, on utilise des qualifications linguistiques ou des quantificateurs symboliques, agissant sur des propositions floues. Ces opérations induisent des distributions de possibilités de différentes natures.

1. Utilisation des qualifications linguistiques :

Dans le langage naturel, un important mécanisme de modification du sens d'une proposition P est l'adjonction de trois types de qualifications linguistiques [DPR 91] :

- de vérité : " P est très vraie ", ...
- de probabilité : " P est peu probable ", ...
- de possibilité et de nécessité : " P est possible ", ...

Les règles gouvernant ces qualifications sont données dans [ZAD 78].

- **Qualifications de vérité :**

Soit une proposition floue de la forme : " P est τ ", où :

P : " x est A ", une proposition floue élémentaire,

τ est un sous-ensemble flou de l'intervalle $[0,1]$, représentant une qualification linguistique de vérité comme, par exemple : *très vraie, peu vraie, ...*

En se basant sur le principe de spécificité minimale, on peut établir que la distribution de possibilités qui en découle est donnée par [DPR 91] :

$$\mu_B(u) = \mu_\tau(\mu_A(u)),$$

Cela est équivalent à interpréter la proposition floue P comme la proposition floue "(P est A) est vraie", et lui appliquer le modificateur linguistique représenté par τ .

- **Qualifications de probabilité :**

On peut de même utiliser des qualifications de probabilité comme : *probable, peu probable, ...* sur des propositions floues de la forme : " x est A ", donnant lieu à une proposition de la forme :

$$"(x \text{ est } A) \text{ est } \tau",$$

où τ est une qualification de probabilité, représentée par un sous-ensemble flou de l'intervalle $[0,1]$.

Cette dernière proposition étant sémantiquement équivalente à la proposition :

$$Prob \{(x \text{ est } A)\} = \tau,$$

τ induit alors une distribution de possibilités sur les valeurs que peut prendre la probabilité de l'événement flou A (voir chapitre 3).

- **Qualifications de possibilité et de nécessité :**

Pour pouvoir interpréter des propositions de la forme : " x est A est très possible ", on va adopter le point de vue de Zadeh et Bellmann, qui ont suggéré que la notion de vérité est locale au lieu d'être absolue [DPR 91]. Cela signifie qu'une proposition ne peut être

considérée comme vraie que par rapport à une autre proposition, c'est à dire, dans son voisinage.

Cela dit, une expression de la forme :

" (x est A) est τ - possible ", où τ est sous-ensemble flou de l'intervalle [0,1].

peut être interprétée par : " $\exists B$, "x est B" est vraie (sûre) et $\mu_B = f(\mu_\tau, \mu_A)$ ".

Ainsi, la détermination de la distribution de possibilités associée revient à déterminer celle de la proposition sûre B.

Par exemple, pour déterminer la distribution de possibilités associée à la proposition "x est A est α - certaine", où A est une partie non floue, on doit résoudre l'équation fonctionnelle suivante par rapport à B :

$$N(A) = \inf_{u \notin A} [1 - \mu_B(u)] = \alpha.$$

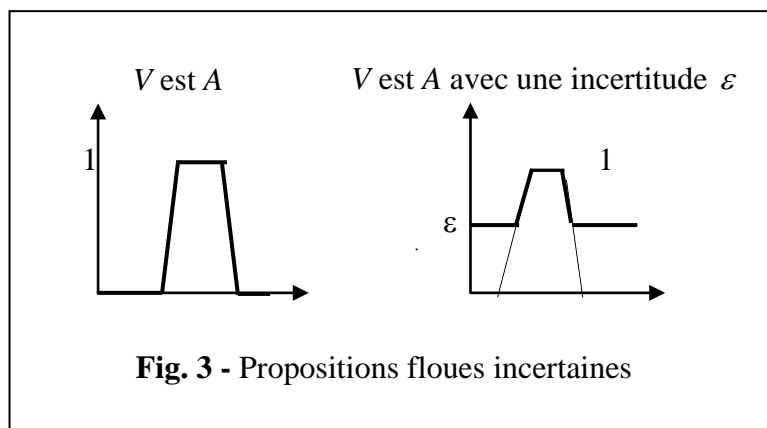
Il faut donc trouver la proposition sûre B (fonction d'appartenance) telle que la nécessité de A, sachant que " x est B ", soit égale à α .

En vertu du principe de spécificité minimale : $\mu_B(u) = \max. (\mu_A(u), 1-\alpha)$.

Notons que cette égalité demeure inchangée si l'on remplace l'égalité $N(A) = \alpha$, par l'inégalité $N(A) \geq \alpha$, donnant lieu à la qualification équivalente :

" (x est A) est certaine au moins au degré α ".

Ce résultat s'étend pour des parties floues.



Autres formes de qualifications de possibilités et de nécessités peuvent être trouvées dans [DPR 91].

2. Utilisation des quantificateurs linguistiques :

En logique classique, deux quantificateurs seulement sont utilisés : \forall et \exists . On introduit en logique floue des quantificateurs flous décrivant des situations intermédiaires [ZAD 79].

Définitions 2.2.4 : Un *quantificateur flou* Q est un sous-ensemble flou de \mathfrak{R} , qui décrit un nombre de cas approximatif.

Pour pouvoir représenter des expressions telles que : " Dans plus de la moitié des cas, on a P ", il est nécessaire de définir la cardinalité d'un sous-ensemble flou, c'est-à-dire, le nombre ou la proportion des éléments de l'ensemble de référence qui vérifient P .

Soit $U = \{u_1, u_2, u_3, \dots, u_n\}$, un ensemble de référence, fini.

Le nombre d'éléments d'un sous-ensemble flou F de U , qui est naturellement un ensemble flou de l'ensemble des entiers naturels N , est défini par [ZAD 79] :

$$|F|_f = \sum_{i=1}^p \frac{\alpha_i}{\text{card}(F^{\alpha_i})}, \text{ (notation 2.1, chap. 1) , où :}$$

- α_i est le supremum des valeurs $\alpha \in [0, 1]$ donnant lieu à la même α -coupe F^α . U étant fini, α_i sont en nombre fini.
- $\text{Card.}(F^\alpha)$ est le cardinal de la α -coupe de F qui est un ensemble ordinaire.

Néanmoins, il est plus convenable d'exprimer la cardinalité d'un sous-ensemble flou par un nombre crisp. Dans ce cas, le concept de la puissance d'un sous-ensemble flou peut être utilisé comme une caractérisation numérique de sa cardinalité [ZAD 79] :

$$|F| = \sum_{i=1}^n \mu_A(u_i),$$

On peut caractériser la proportion des éléments de U appartenant à F par le nombre :

$$p = \frac{|F|}{n}.$$

Ainsi, une proposition floue de la forme : " QU est F ", Q étant un quantificateur flou, induit une distribution de possibilités sur la proportion des éléments de U appartenant à F , donnée par la fonction d'appartenance de Q .

Par ailleurs, la valeur de vérité de la proposition floue quantifiée " QU est B ", sachant que " U est B ", est : $f_Q(p)$ [BOU 95].

Des résultats similaires peuvent être énoncés dans le cas où U est continu [ZAD 79].

3. Raisonnement Approximatif

Dans les paragraphes précédents, on a traité le problème de la représentation des connaissances symboliques, vagues ou imprécises, sous forme de distributions de possibilités. Dans ce qui suit, on va aborder un sujet central, celui de l'inférence avec des connaissances imparfaites, représentées par des distributions de possibilités.

La logique floue a été présentée par Zadeh comme un cadre du *raisonnement approximatif*, une théorie mathématique basée sur la logique floue et ayant pour objet l'étude des méthodes d'inférence à partir de prémisses imprécises et/ou incertaines [ZAD 79]. Les règles d'inférence sont approximatives au lieu d'être exactes et les prémisses sont des propositions floues.

La logique¹ floue est une extension de la logique classique. Les propositions sont des propositions floues définies à partir d'un ensemble L de variables linguistiques et d'un ensemble M de modificateurs linguistiques. Leur valeur de vérité appartient à tout l'intervalle $[0,1]$. Elle est fournie par la fonction d'appartenance de la caractérisation floue utilisée dans la proposition floue. Des quantificateurs flous (ex. "généralement"), de vérité (ex. " P est très vrai "), de probabilité (ex. " P est peu probable ") et de possibilité (ex. " P est très possible ") peuvent être utilisés. Dans le cas particulier où toutes les propositions floues sont booléennes, la logique floue est identique à la logique classique.

Différents systèmes formels pour la logique floue ont été proposés. On en cite par exemple les travaux de Pavelka [PAV 79] pour la formalisation de la logique floue propositionnelle, et ceux de Novak [NOV 89] pour la logique floue du premier ordre. Une synthèse des différents systèmes formels de la logique floue, proposés dans la littérature, peut être trouvée dans [DLP 91].

Les principales règles d'inférence dans le raisonnement approximatif sont la projection, la combinaison/projection, la règle compositionnelle et, comme cas particulier de la règle compositionnelle, le modus ponens généralisé. Les deux premières règles étant déjà traitées dans le cadre de la théorie des ensembles flous, les distributions de possibilités pouvant être interprétées comme des fonctions d'appartenance, on va étudier le modus ponens généralisé.

3.1. Implications floues :

Définition 3.1 :

- L'*implication floue* entre deux propositions floues élémentaires " V est A ", et " W est B ", est une relation floue définie sur $X \times Y$, dont la valeur de vérité est donnée par : $f_R(x, y) = \phi(f_A(x), f_B(y))$, avec $\phi : [0,1] \times [0,1] \rightarrow [0,1]$.

¹ Le mot logique est utilisé ici dans le sens de : " méthodes de raisonnement".

- L'implication floue entre deux propositions floues quelconques se définit de la même manière en considérant les distributions de possibilités associées aux propositions floues en question.

Remarque : La fonction ϕ , sauf exception, est choisie de telle sorte que l'implication floue soit identique à l'implication classique, dans le cas où A et B sont crisp.

Classes générales d'implications floues :

Il n'existe pas une façon unique de généraliser l'implication de la logique classique. Selon le point de départ choisi, on obtient plusieurs classes d'implications floues possédant des propriétés communes. Les principales implications floues sont indiquées dans le tableau 3 [BOU 95].

Non	Notation	Valeur de vérité
Reinchenbach (probabiliste)	R^R	$1 - f_A(x) + f_A(x).f_B(y)$
Willmott (Zadeh)	R^W	$Max(1 - f_A(x), Min(f_A(x), f_B(y)))$
Rescher-Gaines (sticte-standard)	R^{RG}	$\begin{cases} 1 & \text{si } f_A(x) \leq f_B(y), \\ 0 & \text{sin on .} \end{cases}$
Kleene-Dienes	R^{KD}	$Max(1 - f_A(x), f_B(y))$
Brouwer-Godel	R^{BG}	$\begin{cases} 1 & \text{si } f_A(x) \leq f_B(y), \\ f_B(y) & \text{sin on .} \end{cases}$
Goguen	R^G	$\begin{cases} 1 & \text{si } f_A(x) = 0 \\ Min(f_B(y) / f_A(x), 1) & \text{sin on .} \end{cases}$
Yager	R^Y	$B(y)^{A(x)}$
Lukasiewicz	R^L	$Min(1 - f_A(x) + f_B(y), 1)$
Mamdani *	R^M	$Min(f_A(x), f_B(y))$
Larsen *	R^P	$f_A(x).f_B(y)$
* Ces quantités ne généralisent pas l'implication de la logique classique		

Tab. 3 – Principales implications floues

Ces implications peuvent être réparties principalement en trois classes [DPR 91] :

- **S - implications** : A partir de la formule classique $p \rightarrow q = \neg p \vee q$, on passe de la négation à une négation floue et de la disjonction à une disjonction floue. On définit ainsi la 1^{ère} classe par :

$$f_R(x, y) = S(n(f_A(x)), f_B(y)),$$

où S est une t-conorme et n une négation involutive.

Exemples : En choisissant la négation $n(u) = 1 - u$, on obtient :

$$S = \text{Max.} \rightarrow R^{KD}, S = \text{Probabiliste} \rightarrow R^R, S = \text{Luckasiewicz} \rightarrow R^L.$$

- **R - implications (Résiduelles)** : En se basant sur l'idée que l'implication reflète un certain ordre des propositions, c'est-à-dire que si $p \rightarrow q$, alors la valeur de vérité de p est inférieure à celle de q , on introduit la 2^{ème} classe d'implications floues :

$$f_R(x, y) = \text{Sup}\{w \in [0,1], T(f_A(x), w) \leq f_B(y)\},$$

où T est un t-norme.

Exemples : $T = \text{Min.} \rightarrow R^{BG}, T = \text{Luckasiewicz} \rightarrow R^L, T = \text{Produit} \rightarrow R^G.$

- **QL - implications** : En prenant pour définition de l'implication floue l'extension de $\neg p \vee (p \wedge q)$, on définit la 3^{ème} classe d'implications floues :

$$f_R(x, y) = S[n(f_A(x)), T(f_A(x), f_B(y))], \text{ où } n \text{ est une négation floue.}$$

Exemple : En choisissant la conjonction et la disjonction de Zadeh, on obtient R^{KD} .

Remarque : Il est possible de construire d'autres classes d'implications floues, nous avons présenté ici les plus importantes. Elles ne sont pas disjointes; certaines implications peuvent appartenir à plusieurs classes, comme celle de Kleene-Dienes par exemple.

3.2. Modus Ponens généralisé :

Le raisonnement approximatif a été introduit pour permettre d'effectuer automatiquement des déductions qui seraient fournies par des acteurs humains suite d'un raisonnement naturel. Par exemple, une règle floue telle que "Si la surface est grande, alors le prix est élevé" doit par exemple pouvoir être utilisée pour un appartement particulier, dont on connaît précisément la surface, même si celle-ci n'est pas vraiment typique de la caractérisation "grande" et il est intéressant d'obtenir une indication sur le prix, même si la surface n'est que relativement grande par exemple. Dans la logique

classique, le modus ponens¹ (règle de détachement) ne permettait d'obtenir une conclusion que si l'on savait exactement que la surface de l'appartement considéré est grande.

Le schéma d'application du Modus Ponens Généralisé (MPG) est le suivant :

Règle floue :	Si "V est A" Alors "W est B"
Fait observé :	V est A'
Conclusion :	W est B'

où $f_{B'}(y) = \sup_{x \in X} T(f_A(x), f_R(x, y))$ (composition $T - \sup$).

On considère des propositions floues élémentaires pour simplifier les notations, mais le même schéma s'applique pour des propositions floues quelconques.

Pour être compatible avec la logique classique, le modus ponens généralisé doit être identique au modus ponens classique dans le cas où la description de l'observation est précisément celle donnée dans la prémisse de la règle, c'est-à-dire $A = A'$. On doit, dans ce cas, obtenir une conclusion B' égale à B .

Définition 3.2 : Une norme triangulaire T est un opérateur de modus ponens généralisé compatible avec l'implication floue R si, quelles que soient les fonctions d'appartenance f_A et f_B , elle vérifie :

$$\forall y \in Y, f_B(y) = \sup_{x \in X} T(f_A(x), f_R(x, y)).$$

Le tableau 2 indique les principaux choix possibles.

Opérateur de MPG T	Implication floue R
$T(u, v) = \max(u + v - 1, 0)$	$R^W, R^{KD}, R^{RG}, R^{BG}, R^R, R^L, R^M, R^G.$
$T(u, v) = \min(u, v)$	R^{RG}, R^{BG}, R^M, R^P
$T(u, v) = u.v$	$R^{RG}, R^{BG}, R^G, R^M, R^P$
T quelconque	Implication Résiduelle construite à partir de T
Tab. 4 - Principaux opérateurs de MPG et les implications compatibles	

¹ Le modus ponens est un mot Latin : "modus = méthode", "ponens = affirmation". C'est la règle (méthode) classique qui permet d'affirmer B à partir de (A), sachant que $(A \Rightarrow B)$.

3.3. Comportement des implications floues :

On va étudier dans ce paragraphe le comportement des implications floues pour diverses formes de faits observés, pour permettre de guider le choix de l'une d'entre elles en présence d'une application particulière telle qu'un système expert ou un système de commande de processus [BOU 95].

- **Observation plus spécifique ou plus précise que la prémisse : $A' \subseteq A$**

La plupart des implications floues donnent effectivement un résultat identique à la conclusion de la règle ($B = B'$), se comportant d'une façon prudente puisque aucune indication n'est en général donnée relativement à une liaison entre les variations de la caractérisation floue de V et de celle de W .

L'exception concerne R^{RG} qui fournit un résultat plus précis que la conclusion de la règle ($B' \subseteq B$), modélisant ainsi une règle graduelle du type " Plus V est A plus W est B ".

- **Observation moins spécifique ou moins précise : $A \subseteq A'$**

On ne peut pas s'attendre, dans ce cas, à obtenir une conclusion aussi nette que celle proposée dans la règle, puisque les conditions d'utilisation de celle-ci ne sont pas tout à fait satisfaites. La plupart des implications floues donnent effectivement un résultat moins spécifique, moins précis, éventuellement moins certain que la conclusion de la règle ($B \subseteq B'$), adoptant une attitude de prudence conforme au cheminement naturel de pensée.

Les implications R^M et R^P ont un comportement différent et préservent la conclusion de la règle, ce qui est acceptable dans certains cas comme, par exemple, une règle que l'on peut appeler alerte, telle que "Si l'augmentation de pression est grande, le délai d'intervention doit être court", pour laquelle une observation de la forme " l'augmentation de pression est relativement grande" peut raisonnablement conduire à un délai d'intervention court, plutôt que relativement court.

- **Observation précise et spécifique : $A' = \{x_1\}$**

Dans ce cas, l'opérateur de MPG ne joue aucun rôle, car la forme de la fonction d'appartenance de la conclusion prend la forme $\forall y \in Y : f_{B'}(y) = f_R(x_1, y)$.

Dans le cas où la prémisse est elle-même précise et spécifique ($A = \{x_0\}$), on obtient naturellement comme résultat la conclusion de la règle B si $x_1 = x_0$; sinon, on obtient une conclusion indéterminée ($f_{B'}(y) = 1$) pour toutes les implications indiquées, sauf R^M , R^P qui donnent $B' = \phi$.

Dans le cas où la prémisse n'est pas précise, on obtient une conclusion d'autant plus proche de B que l'observation est fortement compatible avec la prémisse.

Remarque : Cette situation est importante car elle est classique dans la commande de processus, les faits disponibles étant des mesures produites par des capteurs.

- **B précise, spécifique et certaine :** $B = \{y_0\}$

Cette situation est très fréquente et correspond à des règles de diagnostic ou de décision comme "Si la température extérieure est élevée alors couper le chauffage". La conclusion B' obtenue est généralement B elle-même, pondérée par une incertitude ε .

Le tableau suivant résume les propriétés des implications floues [BOU 95].

Règle : <i>Si la surface est grande alors le prix est élevé</i>		
Situation	Majorité des implications	Exceptions
<i>A' plus précis que A Surface très grande</i>	<i>B' conservée Prix élevé</i>	<i>B' plus précis que B (R^{RG}) prix nettement élevé</i>
<i>A' moins précis que A surface plutôt grande</i>	<i>B' moins précis que B prix relativement élevé</i>	<i>B conservée (R^M, R^P) prix élevé</i>
<i>A' précis La surface est de 87m²</i>	<i>B' normalisée</i>	<i>B' non normalisée (R^W, R^M, R^P)</i>
<i>B précis (décision) Si la surface est grande alors acheter</i>	<i>B' = B avec une incertitude il est relativement certain qu'il faille acheter</i>	<i>B' = B dès que A' et A non incompatibles (R^M, R^P)</i>
<i>A' contraire de A Surface n'est pas grande</i>	<i>B' indéterminée le prix est inconnu</i>	<i>B' = ϕ ou B' = m(B) (R^M, R^P) aucun prix envisageable</i>

<p>$A' = A$ avec une incertitude <i>la surface est grande(certitude 0.8)</i></p>	<p>$B' = B$ avec la même incertitude <i>le prix est élevé (certitude 0.8)</i></p>	<p>B conservée(R^M, R^P) <i>Le prix est élevé</i></p>
<p>Tab. 5 - Propriétés comparées des implications floues</p>		

Chapitre III

Mathématiques floues

" On fait la science avec des faits, comme on fait une maison avec des pierres : mais une accumulation de faits n'est pas plus une science qu'un tas de pierres n'est une maison."

Henri Poincaré, mathématicien français

Le but de ce chapitre est de donner des exemples de l'investigation de la logique floue dans les domaines classiques des mathématiques. On a choisi les branches les plus abordées dans la littérature.

1. Arithmétique floue :

La notion de quantité floue est un paradigme puissant pour la représentation des imprécisions dans les informations numériques. L'idée que les quantités floues peuvent être arithmétiquement combinées suivant les lois de la théorie des sous-ensembles flous est due à Zadeh [ZAD 75].

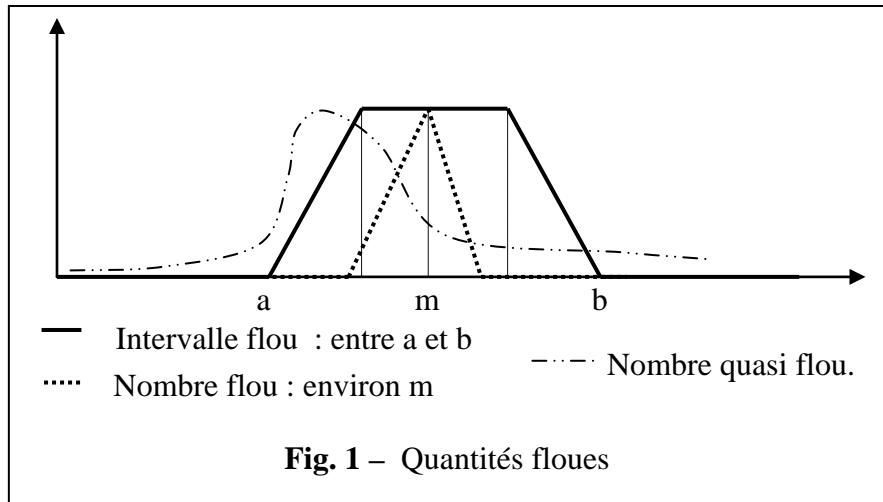
Ce paragraphe introduit les notions principales relatives aux intervalles flous et à leurs manipulations ; des études plus détaillées et plus complètes peuvent être trouvées dans [DPR 02].

1.1. Quantités floues, intervalles flous, nombres flous :

On résume ici les définitions de base des quantités floues, servant à définir l'arithmétique floue.

- Une *quantité floue* est un sous-ensemble flou normalisé de \mathfrak{R} . Une valeur de degré d'appartenance 1 est dite *valeur modale*.
- Un *intervalle flou* est une quantité floue convexe.

- Un **nombre flou** est un intervalle flou de fonction d'appartenance semi-continue supérieure et de support borné, admettant une valeur modale unique.
- Un **nombre quasi-flou** est un intervalle flou de fonction d'appartenance semi-continue supérieure, nulle à l'infini, admettant une valeur modale unique.



Le calcul des quantités floues est une application du principe d'extension. Les imperfections dans les connaissances (imprécisions, flou) peuvent être propagées dans un calcul arithmétique sous forme de degrés de possibilité, donc suivant les règles de la théorie des possibilités [DPR 87].

1.2. Opérations arithmétiques :

Il est souvent nécessaire d'effectuer des additions ou des produits de nombres imprécisément connus. Par exemple, si la largeur du jardin est approximativement 50 mètres et sa longueur environ 80 mètres, si le prix du grillage est aux alentours de 140 dinars le mètre, à combien me reviendra à peu près la clôture du jardin ?

1.2.1. Opérations unaires :

Définition 1.2.1 : Soit φ une opération unaire définie sur \mathfrak{R} .

Une **opération unaire floue** Δ fait correspondre, à toute quantité floue Q , une autre quantité floue ΔQ de fonction d'appartenance :

$$\forall z \in \mathfrak{R} : f_{\Delta Q}(z) = \text{Sup}_{\{x / z = \varphi(x)\}} f_Q(x)$$

Cas particuliers :

<i>Opération</i>	$\varphi(x)$	ΔQ	$f_{\Delta Q}(z)$
<i>Opposé</i>	$-x$	$-Q$	$f_Q(-z)$
<i>Multiplication par un scalaire</i>	$a \cdot x$	$a \cdot Q$	$f_Q(z/a), a \neq 0$
<i>Inverse</i>	$1/x$	$1/Q$	$f_Q(1/z)$
<i>Puissance</i>	x^p	Q^p	$f_Q(z^{1/p}), p \neq 0$
<i>Exponentiel</i>	e^x	e^Q	$f_Q(\log z), z > 0$

Par exemple, l'opposé du nombre flou « environ 4 » peut être exprimé comme « environ -4 » et son inverse comme « environ $1/4$ ».

1.2.2. Opérations binaires :

Définition 1.2.2 : Soit ψ une opération binaire définie sur \mathfrak{R} .

Une *opération binaire floue* \diamond fait correspondre à deux quantités floues Q et Q' , une autre quantité floue $Q \diamond Q'$ de fonction d'appartenance :

$$\forall z \in \mathfrak{R} : f_{Q \diamond Q'}(z) = \text{Sup}_{\{(x,y) / z = \psi(x,y)\}} \min(f_Q(x), f_{Q'}(y)).$$

Propriété : Si l'opération ψ est commutative (resp. associative), \diamond l'est aussi.

