

BELDJILALI B. BESSAOUD T. SULTAN L.

Institut de Mathématiques et Informatique

Université d'es-sénia ORAN ALGERIE.

ABSTRACT:

Dans cet article nous donnons des notions générales sur les bases fondamentales des systèmes flous de contrôle. Nous proposons également des algorithmes pour des modèles de réalisation des programmes flous basés sur la méthode de Zadeh ainsi qu'un algorithme pour un modèle basé sur l'évaluation des prédicats flous. Ces algorithmes peuvent être utilisés par des ingénieurs pour une organisation optimale des systèmes de contrôle utilisant des programmes flous.

1. INTRODUCTION:

Les algorithmes et programmes flous ont été introduits pour des simulations linguistiques de problèmes complexes ou mal définis qui ne peuvent être formulés par des méthodes quantitatives.

Dans l'article de Zadeh [10] un programme est défini comme une séquence ordonnée d'instructions se composant d'un état conditionnel et d'une assignation floue. Il utilise "The compositional rule of inference" pour construire le modèle d'exécution des états conditionnels.

Mamdani-King [4], Kickert [2] et d'autres auteurs appliquent ce modèle à des systèmes de contrôle.

Cependant la réalisation (voir (2.2) et (2.3)) nécessite un temps machine relatif à l'exécution de l'opérateur Max-min et un espace mémoire pour le stockage des relations S importants.

Mamdani [5], Van Nauta Lemke-Kickert ont proposé dans [2] une optimisation de ce modèle pour les programmes flous appliqués à des systèmes de contrôle dans le cas où les inputs du système de contrôle ne sont pas des vecteurs flous mais des valeurs numériques exactes liées aux variables linguistiques.

Une autre alternative a été celle de Chang [1], qui utilise des relations linguistiques comme opérateur de branchement conditionnel de même qu'il représente un programme flou par un graphe orienté fini. Mais il n'utilise pas un critère de branchement pour décider de la sortie du test à exécuter. En effet, les deux branches sont exécutées en parallèle et c'est le résultat de leur évaluation qui est utilisé comme critère de branchement.

Enfin dans [8,9] des méthodes ont été proposées, elles utilisent les "schémas d'opérateurs d'algorithmes" pour décrire des programmes flous et dans lesquelles on procède à l'évaluation des prédicats pour décider du branchement à effectuer.

Ces méthodes peuvent être utilisées pour construire des modèles généraux d'exécution de programmes flous.

Dans ce qui suit nous donnons des notions générales sur les bases fondamentales des systèmes flous de contrôle de processus industriels et proposons des algorithmes pour des modèles de réalisation de programmes flous de contrôle.

2. NOTIONS DE BASE SUR LES SYSTEMES FLOUS DE CONTROLE:

2.1 Le concept de variable linguistique

On définit une variable linguistique floue L_i [3] par le quadruplet:

$(U_i, \{T_{ij}\}, X_{L_i}, \{U_{j1}\})$ avec $j=1, \bar{n}; l=1, m$

$\{T_{ij}\}$: ensemble des termes flous que prend la variable L_i ,

X_{Li} : ensemble des valeurs numériques dans lequel est définie la variable linguistique L_i , et que l'on appelle univers du discours; Les éléments quantitatifs de X_{Li} sont notés d_{il}

$\{U_{ij}\}$ ensemble des fonctions d'appartenance des éléments quantitatifs d_{il} aux termes T_{ij} .

La figure 1 donne la structure d'une variable linguistique.

