UNE FORMALISATION DES PROPRIÉTÉS EN MODÉLISATION DÉCLARATIVE À L'AIDE DES ENSEMBLES FLOUS

Emmanuel DESMONTILS
IRIN
Faculté des Sciences et des Techniques
2 rue la Houssinière
44072 Nantes cedex 03
e-mail : desmontils@irin.univ-nantes.fr

Résumé
La notion de propriété est un élément central en modélisation déclarative. Peu de travaux se sont intéressés à la formalisation des propriétés. Nous proposons ici d'étudier une méthode basée sur la théorie des sous-ensembles flous pour les formaliser. Cette théorie permet de représenter et de modifier des propriétés simples ou paramétrées (qui portent sur une scène ou un objet) et des propriétés plus complexes (portant sur plusieurs objets ou sous-scènes). L'évaluation d'une description à partir de ce formalisme permet d'obtenir des formes plus variées. Cette méthode semble être bien adaptée à la modélisation déclarative.

Mots clés : Synthèse d'image, modélisation géométrique, modélisation déclarative, intelligence artificielle, propriétés imprécises, sous-ensembles flous, évaluation qualitative, modificateur, opérateur flou

Introduction

Un des objectifs principaux de la modélisation déclarative en synthèse d'image ([LMM89]) est d'obtenir une ou plusieurs scènes répondant à un ensemble de propriétés appelé description. Dans les modeleurs géométriques classiques, une « description » est donnée de façon précise et quantitative (série de coordonnées et de primitives). Pour les modeleurs déclaratifs, présentés dans [Des95a] et [LDe96], les descriptions peuvent être des concepts imprécis, vagues et qualitatifs. Cependant, les descriptions précises ne sont pas interdites.

L'objectif de ce travail est de présenter un formalisme inspiré de la Théorie des Sous-Ensembles Flous (voir [Zad65], [Tong95], [DPr93], [Mat93]…) pour gérer les propriétés utilisées dans un modeleur déclaratif. La théorie sert de base mais les modificateurs et les opérateurs flous sont interprétés et utilisés différemment. En particulier, nous proposons une utilisation systématique de modificateurs sur les sous-ensembles flous.

La formalisation des propriétés en modélisation déclarative a fait l'objet de plusieurs travaux effectués par [Col92], [Paj94], [Pou94], [Ple91] et surtout par [Chau94]. Pour eux, les propriétés sont définies sur des intervalles classiques et associent une mesure et une comparaison. [Ple91], [MaM91] et [Pou94] utilisent même des propriétés vérifiées simultanément (notion de <u>multi-description</u>) ou à un coefficient près (notion de <u>vérification "à peu près"</u>, de <u>règle de modification</u> ou de <u>déformation</u>). L'utilisation des sous-ensembles flous est donc le prolongement naturel des travaux réalisés en modélisation déclarative. Ces derniers introduisent aussi l'utilisation de modificateurs ("*très*", "*un peu*"…). Généralement, ils sont dépendants des propriétés utilisées (autant d'intervalles que de modificateurs applicables). Cependant, [Chau94] propose une méthode pour traiter les modificateurs de manière indépendante. Nous partirons de cette idée pour gérer les modificateurs d'ensembles flous, en ajoutant l'utilisation d'opérateurs flous (*"environ", "relativement"*…). Aucun des travaux en modélisation déclarative ne fait apparaître explicitement ce genre d'opérateur.

On note que [Dje91] a déjà employé les sous-ensembles flous pour représenter les propriétés d'une description. Cependant, il ne propose pas de formalisme précis (pas de notion de domaine ni d'étude des différentes propriétés) et les modificateurs sont dépendants des propriétés (pas de notion de modificateurs génériques) et les opérateurs flous n'existent pas.

Cet article se décompose en sept parties. Nous spécifierons d'abord l'environnement de définition des propriétés (§1-2). Ensuite, nous étudierons les propriétés simples et paramétrées qui portent sur une seule scène (§3). Puis, nous aborderons l'utilisation des opérateurs flous (§4), les propriétés relatives (§5) et la négation d'une propriété (§6). Enfin, nous aborderons une méthode pour évaluer une description (§7-8) et terminerons par un survol rapide de son utilisation dans les modeleurs déclaratifs (§9).

1. Domaine de description et mesure

Pour décrire un objet, il existe un certain nombre de propriétés ayant en commun une même caractéristique. Par exemple, si on dit *“le cube est grand”* ou *“le cube est petit”*, on fait référence à un concept commun aux propriétés *“grand”* et *“petit”*, à savoir la *“taille”*. Nous appellerons cela un domaine de description. Un domaine de description D est défini comme l’ensemble des valeurs que peut prendre une scène (ou une forme) du point de vue d’un concept. En théorie des sous-ensembles flous, un domaine de description est considéré comme un référentiel associé à une variable linguistique[1]. Généralement, D est borné (par la sémantique ou par contrainte d’implémentation) et sur plusieurs dimensions ($\mathbb{R}$, $\mathbb{C}$…). Cependant, en pratique, il est de dimension 1 (énumération d’éléments ou sous-ensemble de $\mathbb{B}$, $\mathbb{N}$, $\mathbb{Z}$ ou $\mathbb{R}$) voire 2 dans le cas de notions spécifiques.

La valeur que représente une scène donnée dans un domaine est déterminée à l’aide d’une fonction (quand elle existe) caractéristique de la notion. Nous appelons cette fonction une mesure du domaine. La mesure d’une forme pour un domaine D est une fonction m_D de l’univers des formes U_f dans D qui, à toute scène ou ensemble de sous-scènes, fait correspondre une valeur de D. Un domaine est dit d’ordre n (ou n-aire) si sa mesure porte sur n sous-scènes. Nous supposerons que tout domaine possède une mesure.

2. Propriétés d’un domaine

Un domaine peut se voir associer un certain nombre de propriétés de toutes sortes. Par exemple, le domaine “taille” est accompagné souvent des propriétés suivantes : *“grand”, “moyen”, “petit”, “de 1m80”, “entre 1m70 et 1m80”…*

Elles sont définies sur certaines valeurs du domaine. Cependant, il est souvent difficile de préciser les valeurs limites pour une propriété donnée. Par exemple, comment trouver les bornes d’un intervalle [a,b] pour lequel toutes les valeurs (tailles) sont considérées comme *“grandes”* ? Pourquoi les valeurs « a-ε » et « b+ε » (ε étant très petit) ne seraient-elles pas considérées comme *“grandes”* ? De plus, une même forme ne peut-elle pas commencer à être *“grande”* tout en étant encore *“moyenne”* ?