Cas particuliers : Les opérations binaires floues sur les quantités floues Q et Q' associées aux opérations arithmétiques classiques ont pour résultats :

$$Q \oplus Q', Q \ominus Q', Q \otimes Q', Q \oslash Q', \underline{\max.}(Q, Q'), \underline{\min}(Q, Q').$$

Les fonctions d'appartenance peuvent s'écrire sous la forme :

- $\forall z \in \mathfrak{R} : f_{Q \oplus Q'}(z) = \text{Sup}_x \min(f_Q(x), f_{Q'}(z-x)).$
- $\forall z \in \mathfrak{R} : f_{Q \ominus Q'}(z) = \text{Sup}_x \min(f_Q(z+x), f_{Q'}(y)).$
- $\forall z \in \mathfrak{R} : f_{Q \otimes Q'}(z) = \text{Sup}_x \min(f_Q(x), f_{Q'}(z/x)).$
- $\forall z \in \mathfrak{R} : f_{Q \oslash Q'}(z) = \text{Sup}_x \min(f_Q(y \cdot z), f_{Q'}(y)).$

Propriétés : Pour deux quantités floues Q et Q' , on a [BOU 95] :

- $Q \oplus Q' = Q \oplus (-Q')$,
- $Q \otimes Q' = Q \otimes (1 / Q')$,
- $Q \otimes Q' = (-Q) \otimes (-Q')$,
- $(-Q) \otimes Q' = Q \otimes (-Q') = -(Q \otimes Q')$,
- $Q \oplus 0 = 0 \oplus Q = Q$,
- $Q \otimes 1 = 1 \otimes Q = Q$,
- $Q \otimes 0 = 0 \otimes Q = 0$.

Remarques :

- L'addition et la multiplication floues sont des opérations internes dans l'ensemble des intervalles flous.
- La multiplication floue \otimes n'est généralement pas distributive par rapport à l'addition floue \oplus , mais l'inégalité suivante est vérifiée :

$$Q \otimes (Q' \oplus Q'') \leq (Q \otimes Q') \oplus (Q \otimes Q''),$$

qui devient une égalité dans des cas particuliers, par exemple si Q' et Q'' sont des nombres flous dont les supports sont tous les deux inclus dans \mathfrak{R}^+ ou dans \mathfrak{R}^- .

1.3. Calcul pratique des quantités floues non - interactives :

Les opérations floues que nous venons de définir peuvent être relativement difficiles à mettre en œuvre ; plusieurs méthodes ont été proposées pour alléger les calculs. Elles peuvent être regroupées principalement en deux catégories [DPR 02] : des méthodes basées sur des représentations paramétrées, et des méthodes approximatives utilisant les coupes de niveaux et le calcul d'intervalles classique.

1.3.1. Représentation paramétrique :

- **Nombres flous de type L-R :**

Définition 3.1.1 : Soient trois paramètres réels m, a, b, a et b étant strictement positifs ; deux fonctions, notées L et R , définies sur \mathfrak{R}^+ , à valeurs dans l'intervalle $[0,1]$, telles que :

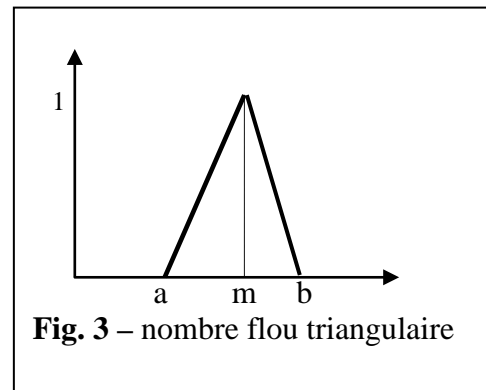
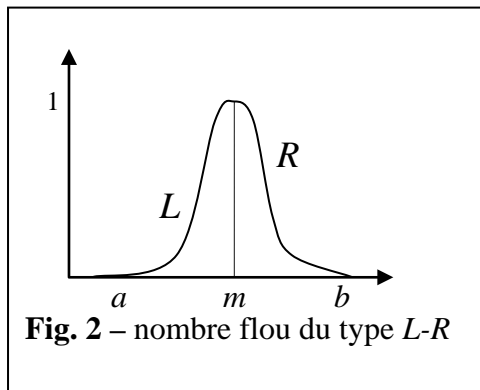
- L et R sont s. c. s.,
- $L(0) = R(0) = 1$,
- $L(1) = 0$ ou $L(x) > 0 \forall x$, avec $\lim_{x \rightarrow +\infty} L(x) = 0$,
- $R(1) = 0$ ou $R(x) > 0 \forall x$, avec $\lim_{x \rightarrow +\infty} R(x) = 0$,

Un nombre flou M est dit du **type L-R** si sa fonction d'appartenance f_M est définie par :

$$f_M(x) = \begin{cases} L((m-x)/a) & \text{si } x \leq m \\ R((x-m)/b) & \text{si } x > m \end{cases}$$

Notation : On note $M = (m, a, b)_{LR}$ un nombre flou du type L-R, m est sa *valeur modale*.

Cas particulier : Pour un choix particulier de L et de R (affines), on obtient un nombre flou triangulaire (figure 3).



Propriétés : Etant donné deux nombres flous du même type L-R, $M = (m, a, b)_{LR}$ et $N = (n, c, d)_{LR}$, les opérations floues de base donnent les nombres flous suivants :

- $-M = (-m, a, b)_{LR}$
- $M \oplus N = (m+n, a+c, b+d)_{LR}$
- $M \ominus N = (m-n, a+d, b+c)_{LR}$ si $L = R$.
- $M \otimes N$ n'est, par contre, généralement pas du type L-R.

- **Intervalles flous de type L-R :**

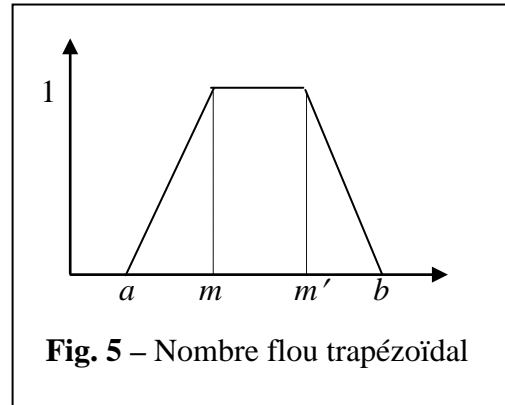
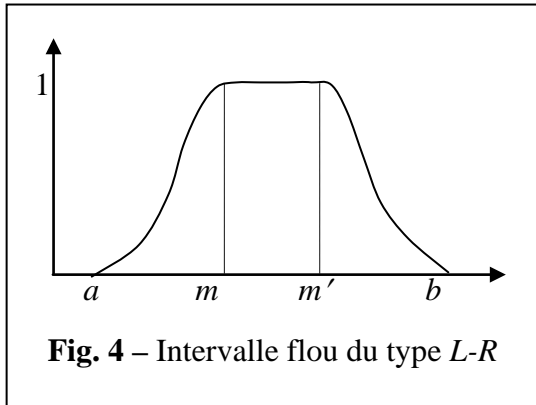
Définition 1.3.2 : Soient quatre paramètres réels m, m', a, b , a et b étant strictement positifs ; deux fonctions, notées L et R , comme définies en 3.1.1

Un intervalle flou M est dit du **type L-R** si sa fonction d'appartenance f_M est définie par :

$$f_M(x) = \begin{cases} L((m-x)/a) & \text{si } x \leq m \\ 1 & \text{si } m < x < m' \\ R((x-m)/b) & \text{si } x \geq m' \end{cases}$$

Notation : On note $M = (m, m', a, b)_{LR}$ un intervalle flou de type L-R.

Cas particulier : Pour un choix particulier de L et de R (affines), on obtient un nombre flou trapézoïdal (figure 5)



Propriétés : Etant donné deux intervalles flous du même type $L-R$, $I = (m, m', a, b)_{LR}$ et $J = (n, n', c, d)_{LR}$. Les opérations floues de base donnent les intervalles flous suivants :

- $-I = (-m', -m, a, b)_{LR}$
- $M \oplus N = (m+n, m'+n', a+c, b+d)_{LR}$
- $M \ominus N = (m-n', m'-n, a+d, b+c)_{LR}$ si $L = R$.

$M \otimes N$ n'est, par contre, généralement pas du type $L-R$.

1.3.2. Utilisation des α - coupes :

Soient M et N deux quantités floues, f une fonction réelle à deux variables réelles.

L'ensemble flou $f(M, N)$ peut être défini, dans certains cas, explicitement en termes des coupes de niveaux de M et N . Pour ce faire, on applique le résultat suivant [DPR 87] :

Résultat fondamental : Plusieurs conditions suffisantes ont été proposées pour assurer que :

$$[f(M, N)]_\alpha = f(M_\alpha, N_\alpha), \quad \alpha \in]0,1].$$

Par exemple :

- f est continue, M et N sont à fonctions d'appartenance s. c. s. (semi-continues supérieurement), et à supports bornés.
- f continue, isotonique (monotone croissante par rapport à chaque argument), M et N des intervalles flous de fonctions d'appartenance s. c. s., tels que leurs α - coupes sont toutes différentes de \mathfrak{R} .

Ce résultat exprime le fait que, sous certaines conditions suffisantes, l'opération floue sur M et N peut être transformée en une opération classique sur les α -coupes, suivant les lois de l'analyse des intervalles, l'ensemble flou du résultat pouvant être reconstruit à partir de ses coupes de niveaux.

- **Calcul approché en utilisant les coupes de niveaux :**

Les méthodes analytiques présentées ci-dessus ont une applicabilité limitée en pratique [DPR 02]. Quand $f(M, N)$ ne peut être explicitement calculé par une méthode analytique, on peut quand même obtenir un échantillonnage de sa fonction d'appartenance en utilisant des coupes de niveaux et en appliquant le calcul des intervalles, tirant ainsi parti du résultat fondamental [DPR 02].

1.4. Comparaison des quantités floues :

Cette question a fait l'objet de nombreuses recherches [DPR 02]. En effet, ce problème est important en pratique : quand on utilise des intervalles flous pour évaluer des situations, ils doivent ensuite être comparés afin de faire un choix et de prendre une décision. Trois méthodes permettant de comparer des quantités floues sont particulièrement importantes [DPR 02].

1.4.1. Classement des intervalles flous par défuzzification :

Une méthode simple pour classer des intervalles flous consiste à associer à chaque intervalle flou une valeur réelle précise (défuzzification) qui le "résume", et qu'il est alors aisé d'ordonner. Cela dit, on peut comparer deux intervalles flous en comparant les valeurs réelles associées. L'avantage qu'offre cette méthode est que l'on obtient toujours facilement un ordre total.

1.4.2. Méthodes basées sur un objectif flou :

Cette classe de méthodes consiste à comparer des intervalles flous à ordonner à un même ensemble flou de référence, vu comme un objectif. Cet ensemble flou peut être interpréter de deux façons : soit compris comme une valeur floue idéale, et alors le critère de comparaison est la distance à cet ensemble, soit il peut être vu comme une contrainte élastique et la comparaison se fait alors suivant le degré de satisfaction de cette contrainte pour chacun des intervalles flous.

1.4.3. Relations d'ordre flou entre intervalles flous :

Quand le but de la comparaison des intervalles flous est d'offrir une vision nuancée de leurs positions relatives, il est naturel de construire un ordre flou sur les intervalles flous et d'exploiter cette relation floue afin d'effectuer un choix. Ceci peut cependant être incompatible avec la capacité à effectuer un choix définitif dans certaines circonstances. Différents types de relations sont présentés dans [DPR 02].

1.5. Distance entre nombres flous :

La notion de distance entre objets jouit d'une grande importance notamment en analyse des données [HEI 97]. Il est dès lors intéressant de pouvoir évaluer les distances entre points de l'espace euclidien, même imprécisément connus.

Une méthode simple utilise les valeurs espérées des nombres flous. On définit une distance entre deux nombres flous A et B par :

$$\sigma(A, B) = |EV(A) - EV(B)|,$$

où EV est la valeur espérée d'un nombre flou, se calculant en utilisant sa fonction d'appartenance exactement comme l'espérance des variables aléatoires.

Une autre méthode se base sur les distances entre les coupes de niveaux, qui sont des intervalles fermés. On définit une famille de distances entre nombres flous indexée par un paramètre p de la manière suivante :

Soit deux intervalles fermés : $A = [a_1, a_2]$, $B = [b_1, b_2]$.

La distance de Minkowski entre A et B est définie par :

$$d_p(A, B) = \begin{cases} \left[\frac{1}{2} \cdot (|a_1 - b_1|^p + |a_2 - b_2|^p) \right]^{1/p}, & 1 \leq p < \infty. \\ \text{Max.}\{|a_1 - b_1|, |a_2 - b_2|\}, & p = \infty. \end{cases}$$

Ainsi, on peut définir la distance de Minkowski entre deux nombres flous C et D par :

$$d_p(C, D) = \int_0^1 d_p(C_h, D_h) dh.$$

L'espace métrique des nombres flous est complet pour cette distance [HEI 97].

Enfin, des distances plus simples peuvent être définies [KAU 73] :

- Distance de Hamming : $d_1(A, B) = \int_{-\infty}^{+\infty} |\mu_A(x) - \mu_B(x)| dx.$
- Distance Euclidienne : $d_2(A, B) = \left[\int_{-\infty}^{+\infty} |\mu_A(x) - \mu_B(x)|^2 dx \right]^{1/2}.$

1.6. Applications :

Sans prétendre être exhaustif, on indique ici quelques directions principales d'application de l'arithmétique floue (pour plus de détails, voir [DPR 02]).

1.6.1. Application à l'Analyse floue :

Une extension de l'intégrale de Reimann à des fonctions ayant pour valeurs des intervalles flous est proposée par Dubois et Prade dans [DPR 82], où l'on peut trouver des problèmes de la différentiation floue.

1.6.2. Recherche opérationnelle :

Le rôle potentiel des intervalles flous en recherche opérationnelle est apparu très tôt (1980). Dans ce domaine, les quantités floues ont été utilisées dans des problèmes de graphes aussi bien qu'en programmation linéaire (voir paragraphe 3).

1.6.3. Analyse décisionnelle :

L'analyse de la décision semble être un des champs d'application les plus naturels des quantités floues. Très souvent, les décideurs expriment leurs opinions de manières linguistiques plutôt que numériques, et sont alors modélisés par des quantités floues.

2. Mesures et intégrales floues

La notion de mesure floue est une extension de celle de mesure de probabilités. Les mesures floues servent à mesurer le degré de flou au même titre que les mesures de probabilités servent à mesurer l'aléatoire [COO96].

2.1. Mesures floues :

Les mesures ordinaires sont des fonctions d'ensembles, additives. Plus généralement, les mesures floues sont des fonctions d'ensembles, monotones mais pas nécessairement additives.

Définition 2.1 : Soit X un ensemble crisp, B une tribu définie sur X .

On appelle mesure floue toute fonction g de B à valeurs entre 0 et 1, satisfaisant :

- i) **Bornée :** $g(\emptyset) = 0, g(X) = 1$.
- ii) **Monotone :** A, B deux éléments de $B : A \subseteq B \Rightarrow g(A) \leq g(B)$.

Le triplet (X, B, g) est dit : espace mesuré flou.

La notion de fonction mesurable se définit de manière analogue à la définition classique.

La notion de mesure floue peut être vue comme une extension de celle de mesure de probabilité, en affaiblissant les conditions de cette dernière, à savoir la σ -additivité.

Un avantage de ce fait est lorsqu'il s'agit de faire l'agrégation de deux critères par exemple, l'importance des deux critères dans le cadre de la théorie des probabilités ne peut être autre que la somme des importances individuelles, tandis que les mesures floues fournissent une grande (mesure super-additive) ou petite (mesure sub-additive) valeur, permettant ainsi la modélisation de l'interaction entre plusieurs critères de manière flexible [YAG 99].

2.2. Intégrales floues :

Différentes versions floues de l'intégration ont été proposées dans la littérature. On présente ici les deux types les plus répandus : l'intégrale de Sugeno et celle de Choquet [FUK 90]. [COO 96], [COO 97].

2.2.1. Intégrale floue de Choquet :

L'intégrale de Choquet a été introduite par ce dernier en 1953 pour les mesures non additives [COO 96]. Soit X un ensemble ordinaire, μ une mesure floue définie sur X , A une partie mesurable, et f une fonction réelle μ -mesurable et bornée.

Définition 2.2.1 : L'intégrale floue de Choquet de f sur A , par rapport à μ est définie par :

$$\int_A f \, d\mu = \int_{-\infty}^0 [\mu(A \cap \{f \geq x\}) - \mu(A)] \, dx + \int_0^{+\infty} \mu(A \cap \{f \geq x\}) \, dx.$$

Le signe d'intégration dans le terme droit désigne l'intégrale impropre de Reimann.

On peut encore l'écrire sous la forme équivalente :

$$\int f \, d\mu = \int_{-\infty}^{+\infty} x \, d\mu_f(x) \, dx,$$

où : $\mu_f(x) = \mu(\{f < x\})$, l'intégration étant prise au sens de Riemann-Stieltjess.

Si μ est additive, cette intégrale coïncide avec celle de Lebesgue.

Dans le cas discret : $X = \{x_1, x_2, \dots, x_n\}$, l'intégrale de Choquet s'écrit :

$$\int f \, d\mu = \sum_{2 \leq i \leq n} (f(x_i) - f(x_{i-1})) \cdot \mu(A_i).$$

avec $0 \leq f(x_1) \leq \dots \leq f(x_n)$ et $A_i = \{x_i, x_{i+1}, \dots, x_n\}$.

2.2.2. Intégrale floue de Sugeno :

Sugeno a introduit dans [SUG 74] une intégration floue en se basant sur les mesures floues d'une façon similaire à la construction de l'intégrale de Lebesgue en utilisant les mesures classiques.

Soit (X, \mathcal{B}, g) un espace mesuré flou, f une fonction réelle mesurable sur X , à valeurs dans $[0,1]$.

Définition 2.2.2 : L'intégrale floue de Sugeno de la fonction f sur une partie mesurable A de X , par rapport à la mesure floue μ est définie par :

$$\int_A f(x) \circ \mu(\cdot) = \sup_{0 \leq \alpha \leq 1} [\alpha \wedge \mu(A \cap F_\alpha)],$$

où $F_\alpha = \{x / f(x) \geq \alpha\}$.

On note $\int f \circ \mu$ l'intégrale sur X tout entier.

Dans le cas discret : $X = \{x_1, x_2, \dots, x_n\}$, l'intégrale de Sugeno s'écrit :

$$\int f(x) \circ \mu(\cdot) = \max_{1 \leq i \leq n} [f(x_i) \wedge \mu(A_i)],$$

avec $0 \leq f(x_1) \leq \dots \leq f(x_n)$ et $A_i = \{x_i, x_{i+1}, \dots, x_n\}$.

On peut étendre la mesure et l'intégrale floue aux ensembles flous de la façon suivante :

Soit A un ensemble flou de X de fonction d'appartenance h_A .

La mesure floue de A est définie par :

$$\mu(A) = \int h_A(x) \circ \mu(\cdot).$$

L'intégrale floue sur A peut être définie alors par :

$$\int_A f(x) \circ \mu(\cdot) = \int [h_A(x) \wedge f(x)] \circ \mu(\cdot).$$

L'intégrale floue ainsi définie, jouit des propriétés suivantes [FUK 90] :

Propriétés :

1. $0 \leq \int f \circ \mu \leq 1$.
2. Soit $a \in [0,1]$: $\int (a \vee f) \circ \mu = a \vee \int f \circ \mu$,
 $\int (a \wedge f) \circ \mu = a \wedge \int f \circ \mu$.

3. Pour deux fonctions mesurables f_1, f_2 :

$$\int (f_1 \vee f_2) \circ \mu = \int f_1 \circ \mu \vee \int f_2 \circ \mu ,$$

$$\int (f_1 \wedge f_2) \circ \mu = \int f_1 \circ \mu \wedge \int f_2 \circ \mu .$$

4. Pour deux parties E, F de X (ordinaires ou floues) :

$$\int_{E \cup F} f \circ \mu \geq \int_E f \circ \mu \vee \int_F f \circ \mu ,$$

$$\int_{E \cap F} f \circ \mu \leq \int_E f \circ \mu \wedge \int_F f \circ \mu .$$

5. Si $(f_n)_n$ est une suite monotone de fonctions mesurables et $\lim_{n \rightarrow +\infty} f_n = f$, alors :

$$f \text{ est mesurable et } \lim_{n \rightarrow +\infty} \int f_n \circ \mu = \int f \circ \mu .$$

6. Pour une mesure de probabilité P , les deux intégrales, de Lebesgue et de Sugeno existent, et vérifient :

$$|\int f(x) dP - \int f(x) \circ P(\cdot)| \leq 1/4 .$$

Notons enfin qu'il existe d'autres versions d'intégrales floues, dont certaines sont des généralisations de celle de Sugeno. Par exemple, Nguyen et Kreinovich [NKR 98] suggèrent l'utilisation des opérations arithmétiques : $+$, \bullet , en plus des opérations logiques : \wedge , \vee . Ceci permet le traitement des connaissances numériques, les opérations logiques étant convenables pour le traitement des connaissances symboliques (degrés de possibilité et de nécessité).

2.3. Complément : Probabilités d'événements flous

La théorie des probabilités a été étendue par Zadeh [ZAD 68], pour permettre de mesurer les probabilités des événements flous.

Soit un espace probabilisé (Ω, D, P) , où Ω est un ensemble d'événements, D un σ -algèbre de parties de Ω et P une mesure de probabilité sur D .

Définition 2.3 : Un *événement flou* A est sous-ensemble flou de Ω , de fonction d'appartenance f_A mesurable.

2.3.1. La probabilité d'un ensemble flou comme un scalaire :

Zadeh [ZAD 68] a défini la probabilité d'un événement flou A par :

$$P(A) = \int_{\Omega} f_A dP .$$

Ainsi, la probabilité d'un ensemble flou est définie comme étant l'espérance mathématique de la valeur de sa fonction d'appartenance.

2.3.2. Probabilité d'un événement flou comme un ensemble flou :

Soit A un ensemble flou représentant un événement flou, $P(A_\alpha)$ la probabilité de la coupe de niveau α .

Yager suggère de définir la probabilité de l'événement flou A comme le sous-ensemble flou de $[0,1]$:

$$P_Y(A) = \{(P(A_\alpha), \alpha) \mid \alpha \in [0, 1]\},$$

α étant le degré d'appartenance de $P(A_\alpha)$.

L'indice Y de P_Y indique que P_Y est la définition de la probabilité due à Yager qui est différente de celle de Zadeh dénotée par P .

2.4. Applications :

2.4.1. Synthèse et agrégation d'informations floues :

Les mesures floues et les intégrales floues sont convenables pour l'évaluation et la synthèse d'éléments flous, donnant lieu à des applications variées de la décision multicritères à la classification et traitement d'images [DPS 00], [YAG 99].

2.4.2. En Théorie des possibilités :

Dans le contexte de la théorie des possibilités, l'intégrale floue joue un rôle similaire à celui de l'intégrale de Lebesgue en théorie classique de probabilités. L'intégration floue possède une part importante dans au moins trois domaines [COO 95] :

- L'extension des mesures de possibilités à des domaines plus larges,
- La construction des mesures produit à partir des mesures marginales,
- La définition des mesures de possibilités conditionnelles.

2.4.3. En Contrôle :

Le contrôle flou reste le grand absent des domaines d'application de l'intégration floue malgré le rôle important que joue l'intégration classique en théorie classique de contrôle, ce qui est surprenant car le contrôle flou est le domaine d'application principal des systèmes flous [NKR 98].

Nguyen et Kreinovich proposèrent une modification de la définition de Sugeno, permettant son application en contrôle flou, mais cette voie de recherche est encore inexploree [NKR 98].

3. Programmation mathématique floue :

La recherche opérationnelle est l'ensemble des méthodes et techniques rationnelles d'analyse et de synthèse des phénomènes d'organisation, utilisables pour élaborer les meilleures décisions [FAU 74].

La RO couvre les domaines d'application suivants [FAU 74] :

1. Problèmes combinatoires : optimisation des niveaux d'activités, affectation, ...
2. Les domaines de l'aléatoire : files d'attente, gestion de stocks, ...
3. Les domaines de concurrence : politique d'approvisionnement, ...

Le but de ce paragraphe est de présenter quelques exemples de l'apport de la logique floue dans la première catégorie, ou tout simplement : *l'optimisation floue*. On va discuter principalement *la programmation mathématique floue*.

L'origine de l'optimisation floue est l'article de Zadeh et Bellmann [BZA 70], publié en 1970, qui a introduit les concepts de : contrainte floue, objectif flou, décision floue,

3.1. Programmation Linéaire Floue :

Depuis que George Dantzig a développé la méthode du simplexe pour résoudre les problèmes de programmation linéaire (PL) en 1947, l'impact de la PL a été extraordinaire [CFH 96].

Un problème réel comme par exemple une prise de décision, ne peut pas toujours être représenté d'une manière précise. C'est le cas quand l'objectif et/ou les contraintes ne sont connus que d'une manière approximative. Ceci nous amène à utiliser la logique floue pour la représentation et le traitement des imperfections des connaissances, présentes dans les problèmes de la programmation linéaire.