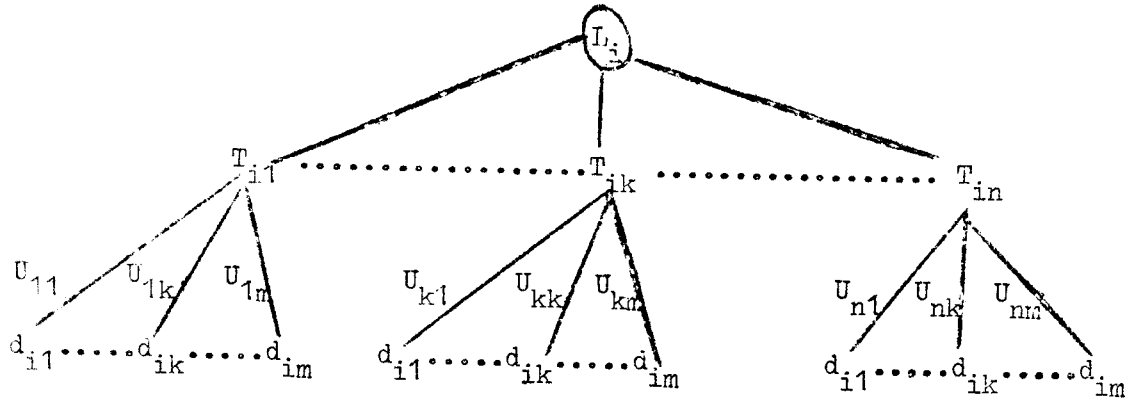


Fig 1: Structure d'une variable linguistique

2.2 Formalisation des systèmes de contrôle flou par l'utilisation de variables linguistiques.

L'une des méthodes de formalisation des systèmes de contrôle est la méthode de contrôle situationnel [7] qui permet de décrire tout système de contrôle par des classes de situations et des classes d'actions relatives à ces situations .

Dans cet article nous appliquons le concept de variable linguistique pour formaliser la technique de contrôle situationnel.

Dans un programme flou de contrôle des processus, les inputs sont caractérisés par des variables linguistiques qui déterminent l'état actuel du processus à contrôler . Ces variables sont appelées variables linguistiques de situation (VL^s) . Les outputs représentent les actions de contrôle à appliquer au processus et sont appelés variables linguistiques de décision (VL^d) .

A l'étape de formulation du programme de contrôle, les variables linguistiques sont représentées par des sous-ensembles flous (termes flous) qui donnent les valeurs linguistiques des variables.

Un système peut avoir une ou plusieurs variables de décision ; dans le cas de plusieurs variables de décision le système est décomposé en plusieurs sous-systèmes . En effet , chaque action de contrôle est évaluée indépendamment des autres et nécessite à elle seule un programme d'évaluation .

Ainsi, nous pouvons représenter un système de contrôle par l'un des trois (3) schémas décrits dans la figure 2 .

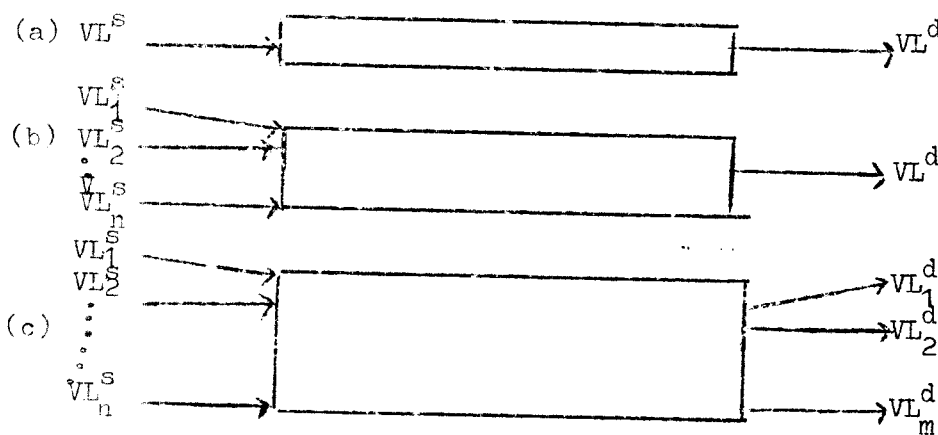


Fig 2. Différents cas de systèmes de contrôle

2.3 Modèle de base de réalisation :

Ce modèle de base proposé par Zadeh [10] a été adapté par Mamdani [5] à la formulation et la réalisation d'un programme flou de contrôle .