Au lieu d’être obligé de donner des limites précises, comme c’est le cas dans la plupart des modeleurs déclaratifs actuels, nous proposons de représenter une propriété d’un domaine D comme une sorte de sous-ensemble flou[2].

---

[1] Dans l’assertion « X est A », X est une variable linguistique associée à un référentiel (ou domaine) D et A est un sous-ensemble flou de ce domaine.

[2] Un sous-ensemble flou A sur le domaine (appelé aussi référentiel) T est défini par la donnée d’une fonction d’appartenance μ_a à valeur dans [0,1]. $\mu_a(t)$ est le degré d’appartenance de $t \in T$ à A.

Ainsi, les valeurs appartenant au noyau[3] de la propriété *"grand"* sont absolument grandes, celles appartenant au support[4] ne sont pas totalement grandes et les autres (hors du support) ne le sont pas du tout.

Plusieurs types de propriétés sont mis en évidence en fonction de l'ordre du domaine sur lequel elles sont définies :

- les propriétés simples qui portent sur toute la scène (ou une seule sous-scène),
- les propriétés relatives qui sont des relations binaires entre deux objets de la scène.

Bien sûr, il existe d'autres types de propriété comme les propriétés complexes ou les propriétés de comparaison (*"A est beaucoup plus grand que B"*). Pour avoir des informations sur le traitement de telles propriétés ainsi que des précisions sur ce qui est présenté dans cet article, on peut consulter [Des95b].

3. Propriétés simples

3.1. Définition

L'objectif est d'étudier des descriptions utilisant une propriété simple, c'est-à-dire de pouvoir traiter des expressions linguistiques (très souvent adverbiales) telles que : *"Le menhir est **assez** grand"*, *"L'immeuble est **très** grand"*, *"Le motif est **très très** petit"*... Nous voulons donc être capable d'interpréter des descriptions de la forme :

*"X est **Modificateur** Propriété_Simple"*.[5]

Un modificateur, ou opérateur de modification, porte sur la sémantique de la propriété. En effet, dire *"Le cube est **assez** grand"* et *"Le cube est **très** grand"* change l'idée de taille que l'on a du cube. Dans la seconde phrase, cet objet possède une taille plus importante et plus précise que dans la première. Les modificateurs sont assez nombreux. Donc, pour simplifier, nous choisirons : *"**extrêmement**"*, *"**très très**"*, *"**très**"*, *"**assez**"*, *"**normalement**"*, *"**assez peu**"*, *"**très peu**"*, *"**très très peu**"* et *"**extrêmement peu**"*. Il peut arriver que pour certaines propriétés, certains modificateurs soient autorisés et d'autres pas. L'opérateur *"**normalement**"* correspond à l'opérateur par défaut *"**ø**"* appelé aussi opérateur vide ou modificateur vide. Désormais, *"X est Propriété"* est équivalent à *"X est **ø** Propriété"* ou *"X est **normalement** Propriété"* (*"grand"* est équivalent à *"**ø** grand"* ou *"**normalement** grand"*). Ainsi, contrairement à la théorie des sous-ensembles flous, dans toute expression il y a un modificateur. De plus, les modificateurs (comme par exemple *"très"*) sont traités différemment.

---

[3] Le noyau d'une propriété est l'ensemble des valeurs du domaine pour lesquelles le degré d'appartenance à cette propriété est 1.
[4] Le support d'une propriété est l'ensemble des valeurs du domaine pour lesquelles le degré d'appartenance à cette propriété est non nul.
[5] On retient ici toutes les expressions pouvant se ramener à cette forme.

Une propriété simple P est un sous-ensemble flou d'un domaine D défini par la fonction d'appartenance μ_p de D dans [0,1] qui, à tout élément de D, fait correspondre $\mu_p(t)$. C'est une valeur possible d'une variable linguistique associée à D. L'évaluation d'une propriété simple P sur un domaine D associé au modificateur par défaut (par exemple *"A est P"*) consiste à appliquer :

$$\begin{array}{llll} U_f \rightarrow & D & \rightarrow & [0,1] \\ f & \rightarrow t = m_D(f) & \rightarrow & \mu_p(t) \end{array}$$

Parmi les propriétés, certaines n'acceptent que l'opérateur vide. En effet, on ne peut pas dire *"Le nombre de voxels de la matrice est **très** premier"*. Par contre, on peut dire *"Le nombre de voxels de la matrice est **très** important"*. Les propriétés qui acceptent d'autres modificateurs que le modificateur vide s'appellent des propriétés modifiables. Nous posons le postulat suivant : une propriété modifiable est une propriété définie à l'aide d'un intervalle flou[6].

3.2. La fonction d'appartenance d'une propriété simple

Le choix de la fonction d'appartenance est a priori libre. Il en existe un grand nombre dans la littérature sur les ensembles flous ([Zad65], [Ton95]...). Pour faciliter les traitements, nous considérerons que les propriétés modifiables sont définies à l'aide de cas particuliers de fonctions d'appartenance L-R. L'expression d'une fonction d'appartenance L-R d'une propriété P d'un domaine D définie par le quadruplet $<\alpha_p,a_p,b_p,\beta_p>$ (α_p, $\beta_p \geq 0$) et les fonctions $L_p(x)$ et $R_p(x)$ est :

$$\mu_{<\alpha_p,a_p,b_p,\beta_p>, L_p, R_p} : \begin{array}{lll} D \rightarrow & [0,1] & \\ t \rightarrow & 0 & \text{si } t < a_p - \alpha_p \text{ ou } t > b_p + \beta_p \\ & L_p((a_p - t) / \alpha_p) & \text{si } (\alpha_p \neq 0) \text{ et } (a_p - \alpha_p \leq t < a_p) \\ & 1 & \text{si } a_p \leq t \leq b_p \\ & R_p((t - b_p) / \beta_p) & \text{si } (\beta_p \neq 0) \text{ et } (b_p < t \leq b_p + \beta_p) \end{array}$$

Ces fonctions L-R permettent de déterminer le noyau, le support et le type de la fonction (nombre classique, intervalle classique, nombre flou, intervalle flou...) très facilement. De plus, elles sont faciles à manipuler, simples à calculer et les fonctions pour modifier leur forme sont faciles à mettre en place. Elles permettent de faire des approximations satisfaisantes de fonctions complexes. C'est pourquoi elles sont très souvent utilisées en théorie des sous-ensembles flous. C'est pour cela que, plus généralement, dès que l'on aura à utiliser des propriétés définies sur des intervalles flous, leur fonction d'appartenance sera une fonction L-R.