Un problème de PL classique se présente sous la forme [FAU 74] :

$$\begin{array}{ll}
 \text{Maximiser} & C^T x = C_1 \cdot x_1 + C_2 \cdot x_2 + \dots + C_n \cdot x_n \\
 \text{Sous Contraintes} & A \cdot x \leq b. \quad (A_{i1} \cdot x_1 + \dots + A_{in} \cdot x_n \leq b_i, \quad i = 1, \dots, m) \quad (1) \\
 & x_i \geq 0.
 \end{array}$$

La Programmation Linéaire Floue (PLF) peut assouplir les conditions de la PL classique principalement de deux façons [CFH 96] :

- P.L.F dans un environnement flou : C'est le cas où les contraintes et/ou la fonction objectif sont floues,
- P.L.F avec des coefficients flous : C'est le cas où les coefficients du problème (1) ne sont connus que de manière imprécises, représentés donc par des ensembles flous.

Différentes méthodes de résolution ont été proposées dans la littérature [CFH 96], [DAV 95]. On va donner ci-dessous quelques unes.

3.1.1. PLF dans un environnement flou :

A. Méthodes symétriques :

Cette classe de méthodes constitue un cas particulier du problème général de la décision dans un environnement flou, traité par Zadeh et Bellmann dans [BZA 70].

Le principe de Zadeh et Bellmann stipule que la décision dans un environnement flou peut être vue comme l'intersection des contraintes floues et de la fonction objectif floue, représentées par des sous-ensembles flous. La relation entre les contraintes et la fonction objectif est complètement symétrique, c'est à dire, il n'y a pas de différence entre contraintes et objectifs (flous).

- Méthode symétrique de Zimmermann :

Soit à résoudre le problème suivant [ZIM 75] :

$$\begin{aligned} \overline{Max.} \quad & C^T x. \\ \text{s. c.} \quad & Ax \lesssim b. \\ & x_i \geq 0 \end{aligned}$$

- C, x étant des vecteurs de R^n , A une matrice de dimension $(m \times n)$, b un vecteur de R^m ,
- $\overline{Max.}$ dénote une maximisation souple (floue),
- \lesssim : une relation d'ordre floue.

La maximisation rigide de la fonction objectif est remplacée par la satisfaction d'un niveau d'aspiration b_0 , ce qui est suffisant dans plusieurs situations pratiques.

Cela s'exprime par : $Max. C^T x \lesssim b_0$.

En appliquant le principe de Zadeh et Bellmann, ce problème s'écrit :

$$\begin{aligned}
 &\text{Trouver } x \\
 &\text{s. c. } \text{Max } C^T x \gtrsim b_0. \\
 &Ax \gtrsim b. \qquad (2) \\
 &x_i \geq 0
 \end{aligned}$$

Par ailleurs, on peut tolérer la violation de la fonction objectif et des contraintes avec des valeurs de tolérance p_i , pour $i = 0, \dots, m$. Dans ce cas, chacune des $(m+1)$ ligne du système (2) peut être représentée par un ensemble flou (figure 6).

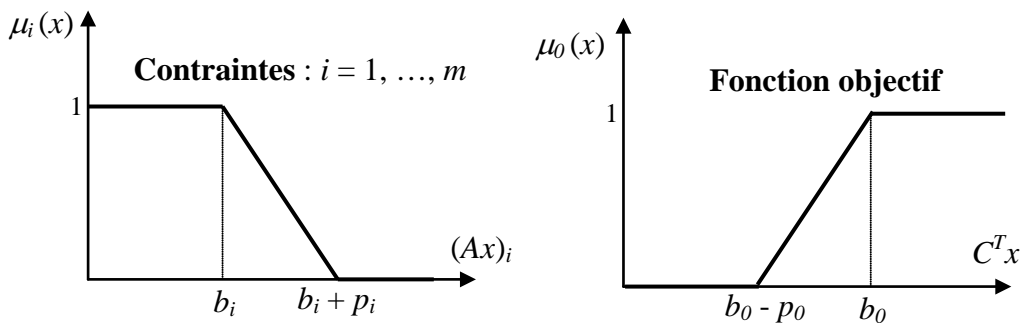


Fig. 6 - Contraintes et objectif

L'ensemble flou représentant la décision floue peut être obtenu comme l'intersection de ces ensembles flous. La meilleure décision (vecteur x) est celle du plus fort degré d'appartenance.

Par ailleurs, on peut transformer ce problème en un problème de PL classique :

$$\begin{aligned}
 &\text{Maximiser. } \alpha \\
 &\text{s. c. } C^T x \geq b_0 - (1-\alpha) \cdot p_0. \\
 &(Ax)_i \leq b_i + (1-\alpha) \cdot p_i \\
 &x_i \geq 0, \alpha \in [0,1].
 \end{aligned}$$

En d'autres termes : essayer d'être proche autant que possible des valeurs typiques.

B. Méthodes non symétriques : méthode de Verdegay :

Soit à résoudre le problème :

$$\begin{aligned}
 &\text{Trouver } z_0 = \text{Max. } C^T x. \\
 &\text{s. c. } Ax \gtrsim b. \qquad (3) \\
 &x_i \geq 0.
 \end{aligned}$$

Les fonctions d'appartenance des contraintes sont définies comme dans la figure 6.

Au lieu d'utiliser une approche symétrique qui résulte en un problème de PL crisp, cette méthode, proposée dans [VER 82], calcule l'ensemble flou représentant la solution optimale en cherchant la valeur optimale de la fonction objectif pour toutes les α -coupes de l'espace de solutions.

Ainsi, la méthode de Verdegay consiste à associer au problème initial (3), la famille suivante de problèmes de PL classique comme suit :

$$\begin{aligned} & \text{Max. } C^T x. \\ (\text{PL})_{0 \leq \alpha \leq 1} \quad & \text{s. c. } x \in X_\alpha, \\ & x_i \geq 0. \end{aligned}$$

X_α représente la α -coupe de l'espace flou des solutions possibles, i. e., satisfaisant les contraintes floues :

$$X_\alpha = \{ x \in R^n / \mu_i(x) \geq \alpha, \forall i = 1, \dots, m, x_i \geq 0 \}$$

Ce problème peut aussi s'écrire sous la forme:

$$\begin{aligned} & \text{Max. } C^T x. \\ (\text{PL})_\alpha \quad & \text{s. c. } (Ax)_i \leq b + (1 - \alpha) \cdot p_i, \quad i = 1, \dots, m. \\ & x_i \geq 0. \end{aligned}$$

L'ensemble flou représentant la décision sera alors:

$$S^* = \int_0^1 \frac{\alpha}{x_\alpha} , \quad x_\alpha \text{ étant la solution du problème } (\text{PL})_\alpha .$$

Cette solution du problème (3), sous la forme d'un sous-ensemble flou au lieu d'une seule valeur, permet un choix souple de la solution convenable.

C. Méthode basée sur le raisonnement approximatif :

Une méthode proposée par Fuller and Zimmermann [FZI 93] consiste à interpréter le problème de PLF comme un schéma d'inférence flou où les contraintes (relations floues dépendantes de x) sont les antécédents, la fonction objectif étant le fait (ensembles flous dépendants de x). La détermination de la solution se fait alors en deux étapes :

1. Pour tout vecteur de l'espace des solutions, calculer la valeur (floue) de la fonction objectif, qui n'est autre que la conclusion de l'inférence obtenue par la règle compositionnelle,

2. Dédire la solution du problème qui rend la fonction objectif maximale, ce qui résulte en un problème crisp de programmation mathématique non linéaire.

Cette approche peut être généralisée aux problèmes non linéaires, multi-objectifs à coefficients flous [FZI 93].

3.1.2. PLF avec des coefficients flous :

Une autre classe de problèmes de PL classique est le cas où les coefficients (A et C) du problème (1) sont mal connus, donc représentés par des ensembles flous.

Soit à résoudre le problème :

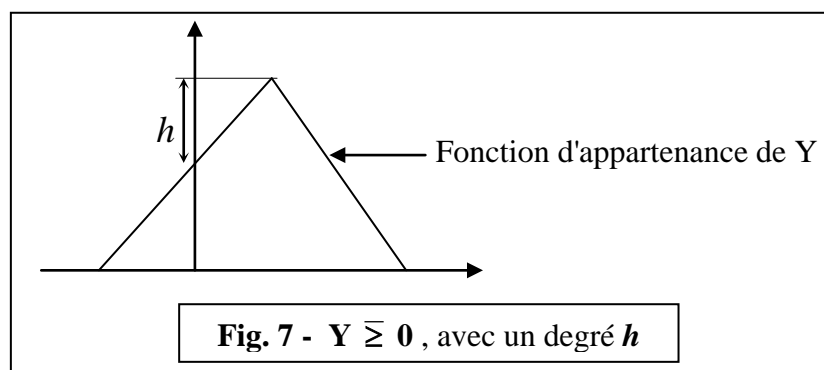
$$\begin{aligned} \text{Max.} \quad & \bar{C}^T x \\ \text{s. c.} \quad & \bar{A}x \leq \bar{b} . \\ & x_i \geq 0. \end{aligned}$$

Les coefficients \bar{C} , \bar{A} , \bar{b} sont représentés par des sous-ensembles flous.

De par leur signification intuitive et leur efficacité dans les calculs, beaucoup d'auteurs considèrent des nombres flous triangulaires ou du type LR pour la représentation des coefficients.

Un exemple des méthodes dédiées à cette classe est celle de Tanaka et Asai [ATN 84]. Après avoir écrit les inégalités floues du problème initial en la forme ($\bar{\geq}$), ils suggèrent de donner le sens de la figure 7 à cette dernière relation floue. A ce niveau, le problème se transforme facilement en un problème d'optimisation non linéaire (par rapport à h), auquel les mêmes auteurs ont associé un algorithme de résolution.

D'autres méthodes ont été proposées dans la littérature, chacune présentant une situation particulière [CFH 96].



Remarques :

1. Dans toutes les méthodes décrites jusqu'ici, on a utilisé des fonctions d'appartenance linéaires ou affines par morceaux pour exprimer le caractère flou, et le minimum

pour modéliser l'opérateur logique " ET ". Cela dit, on peut étendre cette situation par l'utilisation de fonctions d'appartenance non linéaires, et d'autres opérateurs d'agrégation que celui du min.

2. Ces méthodes de résolutions des problèmes de programmation linéaire floue constituent les modèles de base traités dans la littérature. D'autres approches plus récentes peuvent être trouvées dans [DAV 95].

3.2. Programmation mathématique floue :

Les autres domaines de la programmation mathématique floue comme la programmation non linéaire, entière, dynamique, ... n'ont pas été aussi explorés que celui de la programmation linéaire. Quelques résultats peuvent être trouvés dans [DAV 95]. On présente dans ce qui suit une approche de la programmation mathématique floue [CFU 98], basée sur la modélisation des contraintes floues par des règles floues " Si - Alors ".

- Optimisation sous règles floues Si - Alors :

On se propose de résoudre le problème suivant :

Max. $G(x)$

s. c. G vérifie la base de règles floues : $\{ R_1, R_2, \dots, R_n \}$

$x \in X$.

où : R_i sont des règles floues de la forme :

" Si x est A_i alors G est B_i ",

G étant la fonction objectif.

Soit $y \in X$.

On définit la valeur de la fonction objectif à partir de la base de règles floues en utilisant un mécanisme d'inférence comme celui de Tsukamoto, ou de Takagi-Sugeno.

Le problème se transforme alors en un problème d'optimisation crisp :

Max. $G(y)$

$y \in X$, où G est généralement continue.

Ce problème peut être résolu par les techniques classiques d'optimisation.

Les applications de la programmation mathématique floue ont touché plusieurs domaines et différents types de problèmes, notamment la décision multicritères [CFU 96].

Chapitre IV

Les systèmes flous

1. Introduction :

Le développement de modèles d'un système réel, pris au sens d'un ensemble d'éléments interconnectés, se trouvant en interaction, et représentant une entité quelconque qui assure la réalisation d'un certain but déterminé, est un sujet important dans plusieurs disciplines de la science et de l'ingénierie. Les modèles peuvent être utilisés pour l'**analyse** des caractéristiques des systèmes non directement accessibles à la mesure, la **prédiction** du comportement du système à une situation nouvelle, et pour la détermination des lois de **commande** [ZEL 95].

Dans le système réel, les variables X et Y , entrée et sortie du système respectivement, sont toujours liées par une relation de la forme :

$$F(X, Y) = 0 \quad (1)$$

Cette dépendance n'est pas toujours connue. La détermination des composantes du vecteur de sortie Y à partir de celles de X par des expériences n'est pas pratique, les expériences étant souvent longues et coûteuses, ou même impossibles. On fait alors appel à la modélisation.

La modélisation mathématique étudie les systèmes réels au moyen de la construction et de l'analyse de systèmes formels équivalents, c'est-à-dire, ayant des comportements semblables. La dépendance (1) est remplacée par un modèle mathématique :

$$E(X, Y) = 0 .$$

Souvent, cette dernière description dépend d'un vecteur de paramètres A . La recherche de A à partir d'exemples d'entrées et de sorties du système réel s'appelle : L'identification paramétrique.

Plus généralement, l'**identification** est définie par [ZEL 95] : " C'est la détermination sur la base de la connaissance des entrées et des sorties du système, d'un modèle mathématique appartenant à une classe de modèles donnée. Ce dernier, soumis aux mêmes signaux tests, étant alors équivalent au système donné".

Les problèmes de modélisation sont traditionnellement résolus dans le contexte de la modélisation mathématique en utilisant des équations algébriques, différentielles ou aux différences. La méthode la plus intuitive de modéliser un processus est d'utiliser les premiers principes de la physique pour établir les équations liant entrées et sorties. De tels modèles **analytiques** (ou **Boîtes blanches**) requièrent une très bonne compréhension du processus, ce qui n'est pas toujours le cas.

Une autre méthode d'accomplir l'identification du modèle est d'utiliser les seules données numériques réelles du système en question, de la forme entrées-sorties, pour produire des modèles **boîte noire**. La structure des modèles de ce type n'est souvent pas en relation avec la structure du système réel et les paramètres du modèle n'ont aucune interprétation physique. C'est l'exemple des différents modèles de régression et les réseaux de neurones.

Parfois, une approche combinant les deux techniques est utilisée. Les lois physiques sont utilisées pour écrire les équations générales régissant le système et certains paramètres ou fonctions sont identifiés à partir des données numériques. Cette approche de modélisation est dite semi-mécanique (**boîte grise**).

Les méthodes classiques de modélisation présentent plusieurs limitations. Les algorithmes d'identification ont été développés suffisamment pour les systèmes linéaires alors que la plupart des processus réels sont non linéaires et ne peuvent être approchés par des modèles linéaires que localement. Par ailleurs, les connaissances imprécises ou de nature qualitative ne peuvent être utilisées. Ceci a stimulé l'intérêt en la modélisation floue qui, par une approche semi-mécanique, peut pallier ces inconvénients, en permettant l'utilisation des connaissances qualitatives des experts et des données numériques dans le même cadre.

Un **système flou** est un système dont certaines de ses composantes sont floues. Il y a plusieurs niveaux où le flou peut intervenir [BAB 98]. Dans ce chapitre, on discutera les systèmes flous dont la relation liant entrées et sorties est décrite d'une manière symbolique qualitative, par des relations floues. Les systèmes de ce type sont appelés : systèmes flous à base de connaissances car les connaissances qualitatives remplacent les systèmes d'équations ou la fonction analytique utilisés dans la modélisation classique.

2. La modélisation floue à base de connaissances :

La modélisation floue à base de connaissances, permet l'utilisation des connaissances symboliques sur le système, déduites des expériences ou celles des experts.

On peut distinguer deux types de connaissances floues, décrivant la relation entre variable d'entrée X et variable de sortie Y [JAN 95a] :

- des relations floues du couple (X, Y) ,
- des règles floues " Si - Alors ".

Pour le premier type d'informations, on utilise la règle compositionnelle pour déduire une conclusion sur Y sous forme d'un ensemble flou à partir d'une connaissance sur X représentée par un sous-ensemble flou.

En ce qui concerne le deuxième type, le modus ponens généralisé est utilisé pour effectuer l'inférence. Les règles floues " Si – Alors ", sont modélisées par des implications floues. Ce sont donc, un cas particulier des relations floues.

En dépit du fait que le premier type peut paraître plus général, c'est le deuxième type qui est le plus communément utilisé. Cela est du en partie au fait qu'il est plus adapté à la modélisation des systèmes où l'entrée et la sortie sont liées par une relation causale.

La relation fonctionnelle $y = f(x)$, se trouve décrite par des règles floues de la forme :

" Si X est A alors Y est B " .

Par ailleurs, les règles floues " Si – Alors ", ou bien les implications floues associées, peuvent exprimer les relations floues les plus diverses, par définition même des implications floues, qui est très générale (déf. 3.1, chap. 2). C'est le cas, par exemple, pour l'implication de Mamdani ou de Larsen, qui modélisent des conjonctions. Pour ces raisons, on ne considérera par la suite que des bases de connaissances constituées uniquement de règles floues " Si – Alors " .

2.1 Structure interne d'un système d'inférence flou :

L'inférence floue consiste à déduire des conséquences (floues) à partir de prémisses imprécises et/ou incertaines. Cela se fait en trois étapes [DIM 93] : la fuzzification, l'Inférence et la défuzzification

2.1.1 La fuzzification :

La fuzzification est la phase de transformation de l'entrée numérique du système en une entrée floue, pour être compatible avec la forme de la description du système. Deux méthodes de fuzzification sont principalement utilisées [CAS 95] :

- La Fuzzification ponctuelle :

$$\{Z_0\} \leftrightarrow f_{z_0}(z) = \begin{cases} 1 & \text{si } z = z_0 \\ 0 & \text{sin on .} \end{cases}$$

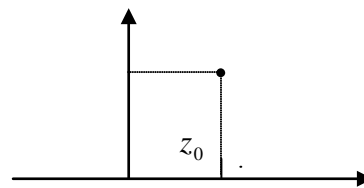


Fig. 1 - La Fuzzification ponctuelle

- La fuzzification approximative :

$$\{Z_0\} \leftrightarrow f_{z_0}(z) = \begin{cases} 1 & \text{si } z = z_0 \\ 0 & \text{si } |z - z_0| > \varepsilon \end{cases},$$

où $\varepsilon > 0$.

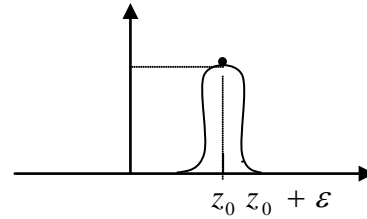


Fig. 2 - La Fuzzification approximative

La première méthode est la plus utilisée. Elle sera adoptée dans la suite.

2.1.2 L'inférence :

L'inférence peut être vue comme méthode d'interpolation [ZAD 94]. Elle répond à la problématique suivante :

" Etant donné une relation floue représentant un lien causal entre les valeurs d'un ensemble d'entrée et les valeurs d'un ensemble de sortie et une connaissance sur les valeurs admises en entrée, quelles sont les valeurs que peut prendre la sortie ? ".

On effectue l'inférence dans les systèmes flous par simple application de la loi compositionnelle, et spécialement, le modus ponens généralisé pour les implications floues.

Dans beaucoup d'applications pratiques de la logique floue, comme la commande floue ou les systèmes experts, plusieurs règles floues sont utilisées conjointement, constituant un seul bloc de la forme :

R_1	Si	x est	A_1	Alors	z est	C_1
R_2	Si	x est	A_2	Alors	z est	C_2
\vdots			\vdots			\vdots
R_n	Si	x est	A_n	Alors	z est	C_n
Fait		x est	A			
						Conséquence
						z est C

Le problème de la détermination de la fonction d'appartenance de la conclusion peut être abordé par deux approches [DIM 93] :

- A. Tout d'abord, combiner les n règles floues en une seule règle R , en utilisant une loi de composition "O" comme la conjonction (t-norme) ou la disjonction (t-conorme). La conclusion C est alors donnée par : $C = A \circ R$.
- B. Eclater les règles d'abord, individuellement, pour obtenir des résultats intermédiaires $C'_i = A \circ R_i$, le résultat final dans sera leur agrégation : $C = Agg(C'_1, C'_2, \dots, C'_n)$, où Agg est un opérateur d'agrégation qui effectue un compromis entre les différentes

contraintes sur les valeurs des variables, représentées par les différents sous-ensembles flous C'_i . Voir [FUL 01] pour plus de détails sur les opérateurs d'agrégation.

Remarque : Les deux méthodes donnent le même résultat lorsque l'entrée est crisp.

2.1.3 La défuzzification :

La sortie du processus d'inférence est un sous-ensemble flou. Souvent, on requiert un résultat crisp comme c'est le cas par exemple en commande de processus. Ainsi, on doit défuzzifier la sortie floue.

Définition 2.1.3 : La défuzzification est un processus de sélection d'un élément représentatif Z_0 de la sortie floue C .

Les opérateurs de défuzzification les plus utilisés sont [DIM 93] :

A. Centre de gravité :

La sélection se fait par :

$$Z_0 = \frac{\int_W Z.C(Z).dz}{\int_W C(z).dz},$$

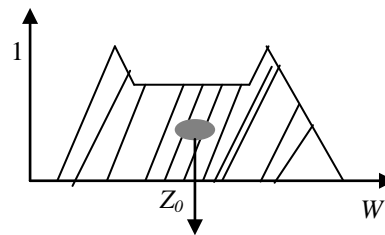


Fig. 3 - Le Centre de gravité -

W étant l'ensemble des valeurs que peut prendre Z .

Dans le cas discret, cela s'exprime par :

$$Z_0 = \frac{\sum_{i=1}^n Z_i.C(Z_i)}{\sum_{i=1}^n C(Z_i)}.$$

Cette méthode, choisissant le centre du domaine, tient compte des degrés d'appartenance des points d'une façon analogue à la recherche du centre de gravité. Elle est largement utilisée, en dépit de sa complexité de calcul élevée.

Une variante de la méthode du centre de gravité est celle du **centre de sommation**, utilisée dans le cas de plusieurs règles floues. Elle applique la défuzzification avant l'agrégation des résultats intermédiaires, effectuant à la fois l'agrégation et la défuzzification de la manière suivante :

- De chaque conclusion intermédiaire C_i , relative à la règle R_i , on choisit un élément z_i en utilisant la méthode du centre de gravité.
- Le résultat z est obtenu comme moyenne des éléments z_i :

$$z = \frac{\sum_1^n \int_W C_i'(z).z.dz}{\sum_1^n \int_W C_i'(z).dz}$$

Cette méthode présente l'avantage d'être plus rapide.

B. Défuzzification en hauteur :

Cette méthode consiste à choisir un seuil α et appliquer la méthode du centre de gravité aux éléments au-delà de ce seuil. En d'autres termes, ignorer les éléments dont le degré d'appartenance est inférieur à α , ce qui donne la formule suivante :

$$Z_0 = \frac{\int_{[C]^\alpha} Z.C(Z).dz}{\int_{[C]^\alpha} C(Z).dz}$$

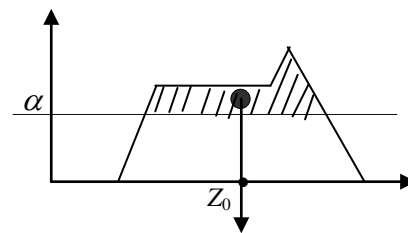


Fig. 4 - La Défuzzification en hauteur -

C. Centre du plus large domaine :

Cette méthode est utilisée dans le cas où C n'est pas convexe, mais décomposable en plusieurs ensembles flous convexes. Elle consiste à choisir l'ensemble flou de plus grande surface, duquel choisir un élément par la méthode du centre de gravité.

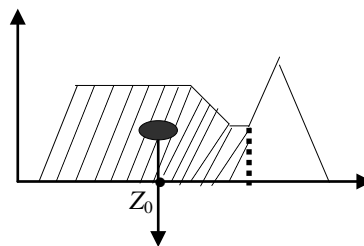


Fig. 5 - Centre du plus large domaine

- Éléments de comparaison :

En présence d'une application particulière, on doit avoir des éléments de comparaison entre les différentes méthodes de défuzzification pour faire le meilleur choix possible.

1. **Continuité** : des petits changements des entrées du système ne doivent pas produire de grandes différences quant au résultat de la défuzzification.
2. **Plausibilité** : Le point choisi z_0 doit avoir un fort degré d'appartenance à C . Cette propriété n'est manifestement pas satisfaite par la méthode du centre de gravité, par exemple.

3. **Complexité de calcul** : Ce critère est particulièrement important. La méthode de la défuzzification en hauteur pour α grand et celle du centre de sommation sont rapides ; tandis que celle du centre de gravité est lente,....

Par ailleurs, le choix de la méthode de défuzzification ne doit pas être fait indépendamment de l'implication floue utilisée (voir [BOU 95] pour plus de détails).

2.2 Différents types des systèmes flous :

Différents choix sont possibles quant aux opérations de conjonction dans les prémisses, implication floue, méthode de défuzzification et d'agrégation. Cela dit, selon la forme des conclusions des règles floues, deux types principaux de modèles flous peuvent être distingués [BAB 98] :

- Les modèles flous linguistiques,
- Les modèles flous de Takagi-Sugeno.

Sans nuire à la généralité, on se restreint aux systèmes à deux entrées et une sortie.