Les règles de décision du modèle, appliqué au cas de la figure 2.a, sont implémentés comme un ensemble d'états conditionnels flous de la forme :

$$\text{" IF } L_1 = T_{1i} \text{ THEN } L_2 := T_{2i} \text{" } \quad (2.1)$$

L'état conditionnel est représenté par une relation floue de la forme

$$\mu_s = \text{Min} (\mu_{T_{1i}}, \mu_{T_{2i}}) \quad (2.2)$$

et l'expression est évaluée en utilisant "The compositional rule of inference" , formule (2.3) pour une valeur particulière T_{1i} de T_{1i} .
Le sous-ensemble flou T'_{2i} sera calculé comme suit:

$$\mu_{T'_{2i}} = \text{max-min} (\mu_{T_{1i}}, \mu_s) \quad (2.3)$$

Dans plusieurs cas, les règles sont plus complexes que l'exemple ci-dessus ; c'est le cas des processus avec plusieurs variables de situation .

Pour le cas de figure 2.b, les règles seront de la forme

$$\text{IF } L_1 = T_{1i} \text{ THEN (IF } L_2 = T_{2i} \text{ THEN (... THEN (IF } L_n = T_{ni} \text{ THEN } L_d := T_{di} \text{)...))} \quad (2.4)$$

mais la même méthode d'évaluation est appliquée.

Plusieurs règles sont nécessaires pour définir complètement un programme de contrôle . Les résultats d'évaluation de chaque règle sont combinés en utilisant l'opérateur " OR " (max) pour obtenir le résultat final qui est un ensemble flou de fonctions d'appartenance pour toutes les actions de contrôle possibles. Afin de choisir l'action à entreprendre , on effectue une approximation linguistique. Elle consiste à déterminer l'action de contrôle qui a la fonction d'appartenance la plus compatible avec le terme flou de sortie.

3. ALGORITHMES POUR L'EXECUTION DES PROGRAMMES FLOUS DE CONTROLE.

La réalisation du modèle de Zadeh est très complexe. En effet, à partir de 2 ou 3 variables de situation, elle nécessite des tableaux à plusieurs dimensions et un temps de calcul très importants. Il est nécessaire de sauvegarder également pour chaque règle la relation correspondante. Des techniques de réalisation optimisées ont été développées par Mamdani [5], Kickert et Lemke [2] .

Nous présentons, dans ce chapitre, des algorithmes pour 2 de ces méthodes.

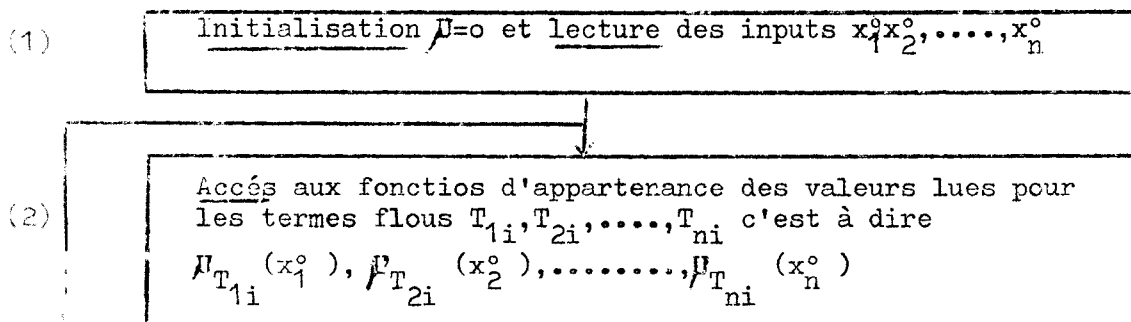
Première méthode: Soit un programme avec n variables linguistiques de situation constitué par une suite d'opérateurs de la forme (2.4) . L'évaluation de cet opérateur se fait par la formule suivante :