---

[6] Un intervalle flou est une quantité floue convexe supposée semi-continue supérieurement. C'est une généralisation de l'intervalle classique. Une quantité floue est un ensemble flou, de noyau non vide, dans l'univers des nombres réels.

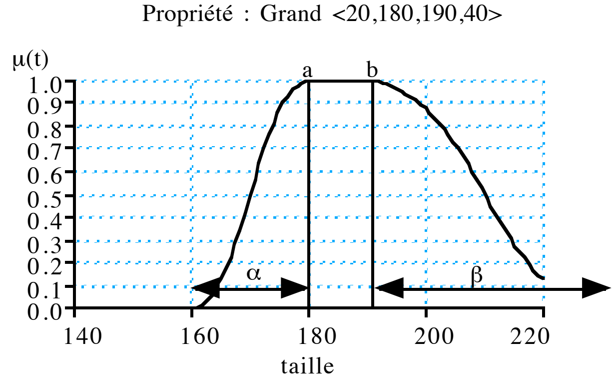


Figure 1 : Une fonction d'appartenance possible pour la propriété « ø grand »

Une propriété P d'un domaine D définie par une fonction L-R sera donc définie par : P = {D, <α_p,a_p,b_p,β_p>, $L_p(x)$, $R_p(x)$}. Si la fonction n'est pas L-R, nous aurons : P = {D, μ_p}. La figure 1 nous donne un exemple de propriété modifiable définie par {*"taille"*, <20,180,190,40>, L(x)=[2Max(0,1-x^2) si x>0.5, Max(0,1-2x^2) sinon], R(x)=L(x)}

3.3. Modificateurs génériques

L'objectif est de déterminer des fonctions génériques, associées aux modificateurs. Au lieu de construire autant d'intervalles que de modificateurs possibles, le concepteur se contente de définir la propriété principale. Il suffit ensuite d'appliquer le modificateur générique requis. Ainsi, pour la propriété *"Grand"*, nous avons : *"**Extrêmement**(Grand)"*, *"**Très_peu**(Grand)"*, *"**ø**(Grand)"*… La fonction générique associée au modificateur s'applique sur la fonction d'appartenance de la propriété pour obtenir la fonction d'appartenance de la propriété modifiée. La fonction générique translate et contracte la fonction de la propriété proportionnellement au modificateur qu'elle représente.

3.4. Coefficient de translation élémentaire

Il faut faire attention à la direction de la translation. En effet, *"**Très**"* ne va pas agir de la même manière sur la propriété *"Grand"* et sur *"Petit"*. Sur *"Grand"* (resp. *"petit"*), la fonction d'appartenance est translatée vers les tailles plus importantes (resp. plus faibles). Il est donc nécessaire d'associer à chaque propriété un signe indiquant la direction de modification. Le signe sera positif (resp. négatif) pour les propriétés qui seront translatées vers des valeurs plus importantes (resp. plus faibles) par l'application du modificateur *"très"*. De plus, il est clair que l'importance de la translation sera fonction de l'opérateur de modification mais aussi de la sémantique de la propriété. Nous associerons donc à chaque propriété P un réel τ_p que nous appellerons <u>coefficient de translation élémentaire</u>.

Nous aurons alors trois cas :

- $\tau_p > 0$ pour les propriétés positives (*"Grand"*),
- $\tau_p < 0$ pour les propriétés négatives (*"Petit"*),
- $\tau_p = 0$ pour celles qui ne seront pas modifiées ou propriétés neutres (*"Moyen"*).

Désormais, une propriété P d'un domaine D sera définie par $\{D, \mu_p, \tau_p\}$ ou $\{D, <\alpha_p, a_p, b_p, \beta_p>, L_p(x), R_p(x), \tau_p\}$ si la fonction d'appartenance est L-R. Si elle est neutre ou non modifiable, nous poserons $\tau_p = 0$. Nous supposerons dans la suite de ce travail que les propriétés modifiées sont soit positives soit négatives.

3.5. Coefficient de modification

Comme pour les propriétés, il existe trois catégories de modificateurs. En effet, *"**très**"* et *"**assez peu**"* ne vont pas agir de la même manière sur la propriété *"grand"*. *"**très**"* (resp. *"**assez peu**"*) va translater la propriété *"grand"* vers des tailles plus importantes (resp. plus faibles). Il est donc nécessaire d'associer à chaque modificateur un signe indiquant la direction de modification. Le signe sera positif (resp. négatif) pour les modificateurs qui déplacerons la fonction d'appartenance d'une propriété positive (par exemple *"grand"*) vers des valeurs plus importantes (resp. plus faibles). De plus, il est clair que l'importance de la translation sera fonction de l'opérateur de modification (*"**très**"* produit une modification plus importante que *"**assez**"*). Nous associerons donc à chaque modificateur un réel k_{modif} appelé coefficient de modification. Nous avons alors trois cas :

- $k_{modif} > 0$ pour les modificateurs positifs (*"**très**"*),
- $k_{modif} < 0$ pour les modificateurs négatifs (*"**assez_peu**"*),
- $k_{modif} = 0$ pour les modificateurs neutres (*"**normalement**"*).

Les modificateurs n'ont pas d'effet sur les propriétés neutres ($\tau_p = 0$). Nous avons alors *"**très** moyen"* équivalent à *"**ø** moyen"*. De même, les modificateurs neutres ($k_{modif} = 0$) n'ont pas d'effet sur les propriétés positives ou négatives.