2.2.1 Les modèles flous linguistiques :

Ce type est caractérisé par des règles floues ayant des prémisses et des conclusions floues. Ils permettent une description linguistique du système par une base de règles floues de la forme :

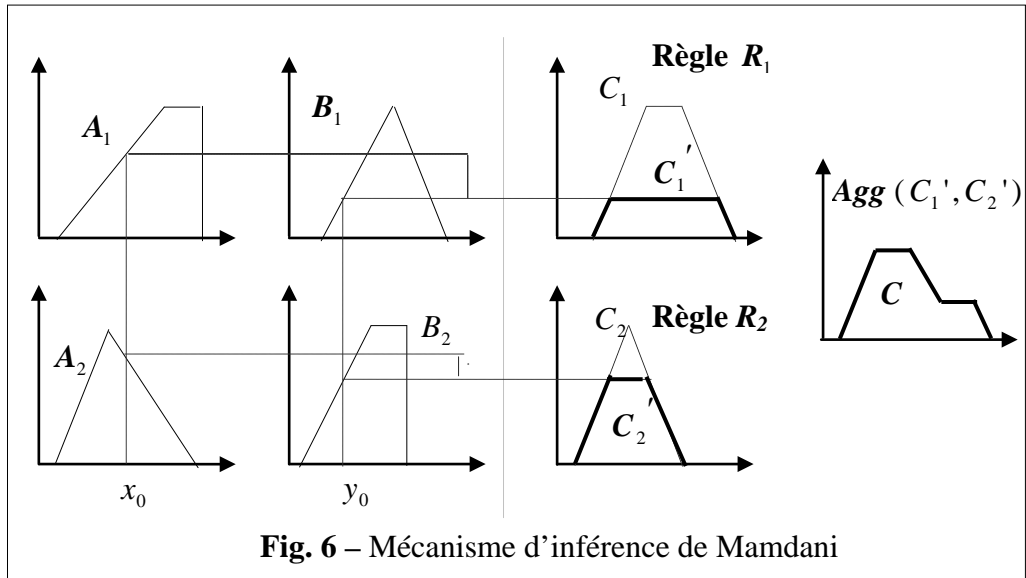
$$R_i : \text{ Si } X \text{ est } A_i \text{ et } Y \text{ est } B_i \text{ Alors } Z \text{ est } C_i, \quad i = 0, \dots, n$$

Dés les années 70, Mamdani et ses collaborateurs ont adopté l'approche linguistique pour transformer les règles heuristiques données par l'opérateur humain en une stratégie de contrôle [MAM 75]. Dans ce sens, un contrôleur flou constitue un modèle de l'opérateur humain et non pas du système lui-même.

Les modèles les plus connus de ce type sont celui de Mamdani et celui de Larsen.

A. Algorithme d'inférence *Max – Min* (Mamdani) :

- Calculer le niveau de chaque règle R_i : $\alpha_i = A_i(x_0) \wedge B_i(y_0)$;
- Calculer les sorties des règles individuellement : $C_i'(z) = \alpha_i \wedge C_i(z)$, $i = 1, \dots, n$.
- L'agrégation des résultats intermédiaires : $C(z) = C_1'(z) \vee \dots \vee C_n'(z)$.
- La Défuzzification de C pour obtenir un résultat crisp par la méthode du centre de gravité.



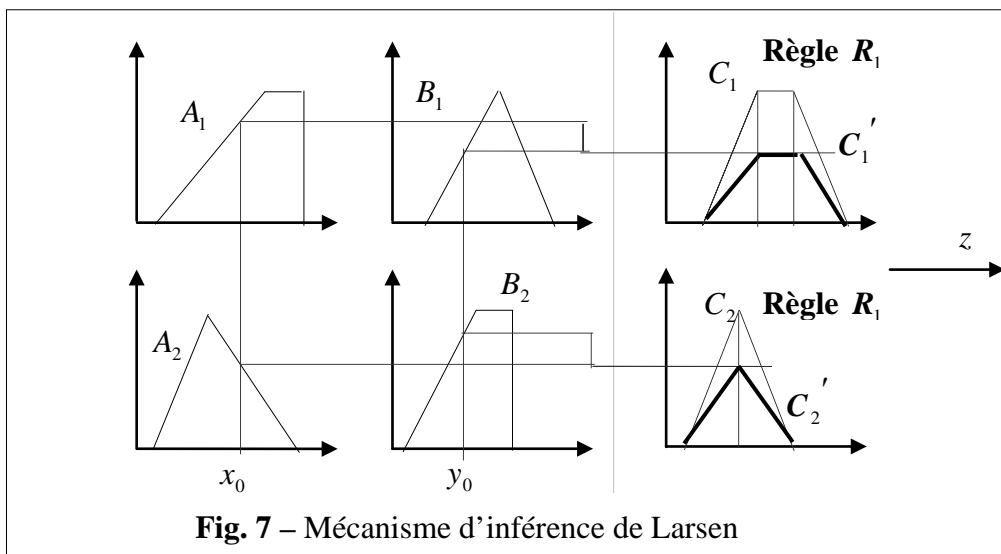
Le résultat final du mécanisme d'inférence de Mamdani est donné par la formule :

$$z_0 = f(x_0, y_0) = \frac{\int \text{Max}_z . \text{Min}[A_i(x_0), B_i(y_0), C_i(z)] . z . dz}{\int \text{Max}_z . \text{Min}[A_i(x_0), B_i(y_0), C_i(z)] . dz}$$

d'où le nom de *Max – Min* (le max. pour l'agrégation et le min. pour l'implication).

B. Algorithme D'inférence Max – Produit (Larsen) :

- Calculer le niveau de chaque règle R_i : $\alpha_i = A_i(x_0) \wedge B_i(y_0)$;
- Calculer les sorties des règles individuellement : $C'_i(z) = \alpha_i . C_i(z)$, $i = 1, \dots, n$.
- Obtenir un résultat crisp par la méthode du centre de sommation.



Le résultat final du mécanisme d'inférence de Larsen est donné par la formule :

$$z_0 = f(x_0, y_0) = \frac{\int \text{Max}_i . \alpha_i . C_i(z) . z . dz}{\int \text{Max}_i . \alpha_i . C_i(z) . dz}$$

avec : $\alpha_i = \text{déffuzification}(C_i)$.

Différentes variantes de la modélisation linguistique ont été proposées [ALC 99a]. On cite, par exemple, le modèle de Mamdani Approximatif, où les A_i et B_i sont des ensembles flous quelconques, sans avoir nécessairement, une interprétation linguistique [ALC 99b]. Cela permet d'avoir des systèmes plus précis, mais moins interprétables.

C. Propriétés des modèles linguistiques :

Dans les modèles linguistiques, les quantités floues sont décrites par des termes linguistiques, et le modèle flou constitue une description du système dans une langue naturelle.

Ces modèles ont les avantages suivants [NAK 97] :

- Les prémisses et les conclusions des règles sont de nature entièrement linguistique. Ceci les rend facilement compréhensibles, et permet de donner une interprétation sémantique claire au modèle.
- Il est tout à fait possible de combiner des règles de ce type identifiées à partir de données numériques par une méthode d'apprentissage, avec des règles formulées par un expert. C'est un avantage indéniable quand les données et/ou l'expertise sont incomplètes.
- Ces modèles sont bien adaptés à l'utilisation des entrées floues. Dans ce cas, la partie fuzzification est superflue.

En revanche, les modèles linguistiques ont les limitations suivantes [NAK 97] :

- Leur capacité de représentation est limitée. En d'autres termes, un grand nombre de règles est souvent nécessaire pour approcher le comportement global d'un système. Par conséquent, le nombre de paramètres à identifier est élevé.
- Le grand nombre de paramètres exige une base d'informations importante et complète pour déterminer tous les paramètres, ce qui veut dire un nombre de données d'identification important, mais aussi des données réparties sur la totalité de l'espace d'entrée.
- Dans le cas des systèmes dynamiques, l'obtention d'un tel ensemble de données constitue une limitation, car certains états ne seront pas atteints lors de l'acquisition

des données d'apprentissage. Ces états ne seront alors représentés par aucune règle du modèle résultant. Un problème se pose donc lorsque le modèle est utilisé, et que des situations absentes de la base de règles se présentent.

2.2.2 Modèles flous de Takagi-Sugeno-Kang (TSK) :

Les modèles linguistiques ne permettent pas d'intégrer directement des connaissances objectives sur le système qui ne sont pas exprimées par des sous-ensembles flous. Ainsi, des connaissances exprimées sous forme analytique décrivant la structure physique du système, par exemple les équations de conservation d'énergie ou du bilan de masse ne peuvent pas être prises en compte directement. Sugeno et ses collaborateurs dans [TAK 85], [SUG 86] ont proposé une autre classe de modèles qui permet de représenter ce type d'informations. Pour un système à deux entrées et une sortie, un modèle de TSK est constitué par une base de règles floues de la forme :

$$R_i : \text{ Si } X \text{ est } A_i \text{ et } Y \text{ est } B_i \text{ Alors } Z = F_i(x, y), \quad i = 0, \dots, n.$$

où F_i sont des fonctions crisp.

Les fonctions F_i sont généralement linéaires, affines ou même constantes. Chacune des règles représente un modèle local pour une région floue d'entrée. Dans chaque région, le modèle flou est défini par la fonction F_i qui relie les entrées à la sortie numérique. Le modèle global est obtenu par interpolation des modèles locaux. Cela nous permet d'accroître la précision du système, mais au détriment de sa lisibilité.

On va présenter les modèles flous de ce type, les plus utilisés en pratique.

A. Le modèle du singleton :

Ce type est caractérisé par des règles ayant des conclusions constantes b_i . Une méthode spéciale pour effectuer l'inférence et la défuzzification à la fois est utilisée pour ce type, appelée : *la moyenne floue* [BAB 98].

Algorithme d'inférence de la méthode du singleton :

- Calculer le niveau de chaque règle $R_i : \beta_i = A_i(x_0) \wedge B_i(y_0)$;
- Calculer la sortie du système (inférence / défuzzification) par :

$$z_0 = \frac{\sum_{i=1}^n \beta_i \cdot b_i}{\sum_{i=1}^n \beta_i} .$$

Remarquons que pour un système SISO, cette sortie s'exprime comme une moyenne pondérée des sorties des règles individuelles b_i . Cela résulte en une approximation affine par morceaux lorsqu'on utilise des fonctions d'appartenance triangulaires ou trapézoïdales [BAB 98], comme l'indique la figure 8.

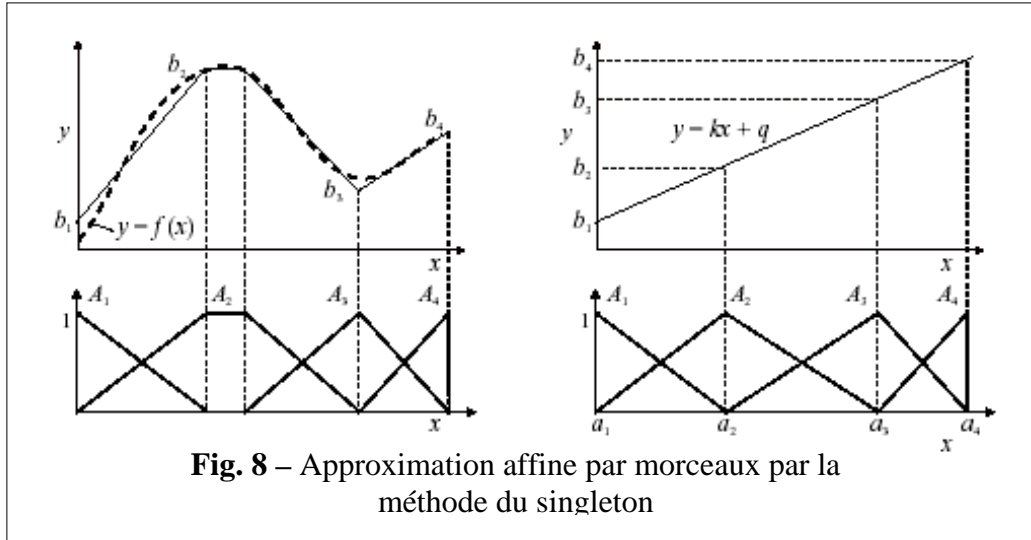


Fig. 8 – Approximation affine par morceaux par la méthode du singleton

B. Mécanisme d'inférence de Sugeno :

Dans cette méthode, les sorties des règles sont combinaisons linéaires (ou affines) des entrées. La base de règles floues se présente sous la forme :

$$R_i : \text{ Si } X \text{ est } A_i \text{ et } Y \text{ est } B_i \text{ Alors } Z = a_i \cdot x + b_i \cdot y, \quad i = 0, \dots, n.$$

La moyenne floue est utilisée pour calculer la sortie du système, étendant ainsi le modèle du singleton.

Algorithme d'inférence de Sugeno :

- Calculer le niveau de chaque règle R_i : $\alpha_i = A_i(x_0) \wedge B_i(y_0)$;
- Calculer les sorties des règles individuellement : $z_i = a_i \cdot x_0 + b_i \cdot y_0$;

- Calculer la sortie finale par :
$$z_0 = \frac{\sum_{i=1}^n \alpha_i \cdot z_i}{\sum_{i=1}^n \alpha_i}.$$

Le résultat final est donc donné par :

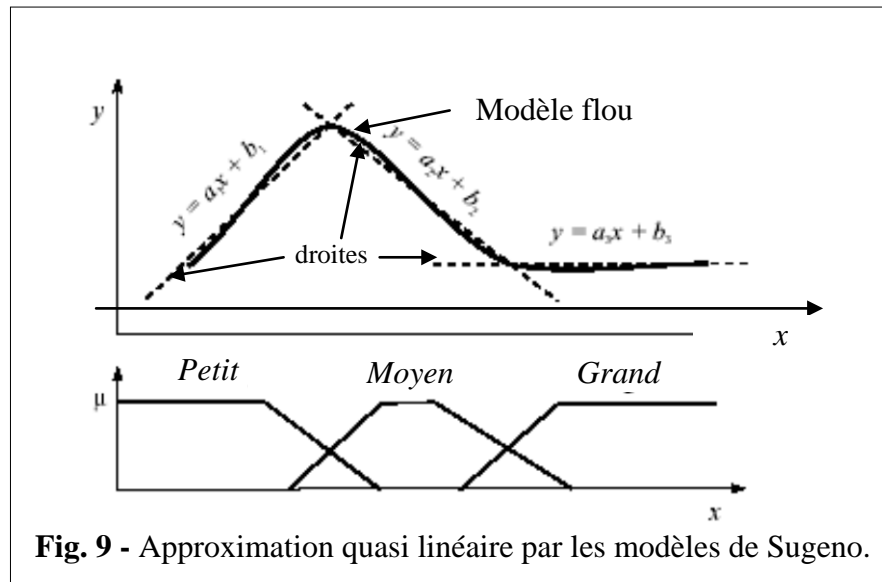
$$z_0 = \frac{\sum_{i=1}^n \text{Min}[A_i(x_0), B_i(y_0)] \cdot (a_i \cdot x_0 + b_i \cdot y_0)}{\sum_{i=1}^n \text{Min}[A_i(x_0), B_i(y_0)]}.$$

Exemple : La figure ci-dessous montre l'approximation quasi linéaire d'une fonction statique (SISO) par un modèle affine de Sugeno, en utilisant les trois règles suivantes :

$$\text{Si } x \text{ est petit Alors } y = a_1 \cdot x + b_1,$$

$$\text{Si } x \text{ est moyen Alors } y = a_2 \cdot x + b_2,$$

$$\text{Si } x \text{ est grand Alors } y = a_3 \cdot x + b_3.$$



C. Propriétés des modèles de TSK :

Les modèles de TSK ont les avantages suivants [NAK 97] :

- Ils permettent de décrire des situations où la structure physique du système est bien connue.
- Ils permettent d'approcher un système complexe par une collection de modèles locaux, généralement linéaires.
- Les modèles de TSK ont une capacité de représentation importante. Le nombre de règles nécessaires pour approcher un système à un certain degré de précision est généralement plus faible que dans le cas des modèles linguistiques. L'identification du modèle à partir de données numériques en est facilitée et nécessite moins de données que pour les modèles linguistiques.
- La possibilité d'utilisation des méthodes classiques pour l'identification des paramètres, comme par exemple les méthodes d'optimisation.

Toutefois, les modèles de TSK ont les inconvénients suivants [NAK 97] :

- De par la nature fonctionnelle des conclusions, les règles n'ont pas toujours une signification sémantique claire comme dans le cas des modèles linguistiques.

- L'interpolation entre les modèles locaux peut engendrer une erreur importante qui réduit la précision globale du modèle.
- Ces modèles ne permettent pas d'intégrer l'expérience humaine formulée par des règles linguistiques.
- La méthode de calcul ne permet pas la manipulation des entrées floues, car le calcul de la sortie exige les valeurs précises des variables d'entrées.

2.2.3 Comparaison :

Les modèles linguistiques ont une représentation moins efficace que les modèles TSK et demandent par conséquent plus de règles pour décrire le comportement d'un système. En revanche, les modèles linguistiques offrent une meilleure interprétation sémantique, et permettent d'incorporer des règles proposées par des experts humains de manière directe.

Ainsi, ces deux types de modèles flous paraissent complémentaires [NAK 97]. Le choix de l'un ou de l'autre dépend en effet des données du problème et du but de la modélisation.

Pour la modélisation des systèmes dynamiques, les modèles de TSK s'avèrent plus appropriés, et se caractérisent par une grande précision.

3. Modélisation et identification des systèmes dynamiques :

Avant de commencer notre discussion des systèmes dynamiques, rappelons, de prime abord, qu'un système pris au sens d'un transfert entre variables d'entrée et variables de sortie, est dit dynamique si la relation liant l'entrée et la sortie n'est pas statique. Autrement dit, la sortie du système ne dépend pas uniquement de l'entrée courante, mais aussi des entrées et sorties antérieures.

L'identification des systèmes dynamiques peut être transformée en un problème d'identification de fonctions statiques, même si d'autres problèmes apparaissent. Contrairement aux systèmes statiques, l'erreur à un moment donné se propage dans un système dynamique de par sa structure récursive. Donc, une plus grande précision est exigée dans l'identification de la fonction statique. Cela dit, on commence par la modélisation floue des fonctions statiques.

3.1 Approximation des fonctions statiques :

La construction des systèmes flous revient à la modélisation d'une fonction de transfert de l'espace d'entrée vers l'espace de sortie en tenant compte éventuellement d'autres

variables comme celles de l'état. Cette constatation nous amène naturellement à lier la construction des systèmes flous à la théorie de l'approximation des fonctions.

Après leurs exploits pratiques, l'emploi des systèmes flous a été justifié théoriquement par leur propriété d'approximation universelle [BOU 94]. Les systèmes flous des deux types, de Mamdani et de Sugeno, peuvent approcher uniformément toute fonction continue sur un compact, avec n'importe quel degré de précision.

Wang et Mendel [WAN 92] sont les premiers à montrer qu'une famille particulière de systèmes flous, celle utilisant des fonctions d'appartenance A_i et B_j gaussiennes, la multiplication comme T-norme d'agrégation des prémisses et une défuzzification par la méthode du centre de gravité, constitue un sous-espace dense dans l'espace des fonctions continues sur un compact. Ils utilisent pour cela le théorème de Stone-Weierstrass. Nguyen et ses collaborateurs montrent que le résultat demeure valide et retrouvent donc des propriétés équivalentes dans le cas de l'utilisation de l'opérateur minimum comme T-norme d'agrégation [NGU 93]. B. Bouchon et ses collaborateurs ont étendu ces résultats aux fonctions mesurables [BOU 94].

On peut trouver dans [BOU 94] une synthèse des différents résultats et approches relatifs à la propriété d'approximation universelle.

3.2 La modélisation des systèmes dynamiques :

Les systèmes flous permettent de modéliser la relation statique entre les entrées d'un système dynamique et ses sorties de façon qualitative, en tenant compte des entrées et sorties antérieures dans le cas discret ou encore des dérivées dans le cas continu.

De par le fait que les systèmes flous sont des approximateurs universels, de tels modèles peuvent être utilisés pour prédire et contrôler des systèmes dynamiques, avec n'importe quelle précision. Les modèles flous de TSK sont plus appropriés et largement utilisés pour la modélisation des systèmes dynamiques.

Deux formes de modélisation des systèmes dynamiques par logique floue sont décrites ci-dessous. La première forme, couramment utilisée, est l'utilisation des descriptions qualitatives des relations de **régression** entre la sortie courante du système et ses entrées et sorties antérieures (systèmes discrets). La deuxième forme concerne la formulation en **espace d'état**, valable pour les systèmes discrets et continues.

3.2.1 Formulation discrète entrée-sortie :

On prend le cas d'un système dynamique SISO. Une formulation en temps discret peut s'écrire sous la forme :

$$y(k+1) = f(y(k), \dots, y(k-n+1), u(k), \dots, u(k-m+1)) ;$$

où : f est une fonction non linéaire.

Un système flou peut être utilisé pour approcher la fonction f [BAB 98].

Ainsi, ce système dynamique SISO peut être modéliser par un système flou MISO dont la forme des règles, pour un choix du modèle linguistique, est :

$$\begin{aligned}
 R_i : \quad & \text{Si} \quad y(k) \text{ est } A_{i,1} \quad \text{Et} \quad \dots \quad \text{Et} \quad y(k-n+1) \text{ est } A_{i,n} , \\
 & \text{Et} \quad u(k) \text{ est } B_{i,1} \quad \text{Et} \quad \dots \quad \text{Et} \quad u(k-m+1) \text{ est } B_{i,m} , \\
 & \text{Alors} \quad y(k+1) \text{ est } C_i .
 \end{aligned}$$

Ce modèle flou peut être vu comme un modèle NARX (Nonlinear AutoRegression with eXogenous inputs).

Par ailleurs, on peut utiliser un modèle de Sugeno, en remplaçant dans le modèle précédent la conclusion C_i par une combinaison linéaire de y_i et de u_i par exemple. La conclusion s'écrit alors :

$$y(k+1) = \sum_{i=1}^n a_i \cdot y_i + \sum_{j=1}^m b_j \cdot u_j .$$

Construction des modèles flous entrée-sortie :

La construction de ce type de modèles consiste en la détermination des paramètres n et m , et la détermination des fonctions d'appartenance.

Le premier problème est commun avec tous les modèles de régression, et peut être résolu par l'utilisation des connaissances qualitatives, par essai et erreur, ou en utilisant des méthodes automatiques basées sur les données numériques expérimentales.

Pour le deuxième problème, les fonctions d'appartenance sont fournies, le cas échéant, par les experts, ou peuvent être générées de façon automatique en utilisant les données numériques (voir chapitre 5).

3.2.2 Formulation de l'espace d'état :

Il est possible de représenter un système dynamique par des variables d'état, qui permettent une analyse interne du système. La formulation d'état s'écrit sous la forme :

$$\begin{cases} S.X(k) = f(X(k), U(k)) \\ Y(k) = g(X(k), U(k)) \end{cases} , \text{ où}$$

- f et g sont des fonctions statiques.
- X représente l'état du système, U l'entrée et Y sa sortie.
- $S.X(k)$ représente $X(k+1)$ dans le cas discret et $X'(k)$ dans le cas continu.

Les systèmes flous des différents types peuvent être utilisés pour approcher les fonctions statiques f et g .

En pratique, les modèles de TSK les plus utilisés sont les modèles linéaires [VER 98], qui se présentent sous la forme :

$$R_i : \text{ Si } Z \text{ est } F_i \text{ Alors } \begin{cases} S.X(k) = A_i.X(k) + B_i.U(k) \\ Y(k) = C_i.X(k) \end{cases} \quad i = 0, \dots, n$$

où Z est vecteur de variables d'état et/ou d'entrée.

Habituellement, le produit est choisi pour modéliser la conjonction dans les prémisses et la moyenne floue pour effectuer la défuzzification. Les sorties finales du modèle s'écrivent alors sous la forme :

$$\begin{cases} S.X(k) = \frac{\sum_{i=1}^n w_i(Z(k)) \cdot \{A_i.X(k) + B_i.U(k)\}}{\sum_{i=1}^n w_i(Z(k))}, \\ Y(k) = \frac{\sum_{i=1}^n w_i(Z(k)) \cdot C_i.X(k)}{\sum_{i=1}^n w_i(Z(k))}. \end{cases}$$

Ou encore :

$$\begin{cases} S.X(k) = \sum_{i=1}^n w_i^*(Z(k)) \cdot \{A_i.X(k) + B_i.U(k)\}, \\ Y(k) = \sum_{i=1}^n w_i^*(Z(k)) \cdot C_i.X(k). \end{cases}$$

Construction des modèles de l'espace d'état :

L'obtention de modèles flous sous la forme précédente peut être réalisée par différentes approches [MOR 01] :

1. obtenir un modèle flou directement à partir du modèle de connaissances,
2. obtenir un modèle flou par linéarisation autour de plusieurs points de fonctionnement,
3. obtenir un modèle flou par identification à partir de données numériques expérimentales (voir chapitre 5).

Remarque : Il convient de signaler que lorsque le modèle de TSK est identifié à partir de données numériques, les "sous- modèles" peuvent être très différents des modèles obtenus par linéarisation [JOH 99].

4. Conclusion :

Ce chapitre contient les éléments de base de la modélisation floue. L'avantage principal de la modélisation floue par rapport aux autres types de modélisation est la possibilité d'incorporer des connaissances symboliques d'une manière naturelle et flexible. Cela permet d'aborder les problèmes complexes, qui peuvent être résolus par des acteurs humains. Ainsi, les règles symboliques " Si - Alors " remplacent les équations différentielles, et le raisonnement exact, rigide, laisse la place à un raisonnement approximatif flexible, qui peut utiliser des connaissances imparfaites, entachées d'imprécisions et d'incertitudes.

Les systèmes flous sont des approximateurs universels. Toute fonction continue peut être approchée avec n'importe quelle précision par des modèles flous. Cette propriété a donné les justifications théoriques au succès de la logique floue dans les réalisations pratiques.