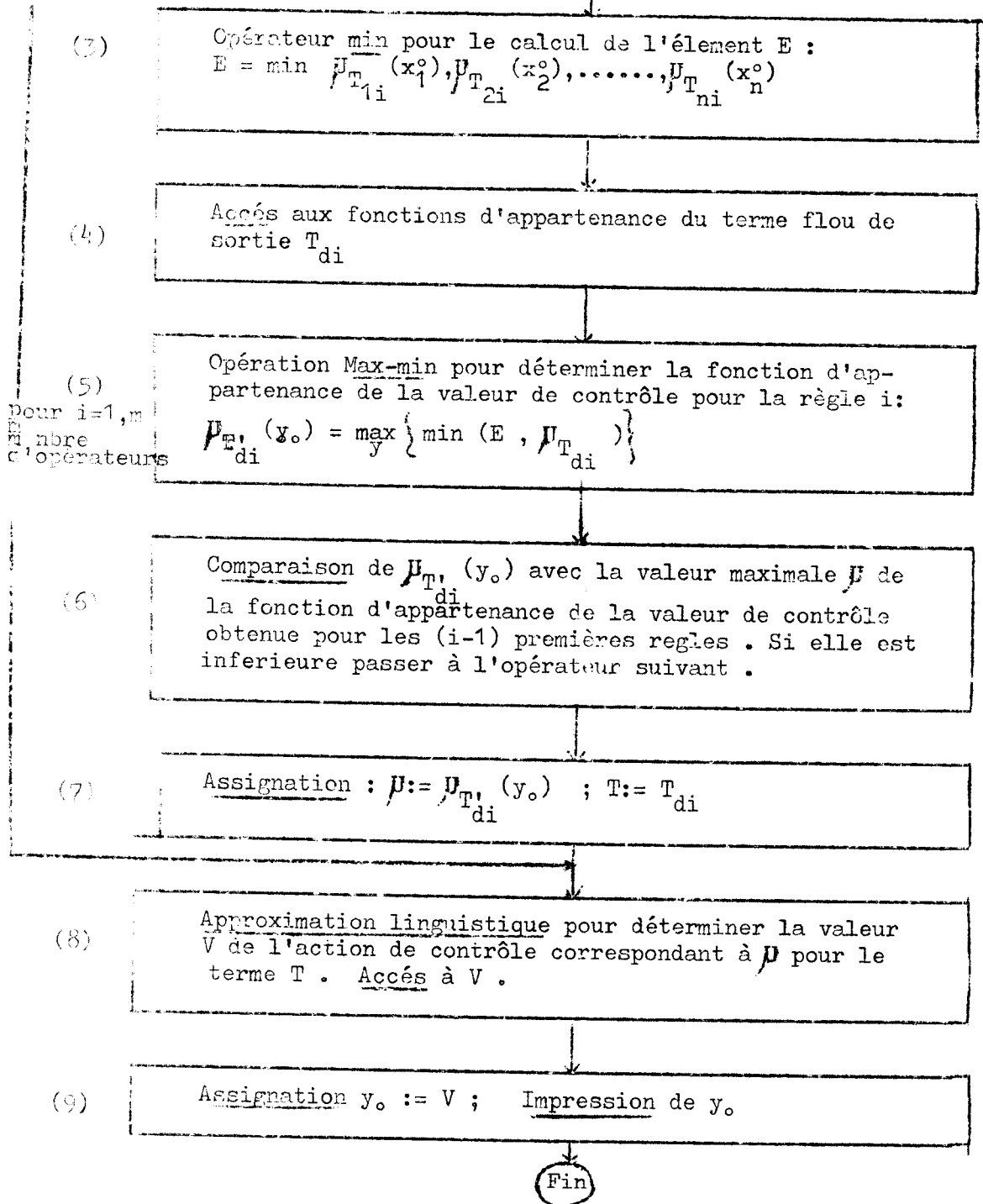
$$\mu_{T_{di}}(y_0) = \text{max}_y \min (\mu_{T_{1i}}(x_1^0), \mu_{T_{2i}}(x_2^0), \dots, \mu_{T_{ni}}(x_n^0), \mu_{T_{di}}(y)) \quad (3.1)$$

$x_j^0, j=1, n$ étant les valeurs actuelles des variables de situation.

Cette méthode ne nécessite pas la sauvegarde des relations. Elle permet de gagner considérablement en espace mémoire et en temps de calcul .