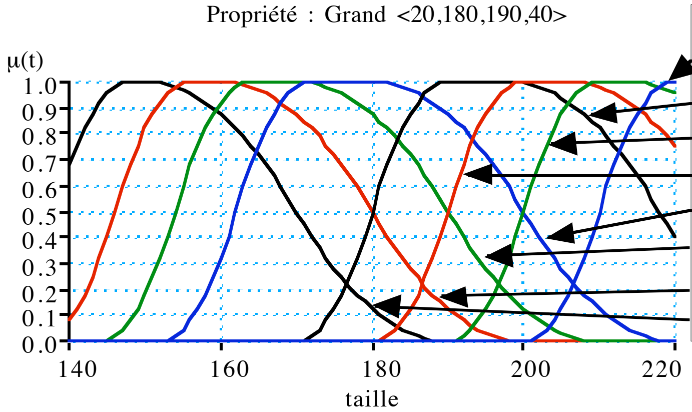
3.6. Application d'un modificateur

Avec une propriété $P=\{D, <\alpha_p, a_p, b_p, \beta_p>, L_p(x), R_p(x), \tau_p\}$ et un modificateur *Modif* de coefficient de modification k_{modif}, on a :

Modif P = {D,

$<\alpha_p, a_p + k_{modif}*\tau_p + | k_{modif}*(b_p - a_p)*10\% |, b_p + k_{modif}*\tau_p - | k_{modif}*(b_p - a_p)*10\% |, \beta_p>,$

$L_p(x), R_p(x), \tau_p\}$



Modificateur	k
Extrêmement	4
Très_Très	3
Très	2
Assez	1
Assez_Peu	-1
Très peu	-2
Très_Très_Peu	-3
Extrêment_Peu	-4

Figure 2 : Modificateurs sur la fonction d'appartenance de la propriété "ø grand"

La translation de la fonction ("$k_{modif}*\tau_p$") est proportionnelle au coefficient de translation élémentaire de P (τ_p) et au coefficient propre au modificateur (k_{modif}). Le coefficient "$\pm \mid k_{modif}*(b_p-a_p)*10^{\%} \mid$" réduit le noyau de la propriété pour traduire la contraction. La réduction est fonction du modificateur et proportionnelle à la taille du noyau.

3.7. Composition de modificateurs

L'application d'un modificateur sur une propriété modifiable donne à nouveau une propriété modifiable (la fonction préserve l'intervalle flou). Par conséquent, il est possible d'employer un autre modificateur. Nous avons donc :

"*X est **Modif**$_\beta$ (**Modif**$_\alpha$ Propriété)*".

En recommençant ce raisonnement autant de fois que l'on veut, on construit des descriptions comme : "L'immeuble est **très très très** grand", "La scène est **très très très peu** remplie"… En pratique, les possibilités d'application successives de modificateurs sont assez restreintes. Nous pouvons, en fait, mettre en évidence les trois règles suivantes :

- "***très***" est le seul opérateur pouvant être répété plusieurs fois dans une même expression. Il porte sur une propriété dont le dernier modificateur est "***très***", "***très très***", "***très très peu***" ou "***très peu***".
- On peut appliquer n'importe quel modificateur sur la propriété "***ø** Propriété*".
- "***très***" est le seul modificateur s'appliquant à une propriété autre que "***ø** Propriété*".

3.8. Influence des modificateurs sur la fonction d'appartenance

Lorsqu'on applique un modificateur sur une fonction d'appartenance d'une propriété modifiable, il change la forme de la fonction par une translation et une contraction. Si la fonction de départ est "collée" à une borne du domaine et que la translation se fait vers l'autre borne, la forme de la fonction obtenue n'est pas triviale (voir la figure 3). Plusieurs politiques sont envisageables. Cependant, le plus simple est de définir la fonction même en dehors du domaine. Elle est alors utilisée comme les autres, il n'y a pas de traitement spécifique.

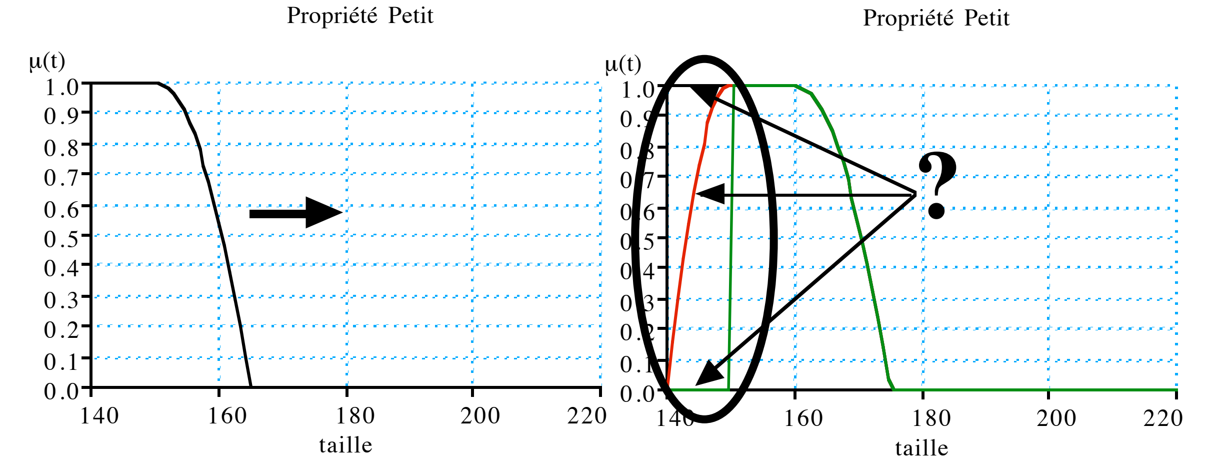


Figure 3 : Modification d'une propriété définie par rapport à une borne du domaine

3.9. Propriétés paramétrées

Les propriétés paramétrées permettent de construire des descriptions comme *"A est **entre** 1m70 **et** 1m90"*, *"A est **proche de** 1m80"*, *"A est **supérieur à** 1m80"* ou *"A est **de** 1m70"*.Ce sont des cas particuliers de propriétés simples. Une <u>propriété paramétrée</u> d'un domaine D par une valeur v (élément de D ou D^2) est une propriété simple modifiable (définie sur un intervalle flou). La fonction d'appartenance est paramétrée par la valeur v. Il en existe un certain nombre mais les plus courantes sont les propriétés neutres *"**De** XXX"* et *"**Entre** XXX **et** YYY"*. Malgré le paramètre (une valeur précise ou un intervalle classique), nous supposerons les propriétés paramétrées légèrement floues (l'intervalle support étant légèrement plus grand que l'intervalle noyau). En utilisant des propriétés L-R, les paramètres sont directement portés sur le quadruplet définissant la fonction d'appartenance. Par exemple, on a : *De 1m70* = {"Taille", <5,**170**,**170**,5>, $L_{de}(x)$, $R_{de}(x)$, 0}.