Les systèmes flous peuvent modéliser des systèmes dynamiques de façon qualitative. Deux principales formes sont décrites dans ce chapitre : les modèles de régression et la formulation en espace d'état. Les modèles flous de TSK sont plus appropriés et largement utilisés pour la modélisation des systèmes dynamiques, de par leur grande précision et leur forme analytique. Ils peuvent être utilisés comme moyen d'interpolation entre plusieurs modèles locaux. Les modèles de Mamdani, largement utilisés en contrôle, sont appropriés à la modélisation des connaissances qualitatives des experts. Les deux types de modélisation s'avèrent donc complémentaires.

Chapitre V

Identification des systèmes flous

1. Introduction :

Dans le chapitre précédent, on a exposé les différents types de modèles flous. L'objet du présent chapitre est de répondre à la problématique suivante : " En présence d'une application donnée, comment peut-on construire un système flou ? ".

La construction des modèles flous repose naturellement sur les connaissances symboliques des experts, exprimées sous forme de règles linguistiques. Cependant, l'utilisation des mesures du système réel de la forme entrées-sorties peut être avantageuse, notamment dans le cas où les connaissances symboliques ne sont pas disponibles ou sont incomplètes.

Cela dit, deux approches de construction des systèmes flous sont possibles : identification des systèmes flous à partir d'une collection de données numériques, ou bien, l'utilisation de la connaissance des experts. Cette dernière approche dépend largement du problème ; plus on est exigeant quant à la précision plus on a besoin de règles heuristiques. Il est clair, toutefois, qu'avec des algorithmes d'identification, la modélisation peut être rendue automatique à un certain niveau seulement et au moins quelques paramètres sont donnés par les experts, comme le type du système, le nombre de variables, nombre de règles, etc. Cette procédure de modélisation est plus ou moins combinaison des deux approches.

Le but de ce chapitre est de présenter les algorithmes d'identification floue les plus répandus dans la littérature. Plusieurs classes sont décrites.

2. Méthodes ad hoc :

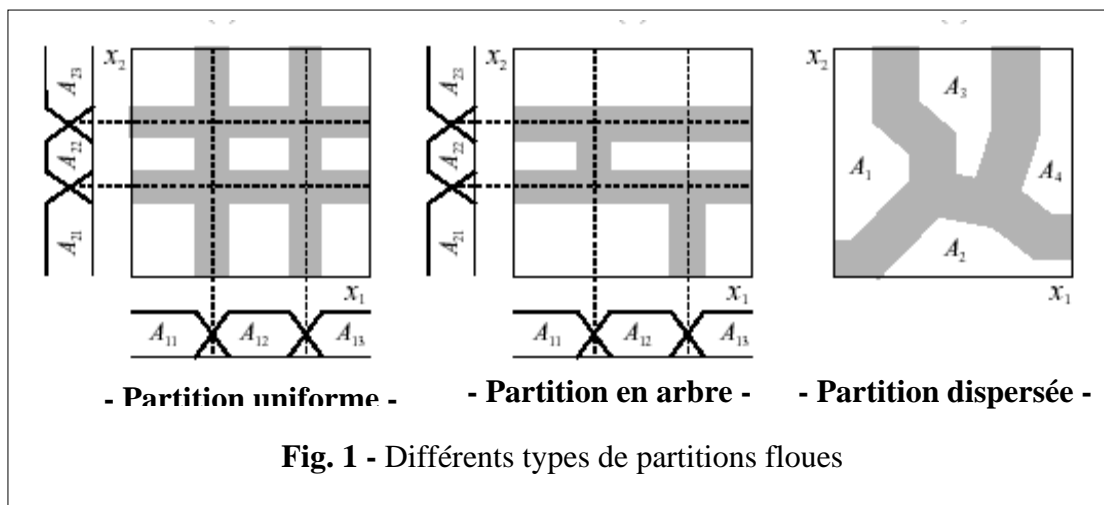
Ces méthodes sont basées sur l'apprentissage des règles floues en extrayant les critères à partir des exemples (entrées-sorties), en ce sens que chaque donnée numérique de la forme entrée-sortie fournit une règle floue. La base ainsi obtenue contient un nombre élevé de règles floues et sera par la suite raffinée. La méthode de Wang et Mendel est un exemple des méthodes ad hoc.

2.1. Méthode de Wang et Mendel :

Cette méthode proposée en 1992 par Wang et Mendel présente l'avantage d'être simple tout en ayant de bons résultats [ALC 99a]. Elle se compose de quatre étapes :

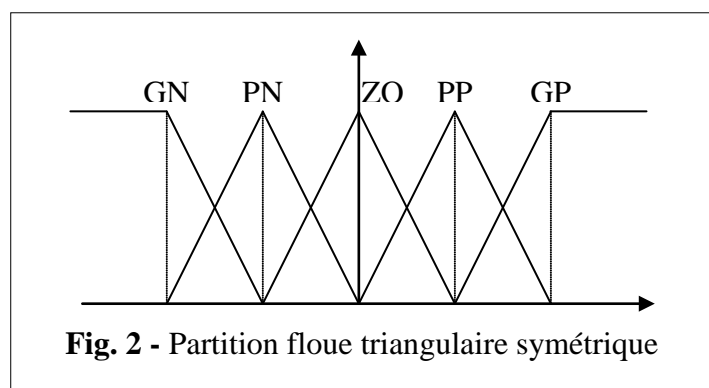
1. Effectuer une partition de l'espace des variables d'entrée.

La partition peut être réalisée directement par un expert. On utilise généralement trois types de partitions, qui sont schématisées sur la figure 1 pour un système à deux entrées [JAN 95b].



Dans le cas où l'on ne dispose pas de connaissances à priori, on peut effectuer une partition uniforme pour chaque univers de discours en choisissant un type particulier de fonction d'appartenance.

Exemple : Partition symétrique uniforme par des ensembles flous triangulaires.



GN : Grand Négatif

ZO : Zero

PP : Petit Positif

PN : Petit Négatif

GP : Grand Positif

2. Générer une base préliminaire de règles floues :

A partir de chaque exemple (entrée-sortie), générer une règle floue en prenant pour variables linguistiques (prémises et conclusions) les ensembles flous qui conviennent au mieux aux valeurs numériques (plus grand degré d'appartenance).

3. Attribuer à chaque règle un degré d'importance :

Assigner à chaque règle R_i de la forme :

Si X_1 est A_1^i et et X_n est A_n^i Alors Y est B^i ,

déduite à partir de $X^i = (x_1^i, \dots, x_n^i)$, le degré d'importance :

$$G(R_i) = A_1^i(x_1^i) \times A_2^i(x_2^i) \times \dots \times A_n^i(x_n^i) \times B^i(y^i).$$

4. Obtenir la base finale des règles floues :

Pour chaque combinaison des antécédents, choisir la règle la plus importante, en tenant compte des degrés d'importance.

2.2. Avantages et inconvénients :

Les principaux avantages des méthodes ad hoc sont [CAS 00] :

- Ces méthodes peuvent être facilement compréhensibles et implémentables,
- La phase d'apprentissage est rapide,

Cependant, les modèles obtenus par ce type de méthodes d'apprentissage ne sont pas parfois très précis [CAS 00]. Ces méthodes peuvent être utilisées pour obtenir un modèle flou préliminaire, qu'on peut raffiner par la suite, en utilisant d'autres techniques.

3. Les méthodes basées sur la coalescence¹ des données :

La coalescence des données est à la base de beaucoup d'algorithmes de classification et de modélisation [JAN 95a]. Le but est d'identifier des coalitions naturelles dans les données (entrées-sorties), pour pouvoir représenter le comportement du système d'une manière concise.

¹ La coalescence est la traduction succincte du mot anglais " Clustering ", et qui veut dire aussi le groupement.

Les deux types de modèles flous (linguistique et de TSK) peuvent être dérivés par coalescence, ceci en choisissant à chaque fois, convenablement le critère (prototype) sur lequel repose la coalescence.

Pour simplifier l'exposé, on se place dans le cas d'un système SISO.

3.1. Dérivation du modèle linguistique par coalescence :

Supposons qu'on ait pu effectuer une partition des couples entrées-sorties disponibles, en tenant compte d'un critère de similarité ou encore de proximité.

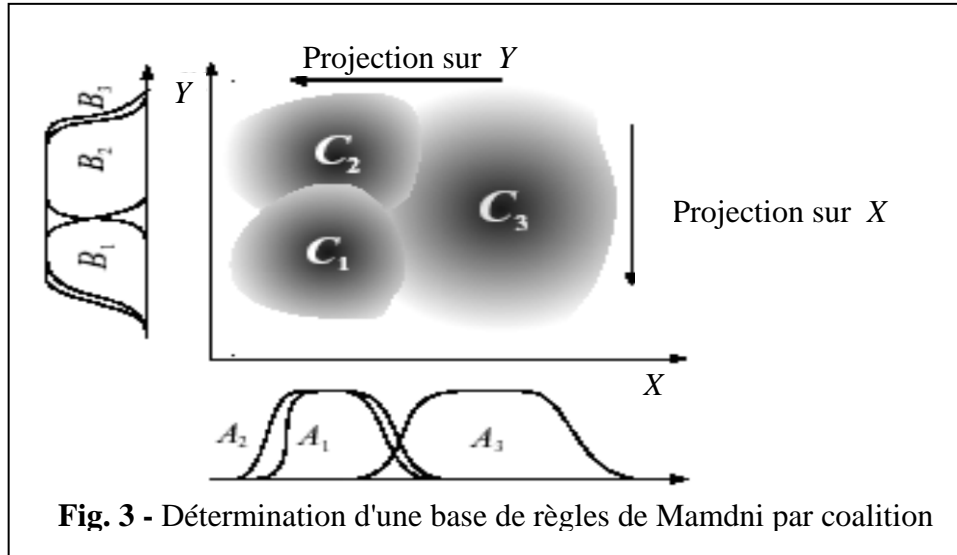
Les fonctions d'appartenance des antécédents et des conclusions des règles peuvent être déduites par projection des coalitions sur les axes.

Exemple : La figure 3 montre comment l'on peut extraire des règles floues à partir d'une coalescence floue basée sur le critère de proximité, donnant lieu à des coalitions sphériques, représentées par les ensembles flous C_1 , C_2 et C_3 . La projection des coalitions sur les deux axes nous permet de déduire les trois règles suivantes :

Si X est A_1 alors Y est B_1

Si X est A_2 alors Y est B_2

Si X est A_3 alors Y est B_3



3.2. Dérivation du modèle de TSK par coalescence :

Si au lieu d'utiliser la proximité comme critère de coalescence, on utilise un autre critère, celui de la détermination des sous-espaces vectoriels (droites), on dérive le modèle affine de TSK.

Exemple : La figure 4 montre comment on peut extraire les règles floues de TSK à partir d'une coalescence floue.

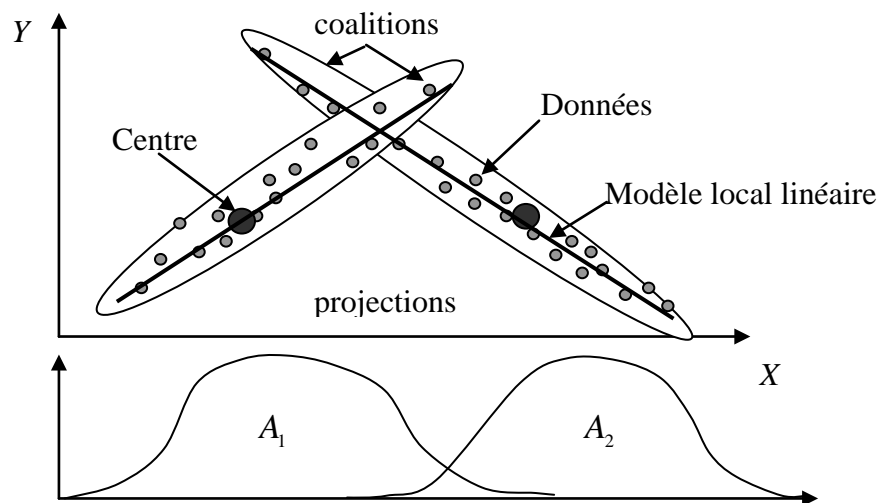


Fig. 4 - Dérivation des règles de TSK par coalition

La projection des coalitions nous permet de déduire les deux règles suivantes :

$$\text{Si } X \text{ est } A_1 \text{ alors } Y = a_1 \cdot X + b_1 ,$$

$$\text{Si } X \text{ est } A_2 \text{ alors } Y = a_2 \cdot X + b_2 .$$

Les coefficients a_i et b_i peuvent être estimés par la méthode des moindres carrés, par exemple.

3.3. Algorithmes de coalescence :

- Algorithme des C-moyens (HCM) :

L'algorithme des C-moyens, introduit par Jim Bezdek en 1981, permet de localiser les coalitions dans l'espace euclidien des données, en suivant les étapes suivantes [JAN 95a] :

1. Munir l'algorithme de C centres des coalitions qu'on cherche (information à priori sur le nombre de coalitions : algorithme supervisé),
2. Chaque point est assigné au centre le plus proche (au sens de la distance euclidienne),
3. Un nouveau centre de chaque coalition est déterminé en prenant le centre de gravité.

4. Si le résultat n'est pas satisfaisant (critère d'arrêt) aller à 2 ; sinon, l'algorithme est terminé.

On peut représenter les coalitions par une matrice $M = (m_{i,k})_{i,k}$:

$$m_{i,k} = \begin{cases} 1 & , \text{ Si le point } u_k \text{ appartient à la classe } C_i . \\ 0 & , \text{ Sinon.} \end{cases}$$

Remarque : La convergence de cet algorithme n'est pas garantie, et dépend du choix initial des centres.

- **L'algorithme flou des C-Moyens (FCM) :**

La fuzzification de l'algorithme des C-moyens, permet à des points d'appartenir à plusieurs coalitions avec des degrés d'appartenance. La matrice M devient à valeurs dans l'intervalle $[0,1]$ tout entier au lieu de 0 et 1 seulement. Le résultat est une partition en classes floues, permettant la déduction du comportement du système sous forme de règles floues [JAN 95a].

- **L'algorithme de « Subtractive clustering » :**

Le FCM est un algorithme supervisé, dans le sens où le nombre de coalitions (C) doit être fourni à l'avance. Dans le cas où une telle information n'est pas disponible a priori, on peut utiliser l'algorithme de « subtractive clustering ». C'est un algorithme non supervisé, basé sur les mesures des densités des points. L'idée est de localiser les régions de l'espace possédant des densités de points, élevées [JAN 95a].

4. Méthodes neuro-floues :

L'utilisation conjointe des méthodes neuronales et floues dans des types hybrides permet de tirer avantage des qualités de l'une et de l'autre. Principalement, des capacités d'apprentissage des premières et de la lisibilité et la souplesse des dernières [BOU 95].

Un bref exposé sur les réseaux de neurones est donné en annexe (annexe A).

Le principal type d'association entre réseaux de neurones et systèmes flous est le cas où un système d'inférence flou est mis sous la forme d'un réseau multicouche, dans lequel les poids correspondent aux paramètres du système, l'architecture du réseau dépendant du type de règles et des méthodes d'inférence, d'agrégation et de défuzzification choisies [JAN 93].

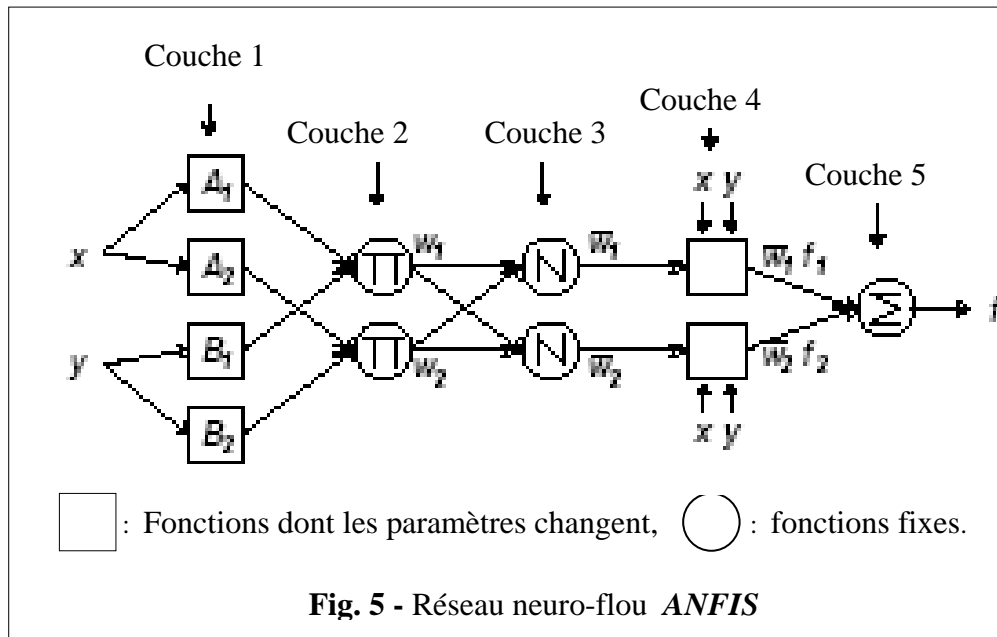
On va présenter une méthode neuro-floue servant à l'apprentissage des fonctions d'appartenance et des paramètres de la conclusion pour un modèle de type TSK.

- **Méthode de Jang (ANFIS) :**

Cette méthode proposée par Jang en 1993 implémente le modèle de TSK d'ordre zéro ou d'ordre un, sous la forme d'un réseau de neurones, dont les paramètres ajustables sont ceux des fonctions d'appartenance ainsi que ceux de la conclusion [JAN 93].

Cette méthode très performante est largement utilisée dans les réalisations pratiques [JAN 95a].

Dans le cas d'un système à deux entrées avec deux termes linguistiques chacune et une seule sortie, ANFIS prend la forme de la figure 6.



La propagation des valeurs de l'entrée vers la sortie se fait de la manière suivante :

- 1^{ère} couche : Calcule les degrés d'appartenance,
- 2^{ème} couche : réalise la connexion « ET » (produit des degrés d'appartenance),
- 3^{ème} couche : Normalisation, valeurs dans l'intervalle [0,1],
- 4^{ème} couche : Calcule les sorties des règles individuellement,
- 5^{ème} couche : l'agrégation, sortie finale du système.

Règle d'apprentissage : On peut appliquer par exemple la rétropropagation, une méthode basée sur le gradient (voir Annexe 1), pour l'apprentissage des paramètres des fonctions d'appartenance (couche 1), ainsi que les paramètres de la conclusion (couche 4). Une alternative permet d'estimer les paramètres de la conclusion d'une façon plus rapide, est l'utilisation de la méthode des moindres carrés, donnant lieu ainsi à une règle combinée.

En se basant sur le théorème de Stone-Weistrass, Jang a démontré que ANFIS possède la propriété d'approximation universelle [JAN 93].

5. Méthodes génétiques-floues :

Les algorithmes génétiques sont des méthodes de recherche probabilistes d'optimisation, basées sur le modèle d'évolution biologique [ZBI 94].

Un bref exposé sur les algorithmes génétiques est donné en annexe (voir annexe B).

Les algorithmes génétiques peuvent servir à l'apprentissage de règles floues, ainsi que l'optimisation des fonctions d'appartenance [BOU 95].

Un exemple d'utilisation des algorithmes génétiques pour l'apprentissage de la base de règles floues, sous forme de table de décision, est décrit dans ce qui suit.

La méthode, proposée par Thrift en 1991, se base sur le codage de toutes les règles possibles (conclusions associées à toutes les combinaisons possibles des antécédents), sous la forme d'une suite finie d'entiers [ALC 99a]. Chaque entier représente le rang de la conclusion associée à une combinaison des antécédents. Dans le cas de l'absence de règle, on associe zéro.

Ainsi, on établit une correspondance entre toutes les bases de règles floues possibles, associées à une partition donnée, et les arrangements avec répétition de m éléments de l'ensemble $\{0, \dots, k\}$, où k représente le nombre de termes linguistiques de l'espace de sortie, et m représente le nombre de règles floues.

La mutation change une allèle² (entier de la suite) par un autre entier, par exemple, son successeur. L'opérateur de croisement utilisé est l'opérateur de croisement classique. La fonction d'adaptation est une mesure des performances du système flou associé à une base choisie.

6. Simulations :

Dans ce paragraphe, nous allons tester les capacités des méthodes de modélisation présentées dans ce chapitre, à travers des exemples numériques. Deux catégories d'exemples sont choisies. La première concerne l'approximation de fonctions statiques, d'une et à deux variables. Les exemples d'approximation de fonctions ont été largement utilisés dans la littérature de modélisation. Ceci nous permet de comparer les performances des modèles linguistiques et ceux de Takagi-Sugeno. La deuxième catégorie d'exemples concerne la modélisation des systèmes dynamiques. Dans ce cas, nous traitons un "Benchmark" largement utilisé en littérature de la modélisation floue et des réseaux de neurones.

² Terme emprunt de la biologie (voir annexe B)

6.1. Approximation des fonctions statiques :

Exemple 1 :
$$y = \frac{\sin(x)}{x}, \quad x \in [-10, 10].$$

Cette fonction non linéaire est à comportement très variable (croissance et décroissance). C'est un test typique pour les méthodes d'identification, largement utilisé dans la littérature [JAN 93]. Un ensemble d'apprentissage, constitué de 101 points uniformément répartis dans l'intervalle [-10,10] a été généré.

La figure 6 illustre les variations des erreurs quadratiques moyennes (MSE) en fonction du nombre de règles floues utilisées, pour les différentes méthodes d'identification vues dans ce chapitre.

$$MSE = \frac{\sum_{i=1}^N (y_i - \bar{y}_i)^2}{N},$$

où :

- y_i : est la valeur réelle de la fonction au point x_i ,
- \bar{y}_i : est la valeur calculée de la fonction au point x_i ,
- N : est le nombre des points d'apprentissage.

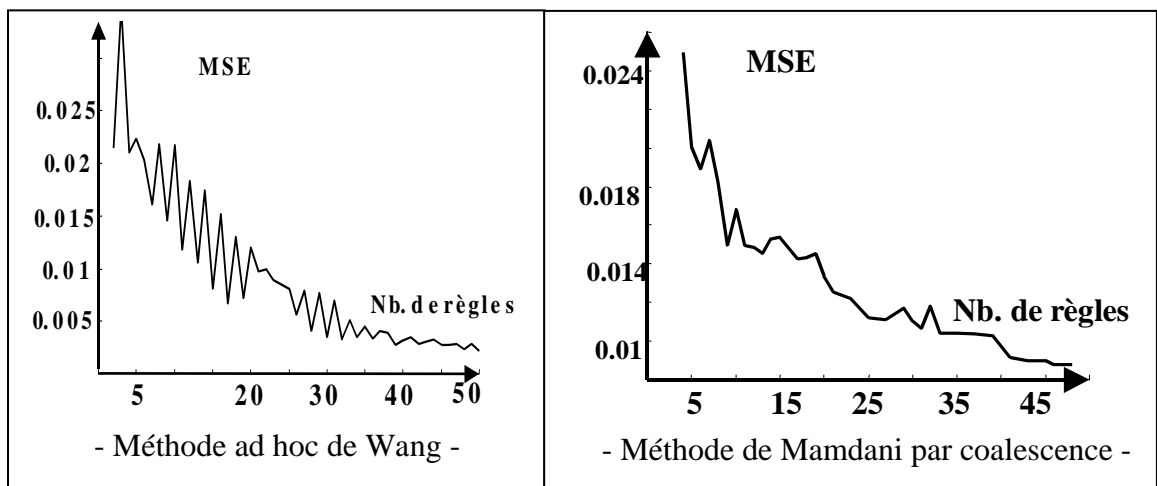


Fig. 6 - Variation de l'erreur (MSE) en fonction du nombre de règles floues pour le modèle de Mamdani

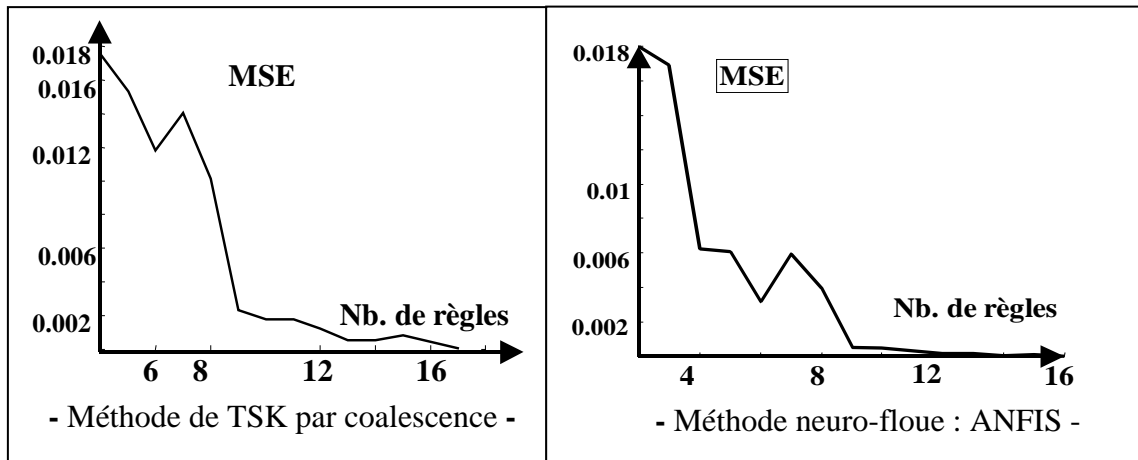


Fig. 7 - Variation de l'erreur (MSE) en fonction du nombre de règles floues pour le modèle de TSK

Interprétation des résultats :

Les résultats obtenus montrent clairement que les modèles de TSK fournissent des résultats plus précis que les modèles linguistiques. Ainsi, pour avoir une précision de MSE = 0.01, on a besoin de 20 règles pour les modèles linguistiques (méthode de Wang), tandis que 4 règles suffisent pour les méthodes de TSK.