4. Opérateurs flous sur une propriété simple

4.1. Définition

L'objectif est d'étudier les effets d'une autre catégorie de modificateurs : les opérateurs flous. Ils s'appliquent sur une propriété simple ou paramétrée. En effet, on veut pouvoir traiter des assertions comme : *"Les menhirs sont **plus ou moins** espacés"*, *"La boîte est **à peu près** centrée"* , *"Le cube est **environ** de 2m"*, *"L'immeuble est **relativement** grand"*, *"La voiture est **vraiment** très longue"*…

Nous voulons être capable d'interpréter des descriptions de la forme :

*"X est **Opérateur_flou** Modificateur Propriété"*.

L'opérateur à appliquer dépend de la propriété. En effet, on peut dire *"A est environ de 2m"* mais on ne dit pas *"A est relativement de 2m"*. Plus généralement, Il semble que ce soit les propriétés paramétrées neutres qui demandent des opérateurs particuliers. De plus, les opérateurs flous sont nombreux et souvent synonymes. Pour simplifier, nous choisirons :

- pour les propriétés paramétrées neutres (*"de 2m"*) : *"**exactement**"*, *"**environ**"* et *"**vaguement**"*.
- pour les autres propriétés simples (*"Grand"*) : *"**vraiment**"*, *"**plus ou moins**"* et *"**relativement**"*.

Ces opérateurs ne transforment pas la propriété de la même manière que les modificateurs étudiés au §3. Les opérateurs flous ne déplacent pas la propriété (en particulier, le noyau n'est pas modifié). Ils augmentent la précision ou l'imprécision par rapport à la propriété d'origine (voir la figure 4). Ils contractent ou dilatent la fonction d'appartenance (l'intervalle support diminue ou grandit autour du noyau). Nous posons le postulat suivant : une <u>propriété acceptant un opérateur flou est une propriété simple ou paramétrée définie à l'aide d'un intervalle flou.</u>

4.2. Opérateurs génériques

L'objectif est de déterminer des fonctions génériques, associées aux opérateurs flous. Le concepteur se contente de définir la propriété principale (éventuellement modifiée). Il suffit ensuite d'appliquer l'opérateur générique requis. Ainsi, pour la propriété *"2m"*, nous avons : *"**Exactement**(2m)"*, *"**Environ**(2m)"*… La fonction générique associée à l'opérateur flou s'applique sur la fonction d'appartenance de la propriété pour obtenir la fonction d'appartenance de la propriété transformée. Les opérateurs ne dépendent pas des caractéristiques sémantiques de la propriété sur laquelle ils portent. En particulier, l'opérateur flou ne dépend pas du signe de la propriété. Par exemple, *"**relativement**"* agit de la même manière sur *"petit"* et sur *"grand"*. Il agit en modifiant la taille du support et la forme des fonctions L et R. Ces modifications dépendent directement de l'opérateur utilisé. Nous définirons donc chaque opérateur en nous inspirant des opérateurs de dilatation et de contraction[7]. Les opérateurs augmentant (resp. diminuant) le flou vont élargir (resp. réduire) le support de la propriété et modifier les fonctions L et R en pour augmenter (resp. réduire) les degrés d'appartenance (voir la figure 4).

---

[7] En T.S.E.F. <u>l'opérateur de contraction</u> est « très » et est traduit par une élévation au carré. <u>L'opérateur de dilatation</u> est « plus ou moins » et est traduit par l'application de la racine carrée.

Nous leur associerons un coefficient k_{flou} défini dans $\mathbb{R}^+$ rendant compte de leur « force » en posant :

- $k_{flou} > 1$ pour ceux diminuant le flou(*“**exactement**”*, *“**vraiment**”*…),
- $0 < k_{flou} < 1$ pour ceux augmentant le flou (*“**plus ou moins**”*, *“**environ**”*…).

Avec une propriété $P=\{D, \langle\alpha_p, a_p, b_p, \beta_p\rangle, L_p(x), R_p(x), \tau_p\}$ et un opérateur flou *“Flou”* de coefficient k_{flou}, on a : **Flou P**$=\{D, \langle\alpha_p/k_{flou}, a_p, b_p, \beta_p/k_{flou}\rangle, L_p(x)^{k_{flou}}, R_p(x)^{k_{flou}}, \tau_p\}$

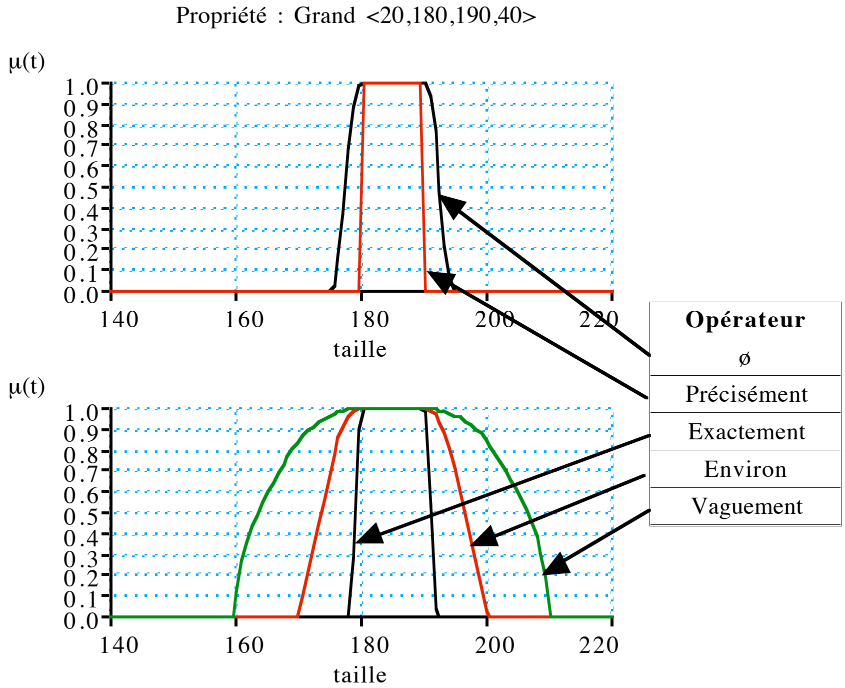


Figure 4 : Opérateurs flous sur la fonction d'appartenance pour la propriété "entre 1m80 et 1m90"

Dans certains cas, *“très”* apparaît pour renforcer un opérateur flou (*“très **exactement**…”*). Cependant, nous considérerons qu’il est soit utilisé par effet de style et il n’est alors pas nécessaire de le prendre en compte, soit il renforce la précision et nous traiterons alors l’opérateur flou *“**très_exactement**”* au même titre que les autres opérateurs flous.

4.3. Rendre non floue une propriété floue

L’objectif est de pouvoir traiter un opérateur de suppression du flou d’une propriété paramétrée (par exemple l’opérateur *“**précisément**”*) pour obtenir une propriété classique (nombre ou intervalle classique). L’opération consistant à rendre non floue une propriété floue est assez facile à mettre en place. Elle consiste à poser un seuil au-dessus duquel l’élément du domaine est considéré comme appartenant totalement à la propriété. Il suffit d’appliquer la fonction N_λ de [0,1] dans [0,1] ($\lambda \in [0,1]$) telle que $N_\lambda(t) = 0$ si $t < \lambda$, 1 sinon.

Remarque : si λ=1, la propriété modifiée est définie sur le noyau de la propriété d'origine et si λ=0, elle est définie sur le support. Déterminer une bonne valeur de λ n'est pas forcément évident.

5. Propriétés relatives

L'interprétation de descriptions du style *"A est **loin de** B"* ou *"A est très **proche de** B"* est un peu différente. En effet, jusqu'ici, les descriptions portaient sur une seule scène (ou sous-scène). La scène devait satisfaire la propriété (modifiée ou non). Désormais, la description porte sur deux sous-scènes de la scène courante. Celles-ci sont "liées" par une propriété éventuellement modifiée. Quand on dit *"A est très **loin de** B"*, l'objet *"A"* est en relation avec l'objet *"B"* par la propriété *"**loin de**"* modifiée par le modificateur *"très"*. Nous avons des propriétés (*"**proche de**"* et *"**loin de**"*) qui sont des relations binaires entre deux sous-scènes. Nous les appellerons des <u>propriétés relatives</u>. Elles ont la forme :

*"X est Modificateur **Propriété_Relative** Y"*.

Les modificateurs et les opérateurs flous sont identiques à ceux des propriétés simples et sont gérés de la même manière. Ces propriétés sont définies sur un domaine binaire (par exemple, le domaine des distances entre deux objets). Cependant, elles sont traitées comme des propriétés simples. Pour une propriété P={D, μ_p, τ_p}, nous avons alors :

$$\begin{array}{lll} U_f \times U_f & \rightarrow D & \rightarrow [0,1] \\ f_1, f_2 & \rightarrow t=m_D(f_1, f_2) & \rightarrow \mu_p(t) \end{array}$$

*"**loin de**"* et *"**proche de**"* sont donc des propriétés simples, c'est-à-dire définies sur le domaine binaire des *"distances"*. Et nous avons pour l'assertion *"A est **loin de** B"* :

$$\begin{array}{lll} U_f \times U_f & \rightarrow D_{distances} & \rightarrow [0,1] \\ A, B & \rightarrow t=m_{distance}(A, B) & \rightarrow \mu_{loin\_de}(t) \end{array}$$

6. Négation d'une propriété

Dans certaines descriptions, plutôt que de donner une série de propriétés d'un même domaine, il est plus facile d'indiquer la négation. Par exemple, on peut dire : *"L'immeuble n'est pas grand"*, *"Le cube n'est pas très loin du cône"*… Nous aurons donc des descriptions de la forme :

*"X **n'**est **pas** …"*.

Nous traitons *"X n'est pas …"* comme *"X est non …"* en ce qui concerne l'interprétation de la description[8]. Lorsqu'une propriété est modifiable, on lui associe un réel τ_p. La négation de cette propriété entraîne une inversion de sa direction (par exemple : *"non grand"* est de même direction que *"petit"*). Donc, la négation d'une propriété simple $P=\{D_p, \mu_p, \tau_p\}$ sera $P'=\neg P=\{D_p, 1-\mu_p, -\tau_p\}$. La négation d'un intervalle flou ne donne pas un intervalle flou dans le cas général. Or un opérateur de modification ne s'applique que sur des intervalles flous. Nous posons le postulat suivant : l'opérateur de négation s'applique toujours après les modificateurs et les opérateurs flous.

7. Evaluation d'une description

Une description peut être considérée comme une disjonction de conjonctions de propriétés (modifiées ou non). Nous allons donc étudier comment interpréter une conjonction, une disjonction et une description.

7.1. Conjonction de propriétés

Une scène doit satisfaire à plusieurs propriétés qui sont demandées simultanément. Nous avons alors une conjonction de propriétés. Par exemple, on peut dire : *"La maison est grande, assez éloignée **et** très allongée"*. Ce sont des descriptions de la forme :

*"P_1 **et** P_2 **et** P_3 ..."*.

Pour interpréter de telles descriptions, nous utilisons l'extension du produit cartésien à *n* propriétés (intersection de *n* ensembles flous). Avec T une t-norme adaptée[9] (par exemple, l'opérateur "min"), nous avons :

$$D_1 \times D_2 \times \ldots \times D_n \rightarrow [0,1]$$
$$t_1, t_2, \ldots, t_n \rightarrow \mu(t_1, t_2, \ldots, t_n) = T_{i \in [1,n]}(\mu_{P_i}(t_i))$$

Remarque : il serait intéressant d'étudier la possibilité d'utiliser une moyenne (pondérée ou non)[10] à la place de la t-norme.

7.2. Disjonction de propriétés

Dans certains cas, l'utilisateur peut laisser le choix entre plusieurs descriptions. Nous avons alors une disjonction de propriétés. Par exemple, nous avons : *"La maison est haute **ou** allongée"*. Ce sont des descriptions de la forme :

*"P_1 **ou** P_2 **ou** P_3 ..."*.

---

[8] Cette interprétation est celle de la théorie des sous-ensembles flous. En linguistique, il faudrait traduire par une disjonction de toutes les propriétés sauf celle sur laquelle on applique l'opérateur.
[9] Une norme triangulaire (t-norme) est une fonction T de [0,1]x[0,1] dans [0,1] commutative, associative, monotone et dont l'élément neutre est 1. N'importe quelle t-norme est un opérateur d'intersection.
[10] La notion de moyenne est intermédiaire entre t-norme et t-conorme (voir [DPr93]).

Pour interpréter de telles descriptions, nous utilisons l'extension du produit cartésien à n propriétés (union de n ensembles flous). Avec $\perp$ une t-conorme adaptée[11] ("max" par exemple), nous avons alors :

$$D_1 \times D_2 \times \ldots \times D_n \rightarrow [0,1]$$
$$t_1, t_2, \ldots, t_n \rightarrow \mu(t_1, t_2, \ldots, t_n)=\perp_{i \in [1,n]}(\mu_{P_i}(t_i))$$

7.3. Description

Globalement, une description a la forme :

"*(P_{11}* ***et*** *P_{12}* ***et*** *P_{13}...)* ***ou*** *(P_{21}* ***et*** *P_{22}* ***et*** *P_{23}...)* ***ou*** *(P_{31}* ***et*** *P_{32}* ***et*** *P_{33}...) ...*".