Par ailleurs, nous remarquons que l'insertion de nouvelles règles diminue généralement l'erreur. Cette propriété n'est pas toujours vérifiée mais semble valide pour un nombre de règles suffisamment élevé. En effet, les nouvelles règles créent des régions floues plus fines qui représentent mieux les comportements locaux de la fonction.

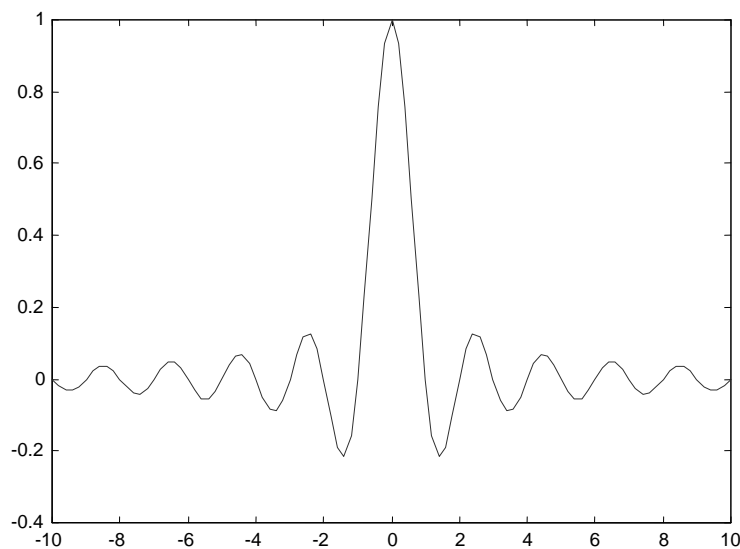


Fig. 8 - Fonction à approcher : $y = \frac{\sin(x)}{x}$

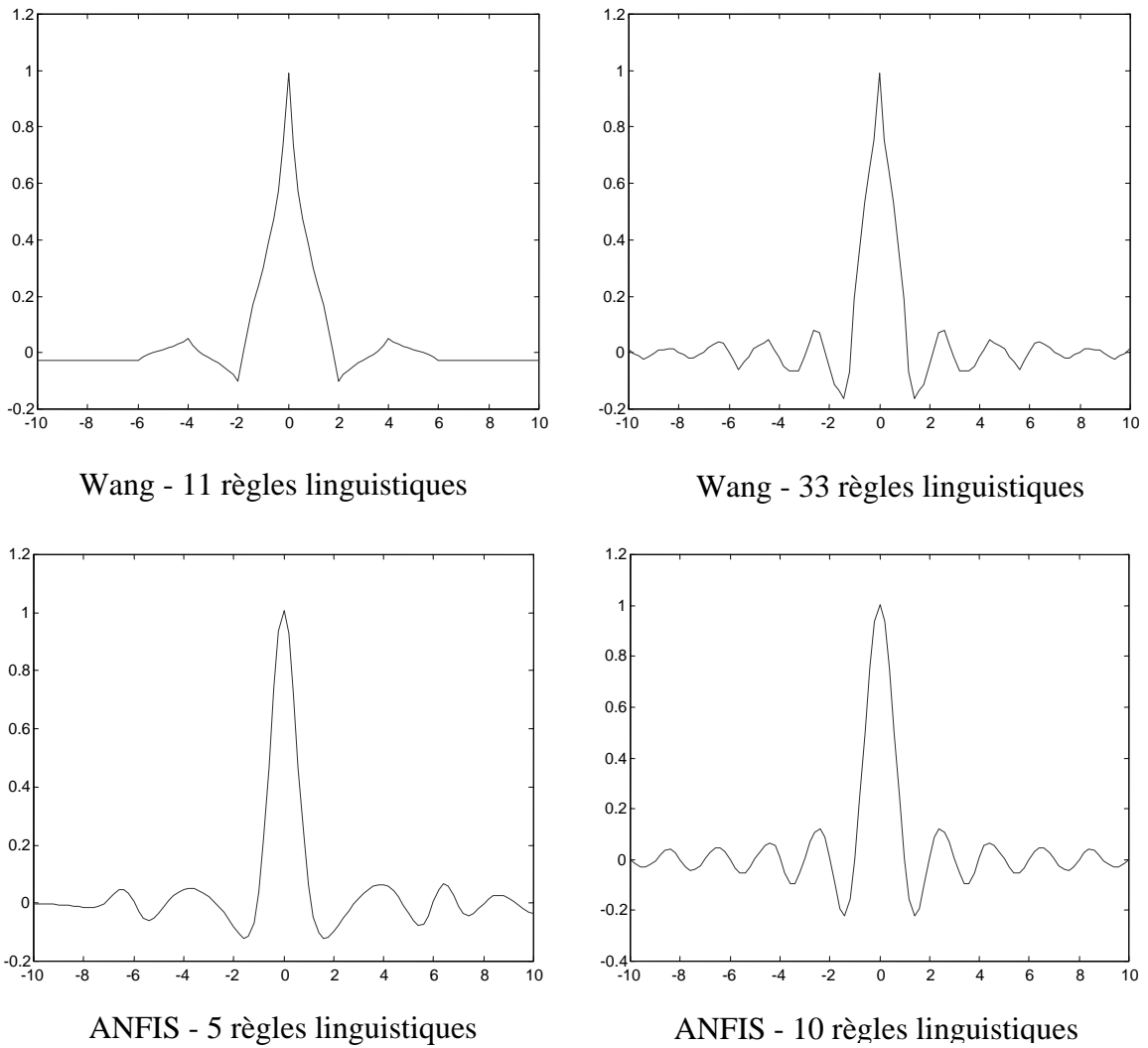


Fig. 9 - Performances en fonction des règles floues

Exemple 2 : $z = x^2 + y^2 - 1$

Dans cet exemple, un ensemble d'apprentissage, constitué de 441 points uniformément répartis dans l'ensemble $[-1,1] \times [-1,1]$ a été généré. Les figures 9, 10, et 11 illustrent la variation de la surface de sortie en fonction du nombre de règles floues utilisées, pour les différentes méthodes d'identification.

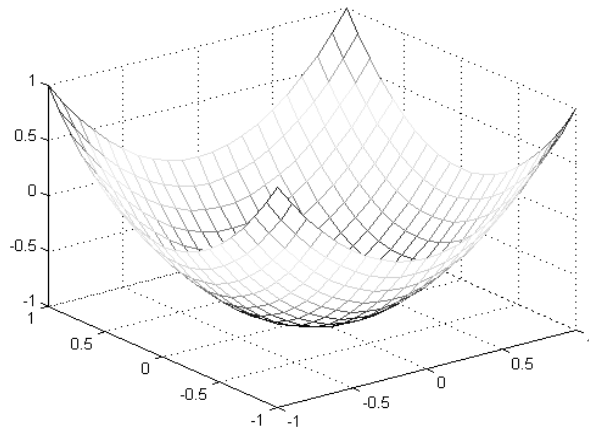


Fig. 10 - La fonction à approcher : $z = x^2 + y^2 - 1$

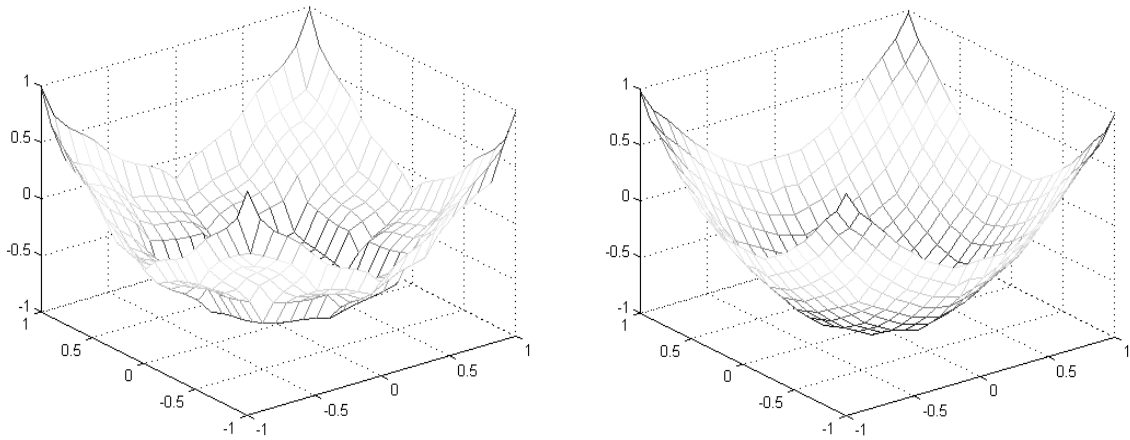


Fig. 10 - Méthode de Wang

(A) - 5 termes linguistiques pour X ,
et 5 termes linguistiques pour Y .

MSE = 1.9455, Nbr. de règles = 25.

(B) - 11 termes linguistiques pour X
et 11 termes linguistiques pour Y .

MSE = 0.5665, Nbr. de règles = 121.

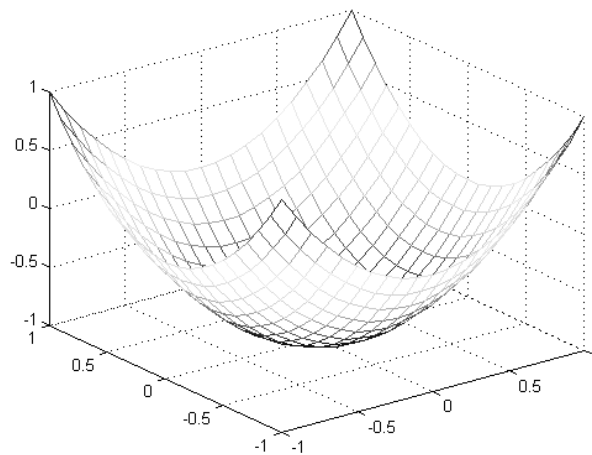


Fig. 11 - Méthode neuro-floue : ANFIS

2 termes linguistiques pour X et 2 termes linguistiques pour Y .

MSE = 0.014, Nombre de règles = 4.

6.2. Prédiction des séries chaotiques :

Dans ce paragraphe, nous allons donner un exemple de modélisation des systèmes dynamiques par logique floue. Une série temporelle chaotique de Mackey-Glass est choisie comme système dynamique.

Le problème de prédiction d'une série temporelle $x(t)$ est un problème pratique très important [JAN 95b]. Le problème de prédiction consiste en l'utilisation des valeurs passées de la série jusqu'à l'instant t afin de prédire la valeur à un instant futur $x(t + P)$. Les applications de la prédiction peuvent être trouvées dans les domaines de l'économie, prévision météo, traitement de signal, contrôle et de nombreux autres domaines [NAK 97].

Les séries chaotiques de Mackey-Glass sont engendrées par l'équation différentielle suivante [JAN 95b] :

$$x'(t) = 0.2 \cdot \frac{x(t - \tau)}{1 + x^{10}(t - \tau)} - 0.1 \cdot x(t), \text{ avec } \tau = 17.$$

La prédiction des valeurs futures de cette série constitue un "benchmark" qui a été utilisé en recherches de systèmes neuronaux et flous.

Une relation de la forme :

$$x(t + P) = F (x(t), x(t - \Delta), \dots x(t - (N - 1).\Delta))$$

est choisie pour approcher la série temporelle. Pour permettre la comparaison avec des résultats précédents rapportés dans [JAN 95b], on a choisit les valeurs : $N = 4$ et $P = \Delta = 6$.

Alors, le modèle utilisé sera de la forme :

$$x(t + 6) = F (x(t), x(t - 6), x(t - 12), x(t - 18)) \dots \dots \dots (1)$$

Un ensemble de 1000 points de la forme :

$$(x(t - 18), x(t - 12), x(t - 6), x(t) ; x(t + 6))$$

est généré à partir de l'équation différentielle, en utilisant la méthode de Runge Kutta d'ordre 4, avec un pas de 0.1, la condition initiale $x(0) = 1.2$, et en supposant que $x(t)$ est nulle pour $t < 0$. Les 500 premiers points sont utilisés pour l'apprentissage, les derniers pour la validation (pour ANFIS).

Dans [JAN 95b], la mesure de performance a été l'indice d'erreur non dimensionnel (Non-Dimensional Error Index NDEI), définie par :

$$NDEI = \frac{MSE}{STD}, \text{ où :}$$

- MSE est l'erreur quadratique moyenne,

- $STD = \sqrt{\frac{\sum_{i=1}^m (y_i - y_m)^2}{m}}$ est l'écart type de la série cible, les y_i étant les valeurs de sortie de la relation (1), y_m leur valeur moyenne.

Les résultats obtenus par les méthodes floues ainsi que ceux obtenus par d'autres méthodes sont donnés dans le tableau suivant [JAN 95b].

Méthodes	NDEI
méthode AR (régression)	0.19
Réseaux de neurones : la rétropropagation du gradient	0.06
Polynôme d'ordre 6	0.04
TSK par coalition	0.06
ANFIS	0.007

Tab. 1 - Comparaison des résultats obtenus pour la modélisation des séries chaotiques de Mackey-Glass

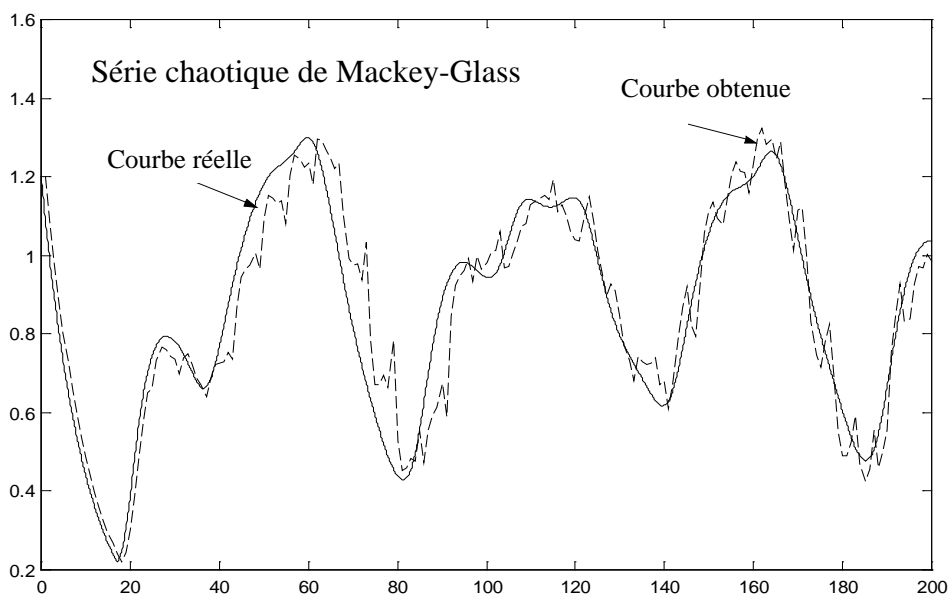


Fig. 12 - Résultat obtenu par ANFIS (28 règles).

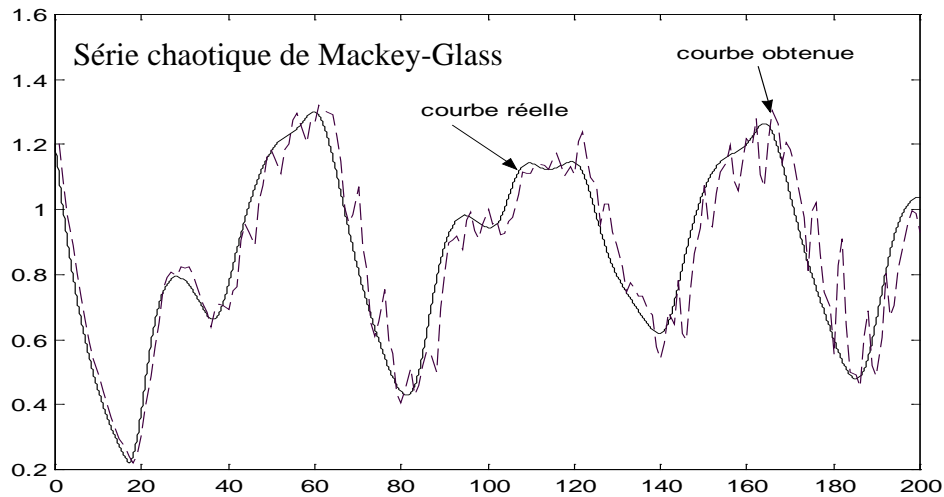


Fig. 13 - Résultat obtenu par les modèles TSK construit par coalition (28 règles).

7. Conclusion :

Ce chapitre contient les méthodes d'identification des systèmes flous les plus répandues dans la littérature. Plusieurs classes sont décrites.

Pour tester les performances des différentes méthodes de construction des systèmes flous, on a réalisé des simulations en utilisant le logiciel Matlab. Les résultats obtenus ont montré que, de façon générale, les modèles de TSK sont plus précis que ceux de Mamdani.

La méthode neuro-floue ANFIS s'est montrée très efficace pour la modélisation des fonctions statiques ainsi que les systèmes dynamiques.

Chapitre VI

Application à la Commande Floue

" La théorie, c'est quand on sait tout et que rien ne fonctionne. La pratique, c'est quand tout fonctionne et que personne ne sait pourquoi."

Albert Einstein.

1. Introduction :

La commande floue est un domaine d'application de la théorie des ensembles flous qui a été abordé très tôt au milieu des années 70. Son principe a été introduit par Mandani et Assilian sur l'exemple d'une machine à vapeur expérimentale [MAM 75]. Depuis, ce principe a été exploité dans plusieurs domaines comme la commande d'une usine d'eau chaude, régulation de vitesse d'un moteur, etc. La commande Floue a été largement développée au Japon au début des années 80. Actuellement, de nombreuses réalisations industrielles existent [BOU 95].

Son but est de traiter les problèmes de commande de processus avec une approche différente de l'automatique classique. La grande différence avec son homologue classique est qu'elle n'a pas besoin d'un modèle mathématique explicite et précis du procédé à piloter. La plupart du temps, elle se sert des connaissances des opérateurs humains pour mettre au point le régulateur. Son utilisation est recommandée lorsque le système à piloter est mal connu, ou encore lorsque les variables du processus ne sont pas précises ou exprimées en langage naturel.

La commande floue est intéressante pour les raisons suivantes [BOU 95] :

- La commande est simple à réaliser et facilement adaptable aux conditions de fonctionnement,
- La coordination de plusieurs objectifs est possible (système multivariable),
- Cette commande est robuste.

La commande floue a montré son efficacité lorsqu'elle est appliquée seule, mais elle permet également de régler les paramètres de régulations existantes. Il est intéressant de conjuguer l'utilisation de la logique floue avec des commandes traditionnelles afin de robustifier celles-ci [NOU 98]. Il s'agit par exemple de travailler sur une boucle de régulation PID déjà existante.

Aujourd'hui, les différents travaux et publications récents semblent indiquer que, en contrôle, les deux domaines d'application de la logique floue les plus importants sont [VER 98] :

- la conception de régulateurs pour des procédés difficilement modélisables,
- la conception de régulateurs non linéaires pour des procédés modélisables.

Dans le premier cas, la régulation floue se distingue de la régulation classique par le fait que le régulateur est synthétisé à partir de l'expertise des actions d'un opérateur humain. Au final, le régulateur obtenu possède au moins les mêmes caractéristiques que le meilleur opérateur humain. Ce type de commande floue permet une bonne adéquation du régulateur au système, mais engendre de fortes non linéarités. Dans le second cas, la structure non linéaire du régulateur permet d'améliorer les performances en terme de précision, de stabilité et de robustesse. Ce type de régulateurs peut être appliqué à des modèles classiques linéaires ou non linéaires ou encore flous de type Takagi-Sugeno [VER 98].

Le second paragraphe de ce chapitre décrit un régulateur flou standard obtenu par fuzzification du régulateur PID classique, qui est utilisé dans pas moins de 90% des réalisations pratiques. Dans le troisième paragraphe, la logique floue est utilisée comme une méthode d'interpolation entre différents modèles, permettant de décrire et commander le système dans différentes régions de l'espace d'état.

2. Les commandes PID floues :

Il s'agit d'une approche de retour de sortie ('Feedback' en anglais). Le régulateur flou consiste en une fonction non linéaire reliant la commande du procédé aux entrées de type erreur, variation de l'erreur, etc.

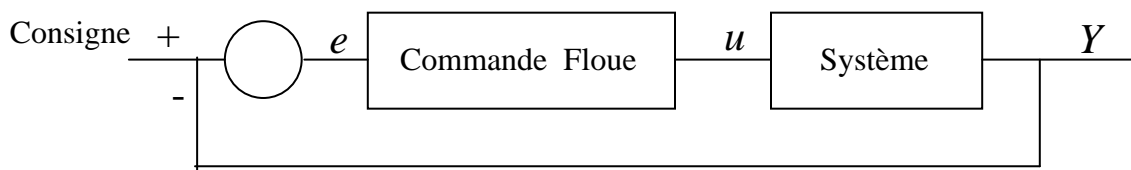


Fig. 1 - Commande Floue à boucle de retour

Les différentes formes de commande floue PID sont résumées dans les points suivants [VER 98] :

Régulateur flou proportionnel (RF-P) : La loi de commande est proportionnelle à la valeur de l'erreur de sortie. Les règles ont la forme suivante :

$$\text{SI } (e(t) \text{ est } E) \text{ ALORS } (u(t) \text{ est } U)$$

La loi de commande ressemble à celle d'un régulateur proportionnel classique avec un gain non linéaire, variable :

$$u(t) = K_p(t) \cdot e(t)$$

Régulateur flou proportionnel dérivé (RF-PD) : La loi de commande est obtenue grâce à l'erreur de sortie et à sa dérivé première. Les règles ont la forme suivante :

$$\text{SI } (e(t) \text{ est } E) \text{ et } (\dot{e}(t) \text{ est } DE) \text{ ALORS } (u(t) \text{ est } U)$$

La loi de commande du régulateur RF-PD est donnée par :

$$u(t) = K_p(t) \cdot e(t) + K_d(t) \cdot \dot{e}(t)$$

Régulateur flou proportionnel, intégral (RF-PI) : Les règles ont la forme suivante :

$$\text{SI } (e(t) \text{ est } E) \text{ et } (\int e(t).dt \text{ est } IE) \text{ ALORS } (u(t) \text{ est } U)$$

La loi de commande s'apparente à celle d'un régulateur proportionnel intégral classique :

$$u(t) = K_p(t) \cdot e(t) + K_I(t) \cdot \int e(t).dt$$

Régulateur flou proportionnel, intégral et dérivé (RF-PID) : La loi de commande est obtenue grâce à l'erreur de commande, son intégrale et sa dérivé première. Les règles ont la forme suivante :

$$\text{SI } (e(t) \text{ est } E) \text{ et } (\int e(t).dt \text{ est } IE) \text{ et } (\dot{e}(t) \text{ est } DE) \text{ ALORS } (u(t) \text{ est } U)$$

La loi de commande s'apparente à celle d'un régulateur PID classique :

$$u(t) = K_p(t) \cdot e(t) + K_I(t) \cdot \int e(t).dt + K_d(t) \cdot \dot{e}(t)$$

La régulation floue PID est conçue sans modèle explicite du procédé à piloter. La difficulté réside dans le choix relatif des fonctions d'appartenance : un choix raisonnable permet d'obtenir de bonnes performances. Le correcteur obtenu est non linéaire.

3. Commande floue multimodèle :

La plupart des modèles de conduite utilisés en pratique sont des modèles locaux autour de points de fonctionnement ; la commande floue multimodèle permet d'interpoler les commandes déterminées pour chacun de ces modèles, entre plusieurs domaines de fonctionnement.

Cette approche consiste donc à construire des modèles locaux M_i dont la capacité de représenter convenablement le système est réduite à des domaines de validité respectifs D_i . On définit une commande u_i pour chacun des modèles locaux M_i ; chaque commande étant déterminée indépendamment des autres. On peut déterminer, par exemple, une commande optimale pour le modèle M_1 , une commande proportionnelle pour le modèle M_2 , etc. La commande globale appliquée au système est obtenue par interpolation des différentes commandes élémentaires. Plus on sera "proche" d'un modèle (ou une région), plus la commande sera semblable à celle qui a été trouvée pour ce modèle.

Les règles floues utilisées sont dans ce cas de la forme :

$$R_i : \text{" Si } X \text{ est } A_i \text{ alors le modèle est } M_i \text{" .}$$

où X est le vecteur d'état.

L'intérêt de cette commande est évident dans la perspective d'applications industrielles. On peut citer par exemple [NOU 98] :

- démarrage d'un moteur (modèles locaux connus, transition non linéaire) ;
- processus manufacturiers avec évolution des caractéristiques des matières premières utilisées.

La figure ci-dessous représente un multimodèle de dimension deux. Les variables d'état sont découpées dans cet exemple en 3 prédicats. On partitionne l'espace d'état en 9 parties, et un modèle M_i est associé à chacune d'elles.