L'interprétation d'une description sur une forme donnée consiste donc à appliquer :

$$D_{11} \times D_{12} \times \ldots \times D_{1n} \times D_{21} \times D_{22} \times \ldots \times D_{2n} \times \ldots \rightarrow [0,1]$$
$$t_{11}, t_{12}, \ldots, t_{1n}, t_{21}, t_{22}, \ldots, t_{2n}, \ldots \rightarrow \mu(t,\ldots)=\perp_{i \in [1,m]}(T_{j \in [1,n]}(\mu_{P_{ij}}(t_{ij})))$$

8. Validité d'une forme par rapport à une description

Une forme F est dite <u>forme solution</u> si <u>toutes les propriétés</u> d'une des conjonctions de la description sont considérées comme <u>vérifiées</u>. Pour simplifier, nous considérerons dans ce paragraphe, une description comme une conjonction de propriétés.

Lors d'une étude de forme, il est indispensable de déterminer si une propriété donnée est ou non vérifiée. Pour cela, nous définissons le <u>Seuil d'Acceptation</u>. C'est un degré d'appartenance à partir duquel la propriété est considérée comme vérifiée. Autrement dit, soit un seuil d'acceptation $\mathbb{S}$ dans]0,1] (si $\mathbb{S}$=0, la propriété est inutile ; si $\mathbb{S}$=1, la propriété est classique), une propriété P={D, μ, τ} est vérifiée si $\mu \geq \mathbb{S}$. L'ensemble des mesures de D solutions par rapport au seuil d'acceptation est donc une α-coupe[12] ($\alpha=\mathbb{S}$) en en théorie des sous-ensembles flous. Donc, pour une propriété donnée P d'un domaine D, une forme est solution si la mesure t selon D appartient à la $\mathbb{S}$-coupe de P.

Le seuil d'acceptation est soit calculé automatiquement, soit déterminé par le concepteur ou par l'utilisateur. Plusieurs politiques sont applicables quant à la détermination de ce seuil d'acceptation suivant l'exigence portée sur la notion d'appartenance. Le choix d'une politique peut être global (pour toutes les propriétés) ou spécifique à chaque propriété. Un seuil d'acceptation pour la description ne semble pas nécessaire car il peut être déduit des seuils

[11] Une conorme triangulaire (t-conorme) est une fonction $\perp$ de [0,1]x[0,1] dans [0,1] commutative, associative, monotone et dont l'élément neutre est 0. N'importe quelle t-conorme est un opérateur d'union.

[12] Une α-coupe (ou coupe de niveau α) de l'ensemble flou A d'un domaine D pour une valeur donnée de $\alpha \in [0,1]$ est l'ensemble des valeurs de D telles que leur degré d'appartenance à A soit supérieur ou égal à α.

d'acceptation des propriétés. Il se détermine alors en fonction des conjonctions, des disjonctions, des t-normes et des t-conormes utilisées. Cependant, un seuil d'acceptation global permet de rendre compte d'une certaine incertitude quant à la description. Par exemple, nous pouvons avoir des débuts de description comme : *"La description est exactement la suivante : …"* ou *"La description est à peu près la suivante : …"*…

9. Modeleurs déclaratifs et propriétés floues

Un modeleur déclaratif doit mettre à la disposition de l'utilisateur des outils de description de la scène, de génération et de prise de connaissance des solutions (voir [LMM89], [Des95a] et [LDe96]). Le formalisme que nous venons d'exposer s'intégre convenablement dans ces outils. Il semble bien adapté aux techniques développées dans les modeleurs déclaratifs et apporte des méthodes nouvelles. Nous allons parcourir rapidement ces outils pour voir, a priori, les apports potentiels.

9.1. La phase de description

Les propriétés floues permettent de gérer plus finement la sémantique. La "forme" de l'intervalle sera fonction de la description. Le modèle de description interne est alors beaucoup plus fidèle. Il y a moins de pertes d'information. Cette représentation rend compte du flou "naturel" des termes utilisés dans la description. Classiquement, une propriété est donnée par rapport à un "idéal" et plus on s'éloigne de cet idéal, moins la propriété est vérifiée. Tout cela se fait progressivement.

De plus, la gestion de la cohérence de la description est plus efficace. Les méthodes de détection de l'incohérence d'une description, proposées principalement par [Don93] (technique basée sur la logique de Allen appliquée aux intervalles) et [Chau94], sont toujours valables (quitte à transformer provisoirement l'intervalle flou en intervalle classique à l'aide, si besoin, d'un seuil d'acceptation). Les méthodes de traitement de l'incohérence sont aussi valables (de la même façon). En particulier, si l'on choisit la méthode de compromis (par exemple celle proposée par [Chau94]), cette théorie affine le traitement. Par exemple, au lieu de modifier les intervalles directement (perte partielle de sémantique), on va modifier les propriétés en cause dans la description en ajoutant des opérateurs flous. Ainsi, la modification pourra être plus compréhensible par l'utilisateur et la sémantique de la description sera mieux conservée.

9.2. La phase de génération

Il n'existe pas de techniques de génération spécifiques au formalisme que nous proposons. Par contre, les techniques classiques de la modélisation déclarative sont utilisables :

- Les méthodes de tirage aléatoires sous contraintes ([Chau94]) sont particulièrement intéressantes. Il semble qu'elles s'adaptent parfaitement à l'utilisation des propriétés floues. Ce formalisme permet, en particulier, de préserver mieux la sémantique de la description tout au long de la génération d'une solution.
- Les méthodes de génération à l'aide d'arbres d'exploration peuvent utiliser très facilement les propriétés floues. En particulier, les vérifications a posteriori se font en appliquant directement la méthode d'évaluation d'une description exposée au §7. Les méthodes d'optimisation (élagages de l'arbre et réduction des contrôles) sont applicables sous réserve de certains aménagements (en particulier, il faut faire attention aux techniques utilisant la négation).
- Les méthodes de génération par arbres de déduction, comme pour les arbres d'exploration, utilisent parfaitement les méthodes de vérification a posteriori. De plus, il serait intéressant d'étudier les techniques de logique floue pour voir ce qu'elles pourraient apporter.
- Les techniques de description par modifications successives sont certainement plus délicates à utiliser avec les sous-ensembles flous. Une idée serait de transformer la fonction d'appartenance en fonction de probabilité. Il suffit ensuite d'opérer un tirage aléatoire sous contraintes.

9.3. La phase de prise de connaissance des solutions

Les techniques classiques de prise de connaissance des solutions sont toujours applicables et peuvent être affinées (en particulier pour le choix d'un bon point de vue). De plus, de nouvelles techniques sont applicables. Elles sont basées sur l'étude de la scène à partir de l'ensemble de propriétés possibles. Ainsi, on peut entrevoir trois axes possibles : la reformulation de la description à partir des propriétés les mieux vérifiées, l'utilisation de l'appartenance globale à la propriété et l'utilisation de techniques d'apprentissage et de classification spécifiques aux ensembles flous.

Conclusion

Nous avons vu que la théorie des sous-ensembles flous propose une base intéressante pour définir les propriétés en modélisation déclarative. Une propriété est définie à l'aide d'un sous-ensemble flou d'un domaine représentant un concept. Celui-ci évalue la forme à l'aide d'une mesure. Une fonction d'appartenance indique le degré d'appartenance de cette mesure à la propriété. Il existe plusieurs types de propriétés. Nous avons étudié les propriétés simples, paramétrées ou non, portant sur une scène ainsi que l'application des modificateurs et des opérateurs flous. Nous avons vu aussi le cas de propriétés relatives portant sur deux sous-scènes. Ensuite, nous avons abordé la gestion de la négation d'une propriété quelconque. Une fois définies ces différentes propriétés, nous nous sommes intéressés à l'interprétation d'une description complète composée d'une disjonction de conjonctions de propriétés et à l'utilisation de cette description pour vérifier la validité d'une forme. Enfin, nous avons parcouru rapidement les principaux outils nécessaires dans un modeleur déclaratif pour analyser, a priori, les apports de ce nouveau formalisme.

Cette formalisation semble bien adaptée pour définir les propriétés. Il faut noter que l'intérêt de cette technique ne réside sans doute pas dans la génération des solutions mais plutôt dans une souplesse accrue de la description et dans des possibilités nouvelles d'optimisation, de déduction et d'apprentissage. Cependant, il serait bon d'étudier précisément la généralisation des propriétés à des propriétés complexes (portant sur plusieurs sous-scènes par exemple) ainsi que le traitement des propriétés de comparaison et de modification (étude commencée dans [Des95b]). Il faut aussi préciser l'étude des propriétés neutres et de la négation. Il reste encore à intégrer ce formalisme dans un modeleur déclaratif et à étudier les nouvelles possibilités qu'il ouvre comme, notamment :

- son utilisation dans une plate-forme de programmation pour faciliter la définition des propriétés,
- l'étude de l'application de la logique floue pour mettre au point un moteur d'inférence permettant de déduire des propriétés plus efficacement,
- l'étude approfondie de nouvelles techniques de gestion de l'incohérence,
- l'étude de techniques d'apprentissage afin d'améliorer la souplesse des applications, la génération et la prise de connaissance des solutions (en utilisant par exemple des techniques de classement)…

Remerciements

Je voudrais particulièrement remercier Daniel PACHOLCZYK, professeur au LERIA (Université d'Angers), pour son aide et ses conseils avisés et Michel LUCAS, professeur à l'Ecole Centrale de Nantes, pour ses nombreuses relectures et ses encouragements.

Bibliographie

Chau94 D. Chauvat,
Le projet VoluFormes :
un exemple de modélisation déclarative avec contrôle spatial,
Thèse de doctorat, Nantes, Décembre 94, 225 pages

Col92 C. Colin,
Les propriétés dans le cadre d'une modélisation géométrique déclarative,
MICAD 92, Paris, 1992, pp 75-94

Des95a E. Desmontils,
Les modeleurs déclaratifs,
Rapport de recherche IRIN-95, Nantes, Septembre 1995, 127 pages

Des95b E. Desmontils,
Formalisation des propriétés en modélisation déclarative à l'aide des sous-ensembles flous,
Rapport de recherche IRIN-106, Nantes, Décembre 1995, 41 pages

Dje91 N. Djedi
Modélisation en synthèse d'images : utilisation d'une méthodologie déclarative,
Thèse de doctorat, Toulouse, Novembre 1991, 196 pages

Don93 S. Donikian,
Une approche déclarative pour la création de scènes tridimensionnelles : application à la conception architecturale,
Thèse de Doctorat, Rennes, 1993, 149 pages

DPr93 D. Dubois, H. Prade,
Ensembles flous, raisonnement et décision,
Rapport IRIT/93-52-R, Toulouse, Décembre 1993, 75 pages

LDe96 M. Lucas, E. Desmontils
Les modeleurs déclaratifs,
Revue Internationale de CFAO et Infographie, volume 10, n°6/1995, pp 559-585

LMM89 M. Lucas, D. Martin, P. Martin et D. Plemenos,
Le projet ExploFormes : quelques pas vers la modélisation déclarative de formes,
Journées AFCET-GROPLAN, Strasbourg, 1989, publié dans BIGRE, n° 67, janvier 1990, pp 35-49

MaM91 P. et D. Martin,
Usage des règles de modification pour la génération déclarative de scènes,
Rapport de recherche LIST 91-08, NANTES, Juin 1991, 44 pages

Mat93 L. Mastyska,
Logic Programming with Fuzzy Sets,
Technical Report, City University, London, December 1993, 20 pages

Paj94 L. Pajot-Duval,
Modélisation déclarative de configurations de segments de droite : le projet FiloFormes,
Thèse de doctorat, Nantes, Juin 1994, 142 pages

Ple91 D. Plemenos,
Contribution à l'étude et au développement des techniques de modélisation, génération et visualisation de scènes : le projet MultiFormes,
Thèse de Doctorat d'état, Nantes, 1991, 308 pages

Pou94 F. Poulet,
Modélisation déclarative de scènes tridimensionnelles par énumération spatiale : le projet SpatioFormes,
Thèse de doctorat, Rennes, Juin 1994, 135 pages

Ton95 J.-R. Tong-Tong,
La logique floue,
Hermès, Paris, 1995, 160 pages

Zad65 L. A. Zadeh,
Fuzzy Sets,
Information and Control, vol. 8, 1965, pp 338-353