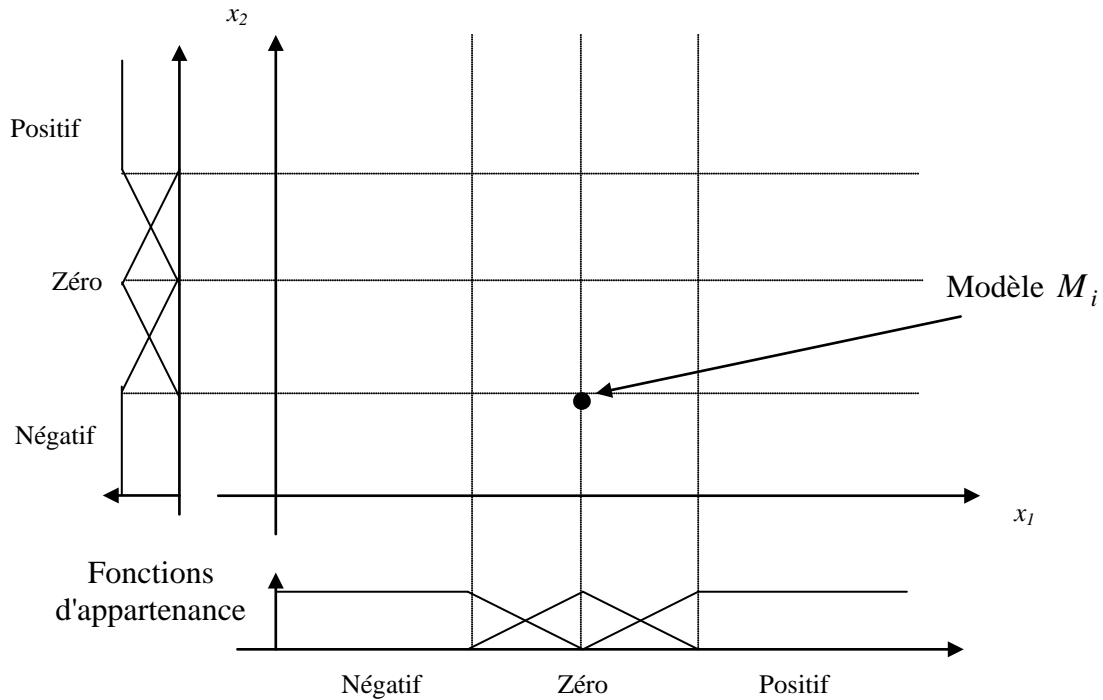


Fig. 2 - Description d'un système multimodèle utilisant la logique floue

4. Application pratique :

Faire suivre un robot une trajectoire désirée ou réaliser une tâche bien spécifiée est une opération très délicate à cause des non-linéarités, des perturbations et la structure complexe du robot, ce qui exige des techniques de commande sophistiquées [SOU 99].

En dépit de sa popularité et sa facilité d'implantation, le contrôleur PID présente l'inconvénient du choix de ces paramètres (coefficients ou gains). En Robotique, la pratique la plus courante consiste à choisir des gains de manière à obtenir une réponse rapide et sans oscillations.

Dans ce qui suit, on va utiliser la logique floue pour la commande d'un robot manipulateur sous forme d'un contrôleur PID flou.

Description du bras manipulateur :

La structure mécanique du bras comprend deux segments placés l'un à la suite de l'autre, chaque segment étant articulé et mobile par rapport à l'articulation. Cette mobilité s'exprime en termes de degrés de liberté. L'organe terminal porte l'outil de travail, qui dépend des tâches que l'on désire réaliser.

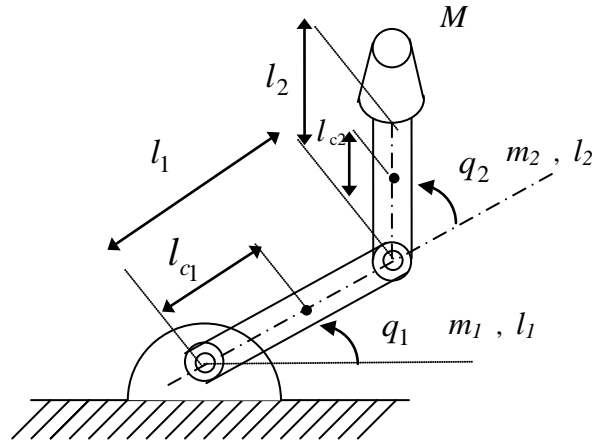


Fig. 3 - Structure mécanique du bras manipulateur

Modélisation :

La formulation de Lagrange-Euler permet de représenter la dynamique d'un robot à n degrés de liberté sous la forme matricielle suivante [MAR 98] :

$$M(q) \ddot{q} + \Gamma(q, \dot{q}) + G(q) = U$$

où :

- $M(q)$ est la matrice inertielle du manipulateur,
- $\Gamma(q, \dot{q})$ est le vecteur des termes centrifugaux et de coriolis,
- $G(q)$ est le vecteur gravitationnel,
- U est le vecteur des couples,
- q est le vecteur des angles des articulations.

La structure du robot à 2 degrés de liberté utilisé pour la simulation se caractérise par [SOU 99] :

$$M = (m_{ij})_{i,j}, \quad \Gamma = [n_1 \quad n_2].$$

$$m_{11} = A + 2 \cdot B \cdot \cos q_2, \quad m_{12} = m_{22} + B \cdot \cos q_2,$$

$$m_{21} = m_{12}, \quad m_{22} = m_2 + (l_{c2})^2 + I_2 \cdot n_1 = -B \cdot (\dot{q}_2)^2 + 2 \cdot \dot{q}_1 \cdot \dot{q}_2 \cdot \sin q_2,$$

$$n_2 = B \cdot (\dot{q}_1)^2 \cdot \sin q_2, \quad G(q) = 0.$$

Avec : $A = m_1 (l_{c2})^2 + m_2 ((l_1)^2 + (l_{c2})^2) + I_1 + I_2$, $B = m_2 \cdot l_1 \cdot l_{c2}$.

Le tableau suivant indique les paramètres du robot utilisé [SOU 99] :

Paramètres	Valeur
l_1	0.30 m
l_2	0.32 m
lc_1	0.15 m
lc_1	0.16 m
m_1	2.81 Kg
m_2	1.64 Kg
M	1.00 Kg
I_1, I_2	0.0234 Kg.m ²

Tab. 1 - Paramètres utilisés

Le bras est commandé à l'aide de deux moteurs générant deux couples de rotations u_1 et u_2 , les deux segments se mouvant autour des deux articulations q_1 et q_2 . Chaque articulation est considérée comme un sous-système. Deux contrôleur flou PD1 et P2 sont développés pour commander les deux articulations.

Les lois de commandes floues PD1 et P2, obtenues d'une manière heuristique, sont régies par les bases de règles floues suivantes :

PD-flou 1 :

Si ($e(t)$ est Négative) et ($\dot{e}(t)$ est Petite) Alors $u = 3600 e(t) + 15 \dot{e}(t)$

Si ($e(t)$ est Négative) et ($\dot{e}(t)$ n'est pas Petite) Alors $u = 3600 \cdot e(t) + 40 \cdot \dot{e}(t)$

Si ($e(t)$ est Nulle) et ($\dot{e}(t)$ est Petite) Alors $u = 2400 \cdot e(t) + 15 \cdot \dot{e}(t)$

Si ($e(t)$ est Nulle) et ($\dot{e}(t)$ n'est pas Petite) Alors $u = 2400 \cdot e(t) + 40 \cdot \dot{e}(t)$

Si ($e(t)$ est Positive) et ($\dot{e}(t)$ est Petite) Alors $u = 3600 \cdot e(t) + 15 \cdot \dot{e}(t)$

Si ($e(t)$ est Positive) et ($\dot{e}(t)$ n'est pas Petite) Alors $u = 5000 \cdot e(t) + 40 \cdot \dot{e}(t)$

Les fonctions d'appartenance sont données sur la figure 4.

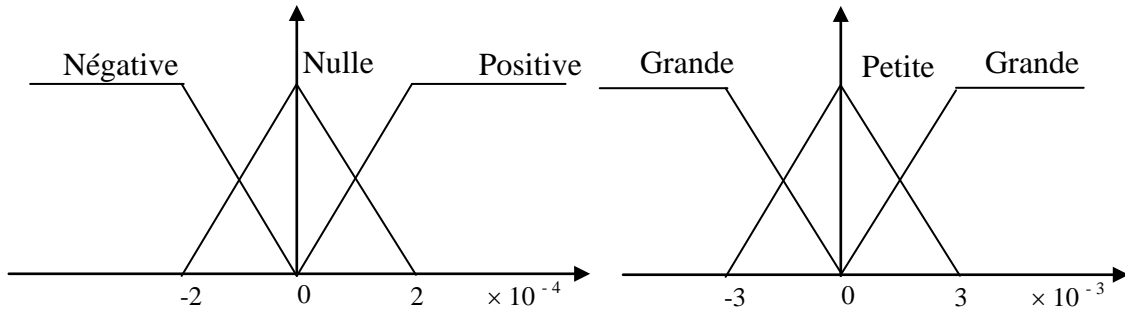


Fig. 4 - Fonctions d'appartenance pour le contrôleur PD1

La loi de commande P2 floue est régie par les deux règles suivantes :

P-flou 2 :

Si $e(t)$ est Petite Alors $u = 1000 \cdot e(t)$

Si $e(t)$ n'est pas Petite Alors $u = 1500 \cdot e(t)$

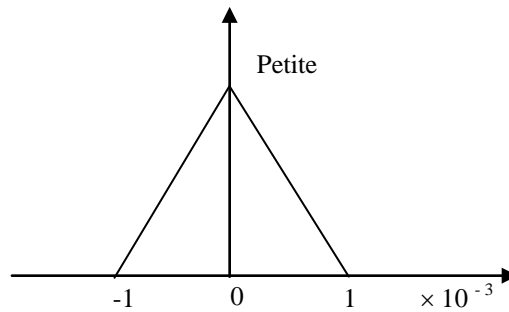


Fig. 5 - Fonction d'appartenance pour le contrôleur P2

La simulation est réalisée en utilisant MATLAB 5.3 et ses boîtes à outils SIMULINK et Fuzzy Logic Toolbox. La méthode numérique utilisée est celle de Runge Kutta d'ordre 4, avec un pas de 0.01.

Les résultats de la simulation sont présentés sur les figures 6, 7.

Test de robustesse :

Pour tester la robustesse de l'approche utilisée, on garde les mêmes paramètres des deux contrôleurs, et on fait des changements sur les paramètres du système. Le poids de la charge M est doublé : $M = 2$ kg.

Les résultats sont donnés sur la figure 8.

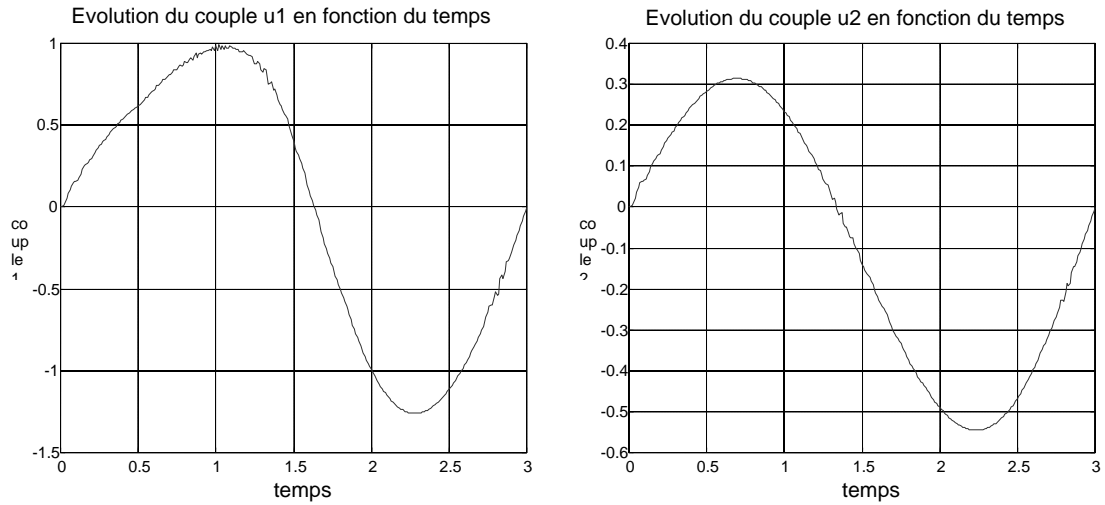


Fig. 6 - Couples en fonction du temps (M = 1 Kg)

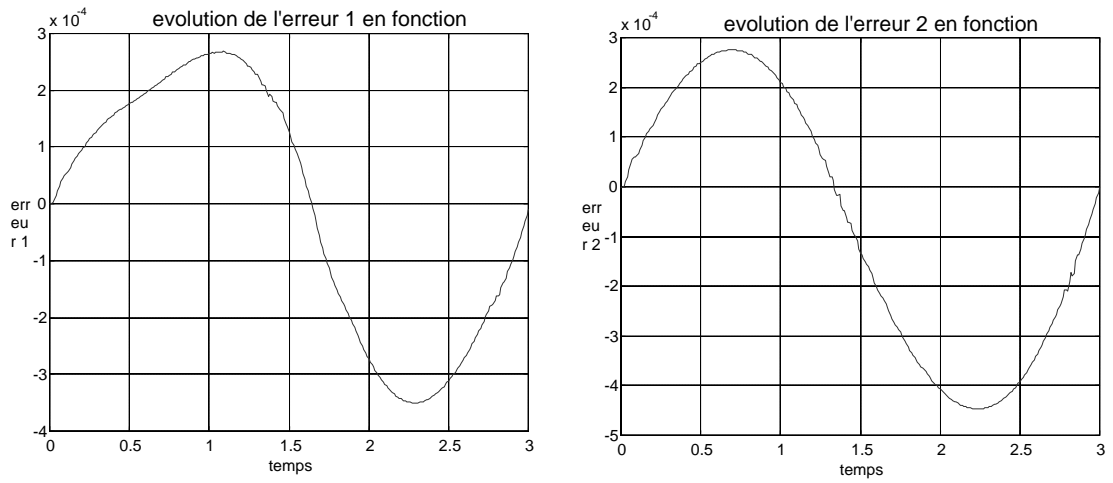


Fig. 7 - Erreurs en fonction du temps (M = 1 Kg)

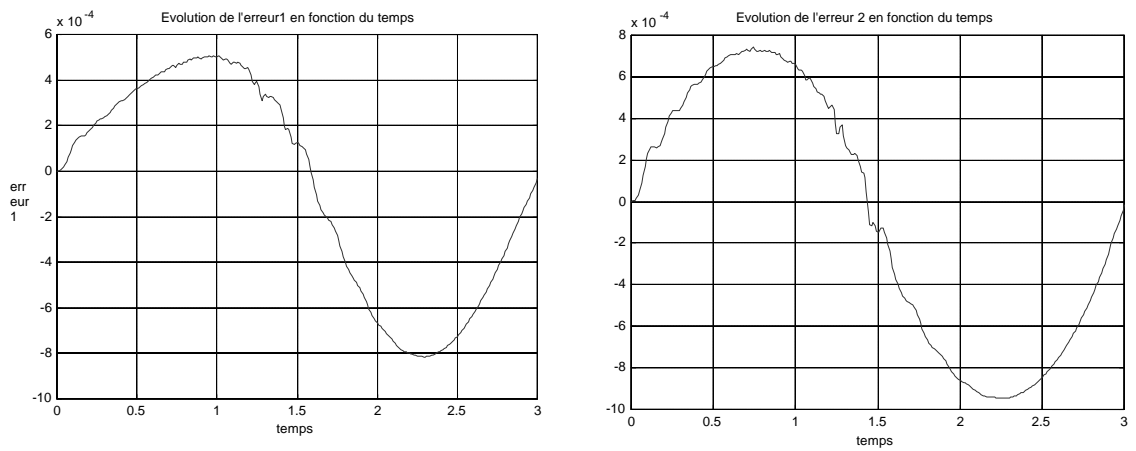


Fig. 8 - Erreurs en fonction du temps (M = 2 Kg)

5. Conclusion :

Les résultats de la simulation démontrent bien l'efficacité de l'approche utilisée pour la conception de contrôleurs accomplissant une tâche de poursuite rapide, précise et robuste. La logique floue constitue un moyen efficace pour robustifier et améliorer les régulateurs PID classiques, en permettant la détermination souple des gains variables.

Conclusion

" Il est rare de trouver un homme qui se livre trois ans à l'étude, sans avoir en vue un salaire ..."

Confucius, Philosophe, historien et homme d'état chinois. Extrait de "Entretiens du Maître avec ses disciples".

En introduisant la théorie des sous-ensembles flous, Zadeh a offert un outil puissant pour la modélisation des systèmes complexes, pour lesquels on ne dispose que d'une spécification approximative ou imprécise. Le but d'un modèle est de représenter la relation entre les entrées et les sorties d'un système. A l'encontre d'un modèle conventionnel qui décrit cette relation par une loi mathématique, un modèle flou la décrit linguistiquement.

De façon générale, les ensembles flous peuvent intervenir efficacement dans la modélisation des systèmes complexes, principalement en raison de leur capacité à synthétiser des informations, à permettre une approche globale de certaines caractéristiques du système grâce à la gradualité qui leur est inhérente, et également, bien sur, en raison de leur aptitude à traiter des connaissances imparfaites, c'est-à-dire par exemple incomplètes, approximatives, vagues, soumises à des erreurs de mesure.

La logique floue présente, entre autres, les avantages suivants [JAN 95a] :

- La logique floue est un concept simple à comprendre : les concepts mathématiques du raisonnement en logique floue sont très simples. Ce qui rend la logique floue élégante, c'est le caractère naturel de ses approches.
- La logique floue est flexible : il est facile de l'appliquer à n'importe quel système.
- La logique floue tolère les imprécisions.
- La logique floue peut modéliser des fonctions non linéaires d'une complexité quelconque.
- La logique floue peut incorporer les connaissances d'un expert ou celles déduites des expériences.

Dans ce mémoire, nous nous sommes intéressés dans un premier temps aux fondements théoriques de la logique floue. Ainsi, la théorie des ensembles flous est présentée, ainsi que celle des possibilités. Par la suite, on a étudié le raisonnement approximatif, qui sert à effectuer des inférences dans un environnement flou. Enfin, des exemples de l'investigation du flou dans les domaines classiques des mathématiques sont exhibés. Il s'agit des versions floues de l'Arithmétique, les Mesures et Intégrations et la Programmation Mathématique.

La deuxième partie a été consacrée à la modélisation floue et, comme cas particulier, la modélisation des systèmes dynamiques. Nous avons identifié deux classes essentielles de modèles flous : les modèles linguistiques et les modèles de TSK. Les modèles linguistiques sont utilisés typiquement lorsque des connaissances qualitatives sous forme de règles linguistiques sont disponibles, alors que les modèles de TSK sont plus appropriés pour la modélisation des systèmes dynamiques.

Les modèles flous ont deux propriétés essentielles :

- le traitement se fait au niveau symbolique. En fait, ces modèles sont conçus pour manipuler des valeurs linguistiques, comme c'est le cas chez l'homme. Ceci est devenu possible grâce aux variables linguistiques et à la représentation des spécifications des valeurs linguistiques par des ensembles flous.
- Ils sont capables de représenter l'imprécision et l'incertitude d'un expert humain. Cette propriété est particulièrement intéressante, car les modèles flous sont souvent inspirés de la connaissance humaine. Il est donc nécessaire d'inclure l'imprécision et l'incertitude que contient cette connaissance dans le modèle flou.

La construction des systèmes flous repose naturellement sur le savoir-faire d'un expert. Toutefois, des méthodes automatiques pour l'identification des systèmes flous à partir des données numériques de la forme entrées-sorties du système réel ont été développées. En particulier, la logique floue peut être utilisée en liaison avec d'autres techniques telles que les réseaux de neurones, par exemple, pour résoudre les problèmes d'acquisition des connaissances, ou les algorithmes génétiques, composant ce que la communauté scientifique internationale appelle actuellement le "soft computing".

La logique floue s'avère plus efficace que les méthodes traditionnelles pour les systèmes complexes pour lesquels la modélisation est difficile, voire impossible, pour les systèmes contrôlés par des experts humains ou quand l'observation humaine est à l'origine d'entrées ou de règles de contrôle du système.

Les domaines d'application de la logique floue se sont multipliés depuis la fin des années soixante. Les domaines d'application dans lesquels il existe des utilisations de la logique floue sont très variés : médecine et biologie, génie industrielle, technique, économie, défense, écologie, sciences humaines, recherche scientifique, ...

Dans le dernier chapitre, on a appliqué la logique floue à la commande d'un bras manipulateur. Une méthodologie particulière de construction d'un contrôleur flou est

Conclusion

choisie, utilisant l'erreur entre la sortie réelle et la sortie désirée (Feedback). Les résultats obtenus sont satisfaisants et témoignent de l'efficacité de la technique utilisée.

Au-delà de l'intérêt médiatique suscité par la logique floue et de la mise en évidence d'applications commerciales, il faut considérer la logique floue avec la même objectivité que celle accordée à toute méthodologie ou technique actuellement considérée comme classique, comme l'automatique ou la théorie des probabilités par exemple. La logique floue a presque quarante ans, elle entre dans sa maturité et ne doit plus être regardée comme une science balbutiante.

Elle repose sur des fondements théoriques établis dans de multiples publications internationales par des chercheurs de la plus haute compétence ; elle n'est pas isolée dans sa recherche, mais des liaisons ont été établies avec d'autres axes tels que les logiques non classiques, la théorie des probabilités, automatique classique, pour n'en citer que quelques-uns.

Ses applications touchent tous les domaines parce qu'elle s'efforce d'apporter des solutions à des situations réelles où le flou est présent, d'autant plus que, nous le voulions ou non, nous vivons dans un monde flou... .

Bibliographie

[ALC 99a] R. Alcalá, J. Casillas, O. Cordon, F. Herrera, I. Zwir, « *Techniques for Learning and Tuning Fuzzy Rule-Based Systems for Linguistic Modeling and Their Applications* », in: C. T. Leondes (ed.), *Knowledge Engineering. Systems, Techniques and Applications*, Academic Press, 1999.

url : www.citeseer.nj.nec.com/alcala99techniques.html.

[ALC 99b] R. Alcalá, J. Casillas, O. Cordon, F. Herrera, *Approximate Mamdani type fuzzy rule-based systems*, Technical Report : #DECSAI-990117, Université de Granada, October 1999.

[ATN 84] Tanaka H. , Asai , K. , " *Fuzzy LP problems with fuzzy numbers*", *Fuzzy sets and systems*, Vol. 13, 1-10, 1984.

[BAB 98] R. Babuska, *Fuzzy Systems, Modeling and Identification*, 1998.

[BOU 94] B. B-M, R. Hartani, H. T. Nguyen, *Sur l'approximation universelle des systèmes flous*, Rapport LAFORIA-IBP 94/14, Université Paris VI, 1994.

[BOU 95] B. Bouchon-Meunier, *La logique floue et ses applications*, Collection Vie Artificielle, Addison-Weiley, 1995.

[BZA 70] R. Bellman, L. A. Zadeh, " *Decision Making in a Fuzzy environment* ", *Man. Sci.* , Vol. B 17, 141-164, 1970.

[CAS 95] J. L. Casrto, M. Delgado, *Fuzzy Systems with Defuzzification are Universal Approximators*, *IEEE trans. on man and cyb.* 1995.

[CAS 00] J. Casillas, O. Cordon, F. Herrera, *A Methodology to improve Ad Hoc Data-Driven Linguistic Rule Learning Methods by inducing Cooperation among Rules*, Technical Report #DECSAI-00-01-01, Feb. 2000.

[CFH 96] Cheng- Feng Hu et Ta- Wei Hung, " *Literature Review on Fuzzy Linear Programming* " , OR 605, Large Scape LP Project, Mars 1996.

[CFU 96] C. Carlsson , R. Fuller , *Fuzzy multiple criteria decision making: recent developments*, *Fuzzy Sets and Systems*, 78, 139-153, 1996.

[CFU 98] C. Carlsson and R. Fuller, *Optimization under fuzzy if-then rules*, *Fuzzy Sets and Systems*, 1998.

[COO 95] De Cooman, G. , " *The formal analogy between possibility and probability theory.*" In: *Foundations and Applications of Possibility Theory*, edited by G. de Cooman, D. Ruan and E. E. Kerre, World Scientific, Singapore, pp. 71–87. 1995.

[COO 96] G. de Cooman et E. E. Kerre, " *Possibility and necessity integrals.*" *Fuzzy Sets and Systems*, 77, p.207–227, 1996.

- [COO 97] G. de Cooman. " *Possibility theory* ", Parts I, II, III, *International Journal of General Systems*, 25, p. 291–323, 325–351, 353–371, 1997.
- [DAV 95] A. L. Davis, " *A Litterature Review : Fuzzy Mathematical Programming* ", Décembre 1995.
- [DIA 94] P. Diamond et P. Kloeden, *Metric Space of Suzzy Sets*, World Scientific, Singaphore, 1994.
- [DIM 93] D. Driankov, Hans Hellendoorn, Michael Reinfrank, *An introduction to fuzzy control*, Springer-Verlag, 1993.
- [DLR 91] D. Dubois, J. Lang, H. Prade, *Fuzzy sets in approximate reasoning*, Part 2, *Fuzzy Sets and Systems* 40, pp. 203-244, North-Holland, 1991.
- [DPR 82] D. Dubois, H. Prade, " *Towards Fuzzy Differentiel Calculus*", *Fuzzy Sets and Systems* 8, Parts 1,2,3, p. 1-17,105-116,225-233, 1982.
- [DPR 87] D. Dubois, H. Prade, Fuzzy numbers : an Overview, dans J. C. Bezdek (dir.) *Analysis of fuzzy information*, Vol. 1, Math. and logic, CRC Press, p. 3-39, 1987.
- [DPR 91] D. Dubois, H. Prade, *Fuzzy sets in approximate reasoning*, Part 1, *Fuzzy Sets and Systems* 40, pp. 143-202, North-Holland, 1991.
- [DPR 02] D. Dubois, H. Prade, *Le calcul des intervalles flous*, dans : *Le traité IC2, " Utilisation de la logique floue"*, (B. Bouchon-Meunier, C. Marsala, Coordonnateurs), Hermès, à paraître.
- [DPS 00] D. Dubois, H. Prade, and R. Sabbadin, " *Qualitative decision theory with Sugeno integrals* ", in *Fuzzy Measures Integrals*, M. Grabisch, T. Murofushi, and M. Sugeno, Eds. Hiedelberg, Germany : Physica-Verlag, 2000, pp. 314–332.
- [FAU 74] R. Faure, *Précis de recherche opérationnelle*, Dunod, 3ième édition, 1974.
- [FUC 93] A. Fuchs, G. Reeb, *Logique*, OPU, Réimpression 1993.
- [FUL 98] Robert Fuller, *Lecture Notes : Fuzzy systems and identification*, url : <http://tiger.ees.kyushu-u.ac.jp/~hu/Reference/Online/fuzzmod.pdf> , 1998.
- [FUL 01] R. Fuller, *Lecture Notes : Soft Decision Analysis*, url : www.abo.fi/~rfuller/asda.html, 2001.
- [FUK 90] *Lecture Notes on Statistical Pattern Recognition*, from K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, 1990.
- [FZI 93] R. Fuller and H. J. Zimmermann, *Fuzzy reasoning for solving fuzzy mathematical programming problems*, *Fuzzy Sets and Systems* 60(1993) 121-133.
- [GUP 91] M. M. Gupta, J. Qi, *Theory of T-norms and Fuzzy Inference Methods*, *Fuzzy Sets and Systems* 40, 3, pp. 431-450, 1991.

- [HEI 97] S. Heilpern, *Representation and application of fuzzy numbers*, Fuzzy sets and systems 97, pp. 259-268, 1997.
- [JAN 93] Jang , J- S. R., *ANFIS : adaptive- network-based fuzzy inference system*, IEEE trans. on systems, man and cyber., Vol. 23, no. 3, pp. 665-685, May 1993.
- [JAN 95a] Jang J.-S. R., N. Gulley, *Fuzzy Logic Toolbox*, The Mathworks Inc., 1995.
- [JAN 95b] Jang , J- S. R., Chuen-Tsai Sun, *Neuro-fuzzy modeling and Control*, Proc. Of the IEEE, March 95.
- [JOH 99] T. A. Johansen, R. Shorten, R. Murray Smith, *On the interpretation and identification of dynamic Takagi-Sugeno Fuzzy models*, 1999.
- [KAU 73] Kaufmann A., *Introduction à la théorie des sous-ensembles flous*, Vol. 1, Masson, 1973.
- [KAU 75] A. Kauffmman, *Introduction à la théorie des sous-ensembles flous*, tome 2, Masson et Cie, éditeurs, 1975.
- [KOH 94] J. Kohlas and P.A. Monney, *ZOR-Mathematical Methods of Operations Research*, 39, 35--68. 1994.
- [MAM 75] E. H. Mamdani, S. Assilian, *An experiment in linguistic synthesis with a fuzzy logic controller*, Int. J. of Man-Machines Studies, 7, p. 1-13, 1975.
- [MAR 98] J. Mareczek, M. Buss, G. Schmidt, *Robust Global Stabilization of the Underactuated 2-DOF Manipulator R2Dl*, Proceedings of the 1998 IEEE, International Conference on Robotics & Automation, Leuven, Belgium. May 1998
- [MOR 01] Y. Morère, *Mise en oeuvre de Lois de Commande pour les Modèles Flous de Type Takagi-Sugeno*, Thèse de Doctorat, Université de Valenciennes, 2001.
- [NAK 97] Y. Nakoula, *Apprentissage des modèles linguistique flous par jeu de règles pondérées*, thèse doctorat, Université Savoie, 1997.
- [NGU 93] Nguyen, H.T., Kreinovich, V. , *On approximation of controls by fuzzy systems*, Proc. Fifth IFSA, 1414-1417, 1993
- [NKR 98] H. T. Nguyen, V. Kreinovich, " *A modification of Sugeno integral describes stability and smoothness of fuzzy control* ", FUZZ-IEEE'98.
- [NOV 89] V. Novak, *Fuzzy Sets and their Applications*, Adam Hilger, Bristol-Philadelphia, 1989.
- [PAV 79] J. Pavelka, *On Fuzzy Logic I- Many-Valued rules of inference*, Zeitschr. F. Math. Logik und Grundlagen d. Math. 25 pp. 45-52, 1979.
- [SUG 74] M. Sugeno, *Theory of fuzzy integrals and its applications*, PhD Thesis, Tokyo Institute of Technology, 1974.

- [SOU 99] A. Soukkou, K. Benmahammed, *Conception d'un controleur PID robuste par les algorithmes génétiques*, Actes de la 2^{ème} conf. Int. D'électronique, Blida 1999.
- [SUG 86] M. Sugeno, G. T. Kang, *Fuzzy modelling and control of multilayer incinerator*, Fuzzy Sets and Systems, Vol. 18, p. 329, 1986.
- [TAK 85] T. Takagi, M. Sugeno, *Fuzzy Identification of Systems and its Applications to Modelling and Control*, IEEE Trans. on systems, man, and cyb., Vol. 15, No. 1, 1985.
- [VER 98] L. Vermeiren, *Proposition de lois de commande pour la stabilisation de modèles flous*, Thèse Doctorat, Université de Valenciennes, 1998.
- [WAN 92] Wang, L. X., et Mendel J., *Fuzzy basis functions, universal approximation and orthogonal least squares*, IEEE trans. On neural networks, Vol.3, 807-814., 1992.
- [YAG 91] R. R. Yager, *Connectives and Quantifiers in Fuzzy Sets*, Fuzzy Sets and Systems 40, 1, pp. 39-75, 1991.
- [YAG 99] R. R. Yager, "Criteria Aggregations Functions Using Fuzzy Measures and the Choquet Integral", International Journal of Fuzzy Systems, Vol. 1, No. 2, Décembre 1999.
- [ZAD 65] L. A. Zadeh, *Fuzzy sets*, Information and Control 8, pp. 338-353, 1965.
- [ZAD 71] L. A. Zadeh, *Similarity Relations and Fuzzy Orderings*, Information Sciences, 3, pp. 177-200, 1971.
- [ZAD 75] L. A. Zadeh, *The Concept of a Linguistic Variable and its Applications to Approximate Reasoning*, Parts 1 and 2, Information Sciences 8, p. 199-249, 301-357, 1975.
- [ZAD 78] L. A. Zadeh, *PRUF- A Meaning Representation Language for Natural Languages*, Int. J. Man-Machine Studies 10, pp. 395-460, 1978.
- [ZAD 79] L.A. Zadeh. *A theory of approximate reasoning*. In J.E. Hayes, D. Michie and L.I. Mikulich, ed. Machine Intelligence, Vol. 9, pp. 149-194. Elsevier, Amsterdam, 1979.
- [ZAD 94] L. Zadeh, *Soft Computing and Fuzzy Logic*, IEEE Soft., p. 48-56, Nov. 1994
- [ZBI 94] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Secon Edition, Spinger-Verlag, 1994.
- [ZEL 95] M. Zelmat, *Introduction aux principes des systèmes de commande*, OPU, 1995.
- [ZIM 75] H. J. Zimmermann, *Description and Optimization of Fuzzy systems*, Int. J. General Sys. 2, pp. 209-215, 1975.

Annexe - A

Réseaux de Neurones Artificiels

1. Introduction :

Les réseaux de neurones artificiels sont des modèles mathématiques simplifiés du fonctionnement du cerveau. Les réseaux de neurones constituent une technique de calcul se distinguant des autres techniques par le traitement parallèle et massif des opérations, et la capacité d'apprentissage.

Les réseaux de neurones offrent un cadre général pour la représentation des fonctions non-linéaires, sous la forme paramétrique :

$$y = f(x, \theta),$$

où la relation fonctionnelle dépend d'un nombre de paramètres ajustables θ . Le but est de déterminer θ , en utilisant une règle d'apprentissage, qui convient au mieux pour approcher la fonction désirée. Les réseaux de neurones possèdent la propriété de l'approximation universelle.

Les réseaux de neurones ont été utilisés principalement pour :

- 1) L'approximation des fonctions.
- 2) La modélisation des systèmes dynamiques.
- 3) Les problèmes de classification et de reconnaissance de formes.
- 4) La commande de processus.

2. Concepts de base :

Les éléments de calcul de base des réseaux de neurones sont appelés : *les neurones*, ou encore les *nœuds*. Chaque nœud est caractérisé par une fonction d'activation, les poids, plusieurs entrées et une sortie.

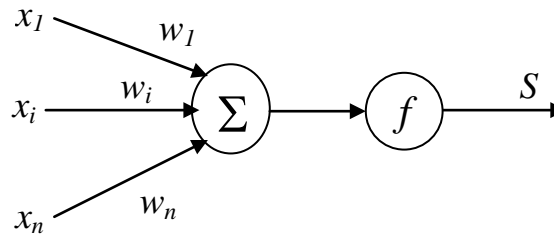


Fig. 1 - Structure d'un Nœud

Les entrées d'un neurone proviennent d'autres neurones en amont, via des liens pondérés appelés : liens synaptiques. La fonction d'activation calcule la sortie du neurone par :

$$S = f (W^T \cdot X) = f (\sum_{i=1}^n w_i \cdot x_i)$$

Les réseaux de neurones sont des assemblages de neurones. Chaque neurone accomplit un rôle relativement simple, mais le comportement collectif (ou émergent) peut être fort complexe.

Fonctions d'activation : Plusieurs types de fonctions d'activations ont été proposés dans la littérature, chacune pour un problème spécifique. Les fonctions les plus répandues sont schématisées sur la figure 2 :

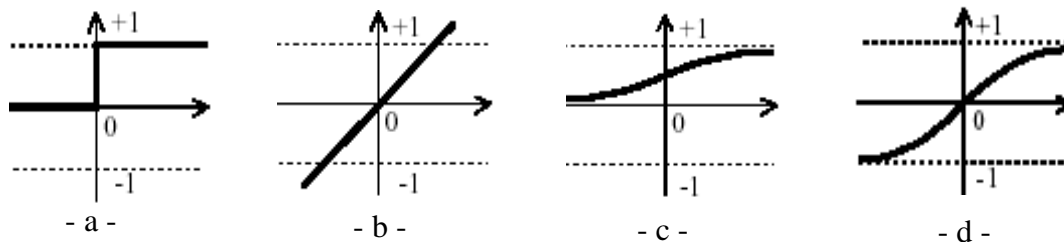


Fig. 2 - Différents types de fonctions d'activation

La fonction sigmoïde (-c- et -d-) est couramment utilisée car elle est monotone, bornée, non-linéaire et indéfiniment dérivable.

3. Architecture :

Un aspect important des réseaux de neurones est le mode dans lequel les unités sont connectées. Les réseaux de neurones peuvent se répartir en deux grandes classes, selon qu'ils interdisent ou qu'ils permettent des liens récurrents entre les neurones.

Réseaux à couches : tous ses neurones cachés et de sortie reçoivent leurs entrées des neurones de la couche précédente seulement. L'entrée est présentée au neurone, ensuite propagée en avant à travers le réseau. La sortie ne forme jamais une part de son entrée propre.

Les réseaux à couches sont des fonctions statiques. Ils ne possèdent pas un comportement dans le temps. Toutefois, ils peuvent représenter des systèmes dynamiques en adoptant une approche similaire à celle de la théorie linéaire des modèles de régression.

Les réseaux récurrents : Les réseaux récurrents possèdent au moins une boucle fermée, i.e., une connexion cyclique. Le comportement de tels réseaux peut être très complexe. Ces réseaux sont vus comme des systèmes dynamiques non linéaires.

Par ailleurs, les réseaux à couches sont dits *complètement connectés* si chaque nœud dans chaque couche est connecté avec tous les nœuds de la couche suivante. Si une connexion manque, le réseau est dit alors *partiellement connecté*. Les réseaux partiellement connectés peuvent être formés si des informations à priori sur le problème sont disponibles, indiquant l'utilisation d'une telle structure.

Réseaux simple couche, réseaux multicouches :

Un réseau est dit simple couche s'il ne possède pas d'autres couches que celles de l'entrée et de la sortie. Il est dit multicouches s'il possède au moins une couche cachée.

Ainsi, pour un réseau simple couche, la sortie s'écrit sous la forme :

$$y(x) = f\left(\sum_{i=1}^n w_i \cdot x_i\right)$$

Perceptron multicouches :

Les réseaux multicouches, appelés aussi Perceptrons (PMC) sont utilisés principalement pour l'approximation des fonctions ne possédant pas des variations locales significatives. La propriété d'approximation universelle est vérifiée. Quant à la dimension de l'ensemble d'apprentissage, des études ont montré qu'il doit être approximativement 10 fois le nombre des poids.

4. Apprentissage d'un réseau de neurones :

La règle d'apprentissage est la procédure de modification des poids d'un réseau de neurone dans le but de produire une sortie désirée. On peut distinguer deux types d'apprentissage :

Apprentissage supervisé : l'ensemble des entrées et les sorties réelles est utilisé. Le réseau utilise les entrées pour produire ses propres sorties. Ces dernières sont comparées avec les sorties désirées et les différences sont utilisées pour modifier les poids.

Apprentissage non-supervisé : le réseau développe ses règles de classification pour classifier les données de l'ensemble test qui est constitué d'entrées seulement. Le réseau doit être capable de découvrir les caractéristiques statistiques des entrées.

Règles d'apprentissage :

Plusieurs règles d'apprentissage existent pour les différents types de réseaux de neurones. La règle la plus utilisée, qui a été proposée en 1986, est *la rétropropagation de l'erreur*. L'idée de base est de quantifier l'erreur entre les sorties désirées et les sorties du réseau. L'erreur globale est ensuite propagée dans le réseau selon la contribution de chaque neurone dans l'erreur, déterminée à l'aide du gradient.

L'erreur est définie à chaque étape par :

$$E(k) = \frac{1}{2} \cdot (y_p(k) - y_m(k))^2,$$

où y_p est la sortie réelle, y_m est la sortie du réseau, k étant l'indice de la donnée de l'ensemble test.

L'objectif est de minimiser cette erreur en calculant le gradient par rapport au poids $w_{i,j}$.

Les poids sont alors adaptés par :

$$W(k) = W(k-1) + \theta \cdot \left(-\frac{dE(k)}{dW(k-1)} \right),$$

θ étant la vitesse d'apprentissage.

D'autres variantes de l'algorithme de rétropropagation ont été proposées. On en cite par exemple l'ajout d'un terme de régularisation à l'erreur. Toutefois, cette forme standard reste très populaire.

5. L'hybridation neuro-floue :

Les réseaux de neurones sont des structures de calcul efficaces, capables d'apprendre et de s'adapter par des exemples, les systèmes flous n'étant pas puissants dans l'apprentissage,

l'adaptation, et le calcul parallèle. Les réseaux de neurones offrent de bonnes performances dans ces aspects. Les réseaux de neurones manquent de flexibilité et la représentation symbolique des connaissances, alors que la logique floue en est puissante. Dans les dernières années, plusieurs recherches ont été faites pour réaliser des hybridations entre la logique floue et les réseaux de neurones, pour tirer profit des avantages de l'une et de l'autre.

Bibliographie :

- Jean-François Jodouin, Les réseaux neuromémitiques : modèles et applications, eds. Hermès, Paris 1994.
- G. W. Ng, Application of neural Networks to adaptive Control of Nonlinear Systems, Research Studies Press Ltd, 1997.

Annexe - B

Algorithmes Génétiques

" Tout ce qui existe dans l'univers est le fruit du hasard et de la nécessité. "

Démocrite.

Parmi les problèmes rencontrés par le chercheur et l'ingénieur, les problèmes d'optimisation occupent à notre époque une place de choix.

Les Algorithmes Génétiques sont des méthodes stochastiques d'optimisation basées sur la théorie de l'évolution de Darwin. Ils consistent à faire évoluer une population de solutions candidates à un problème donné à l'aide de différents opérateurs : sélection, croisement, mutation. Ils sont en particulier utilisés pour les problèmes d'optimisation comportant de multiples paramètres et plusieurs objectifs.

Procédure d'un algorithme génétique

Début :

$t \leftarrow 0;$

Initialiser $P(t)$,

% $P(t)$ représente la population de solutions candidates à l'instant t %

Evaluer $P(t)$,

Quand (condition terminale non satisfaite) **Faire :**

Début :

$t \leftarrow t + 1,$

Sélectionner $P(t)$ à partir de $P(t-1)$,

Reproduction de $P(t)$: Croisement et Mutation,

Evaluer $P(t)$,

Fin.

Fin.

Caractéristiques principales des AGs :

1. Les AGs. utilisent une représentation codée des solutions,
2. Les AGs. travaillent sur une population de points au lieu d'un point unique,
3. Les AGs. n'utilisent que les valeurs de la fonction à optimiser : pas celles de sa dérivée ni autre information auxiliaire,
4. Les AGs. peuvent être utilisés facilement dans des hybridations avec d'autres techniques,
5. Les AGs. sont robustes,
6. Les AGs. ne garantissent pas de trouver une solution optimale pour le problème donné mais plutôt une solution acceptable.

Mécanisme d'un AG simple :

- **Codage des variables :** La première étape est de définir et de coder convenablement le problème. A chaque variable d'optimisation x_i , nous faisons correspondre un *gène*. Nous appelons *chromosome* un ensemble de gènes. Chaque *individu* (solution candidate) est doté d'un chromosome.

Nous appelons *population* un ensemble de N individus que nous allons faire évoluer.

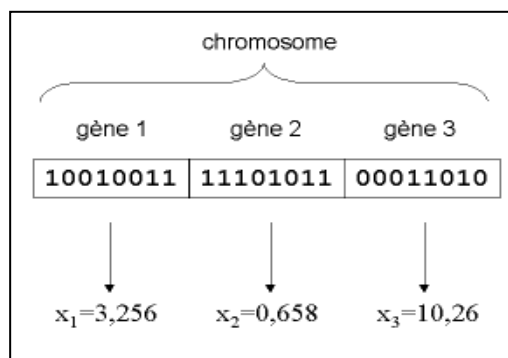


Fig. 1 - : Codage binaire des

La représentation des solutions par une suite de 0 et 1 est le plus souvent utilisée. Un des avantages du codage binaire est que l'on peut facilement coder toutes sortes d'objets : des réels, des entiers, des valeurs booléennes, des chaînes de caractères

- **Genèse de la population :** La seconde étape de l'algorithme est la genèse de la population, c'est-à-dire, le choix des dispositifs de départ que nous allons faire évoluer. Une initialisation aléatoire par exemple est simple à réaliser : les valeurs des gènes sont

tirées au hasard selon une distribution uniforme. Notons qu'on peut, si nécessaire, introduire des individus déjà calculés.

La taille N de cette population, résultera d'un compromis entre temps de calcul et qualité de la solution.

- **Evaluation** : L'évaluation de chaque dispositif est réalisée en utilisant une fonction d'adéquation représentant un critère donné : coût, objectif, La majeure partie du temps de calcul sera probablement due à l'exécution de cette étape.

- **Sélection – élimination** : Nous appelons *génération* la population à un instant donné t . Une fois réalisée l'évaluation de la génération, on opère une sélection à partir de la fonction d'adéquation. Seuls les individus passant l'épreuve de sélection (rappelons qu'il ne s'agit ici que de dispositifs !) peuvent accéder à la génération intermédiaire (mating pool en terminologie anglo-saxonne) et s'y reproduire. En fait, cette génération intermédiaire est deux fois plus petite ($N/2$ dispositifs) que la génération dont elle est issue (figure 2).

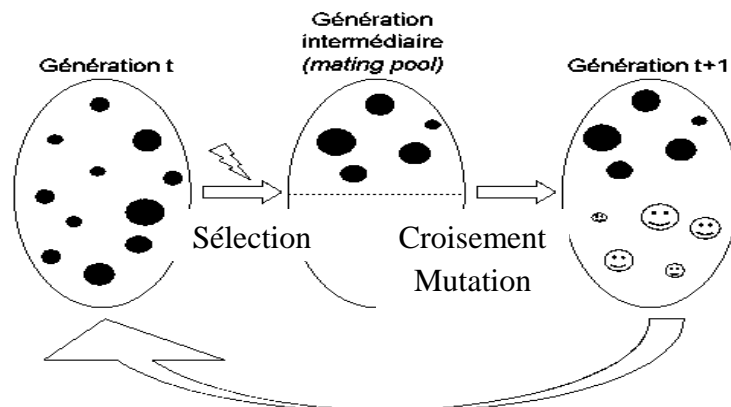


Fig. 2 – Algorithme de Sélection.

On peut citer deux techniques simples de sélection :

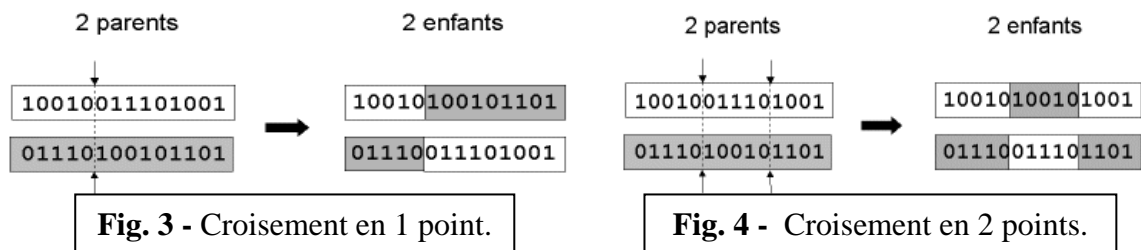
- " **$N/2$ -élitisme**" : les individus sont triés selon leur fonction d'adaptation. Seule la moitié supérieure de la population, correspondant aux meilleurs composants, est sélectionnée.
- "**Sélection par tournoi**" : Deux individus sont choisis au hasard et combattent (on compare leurs fonctions d'adaptation) pour accéder à la génération intermédiaire. Le plus adapté l'emporte. Cette étape est répétée jusqu'à ce que la génération intermédiaire soit remplie ($N/2$ composants).

- **Opérateur croisement** : Une fois la génération intermédiaire à moitié remplie, les individus sont aléatoirement répartis en couples. Les chromosomes des parents sont

alors copiés et recombines de façon à former deux descendants possédant des caractéristiques issues des deux parents. On forme ainsi la génération $t+1$ (Figure 2).

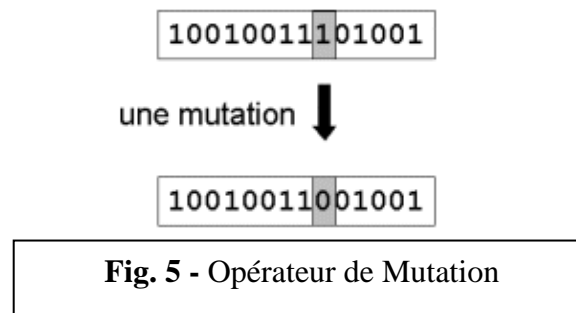
L'opérateur de croisement favorise ainsi l'exploration de l'espace de recherche. Cet opérateur permettra de combiner rapidement deux gènes A et B dans la descendance des deux parents pour produire des individus encore plus adaptés.

L'opérateur de croisement le plus classique est : " *le croisement en un point* ". On choisit au hasard un seul point de croisement (Figure 3).



D'autres formes de croisement existent : croisement en k points, ...

- **Opérateur mutation :** Nous définissons une mutation comme étant l'inversion d'un bit dans un chromosome (Figure 5). Cela revient à modifier aléatoirement la valeur d'un paramètre du dispositif. Les mutations jouent le rôle de bruit et empêchent l'évolution de se figer. Elles permettent d'assurer une recherche aussi bien globale que locale, selon le poids et le nombre des bits mutés. De plus, elles garantissent mathématiquement que l'optimum global peut être atteint.



Souvent la probabilité d'application de l'opérateur de mutation p_m par bit et par génération est fixée entre 0,001 et 0,01.

Remarques :

1. Cette présentation des algorithmes génétiques se limitant aux grandes lignes et aux formes les plus classiques, est destinée plus à ceux qui découvrent les algorithmes

génétiques pour la première fois. Différentes présentations ont été proposées dans la littérature (voir par exemple [MIC 94]).

2. L'efficacité d'un algorithme génétique dépend généralement des caractérisations suivantes :
 - i) La manière de représenter les solutions : Codage,
 - ii) La population initiale des solutions,
 - iii) La fonction d'adéquation ou encore d'évaluation,
 - iv) Les paramètres utilisés (taille de la population, probabilités d'application des opérateurs génétiques de croisement et de mutation, ...).

Références :

[MAG 01] Vincent MAGNIN, Cyber-cours : Optimisation et algorithmes génétiques, <http://www.univ-lille1.fr/~eudil>, 2001.

[MIC 94] Zbigniew Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Secon edition, Springer-Verlag, 1994.