L'algorithme sera le suivant:

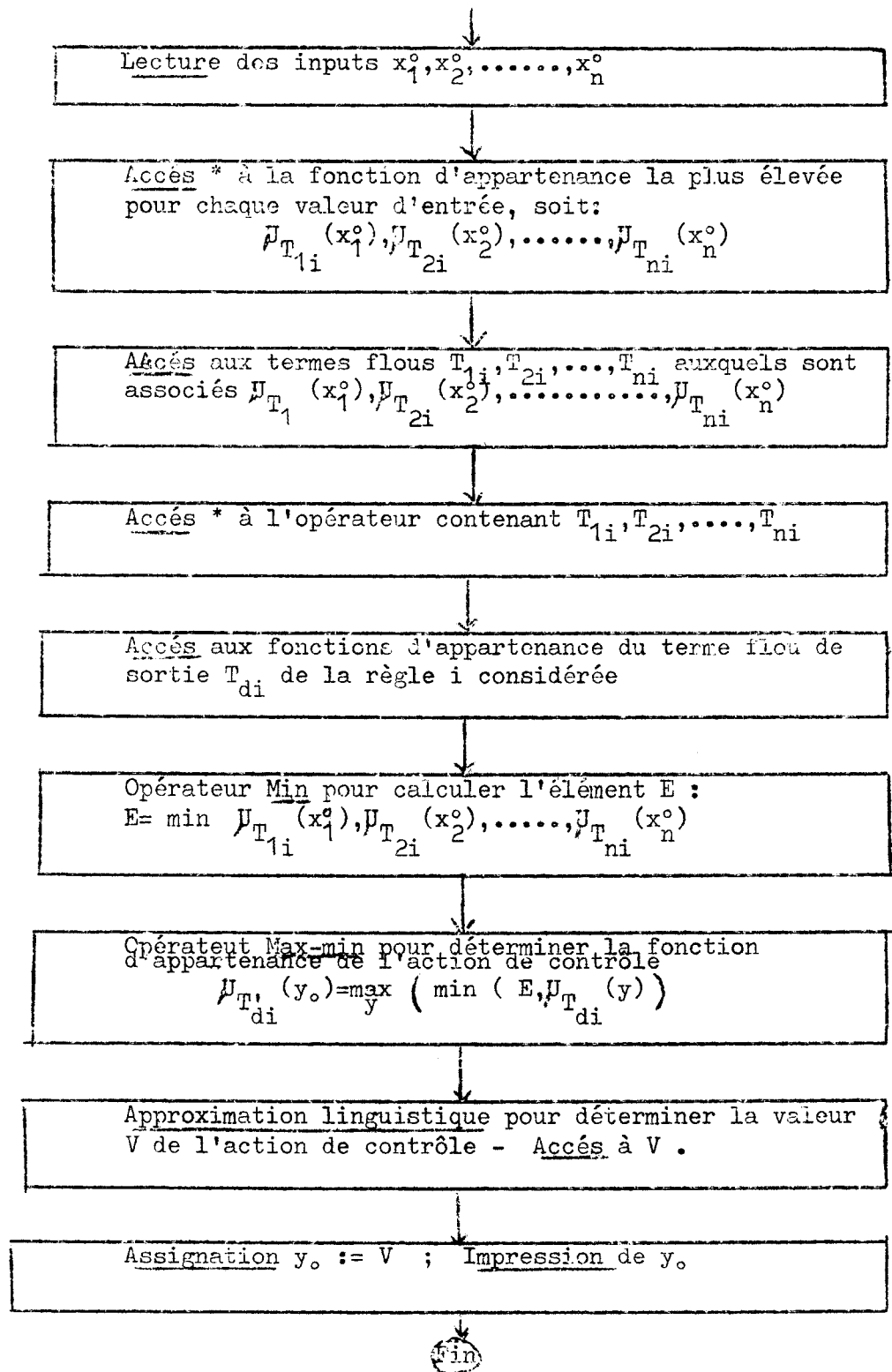




Deuxième méthode: La première méthode nécessite l'évaluation de tous les opérateurs, par contre dans cette seconde méthode on n'exécute que l'opérateur qui contient les termes les plus compatibles avec les valeurs d'entrée, ce qui entraîne un gain de temps appréciable.

La compatibilité est associée aux fonctions d'appartenance de la façon suivante : le terme ayant la fonction d'appartenance la plus élevée pour la valeur d'entrée sera la plus compatible avec cette valeur. La réalisation de l'opérateur se fera toujours en utilisant "The compositional rule of inference" sous sa forme optimisée donnée par la formule (3.1).

L'algorithme sera donc :



* il y'a possibilité pour les blocs (2) et (4) de réaliser un opérateur d'accès en utilisant le principe de la mémoire associative [8]

4 METHODE D'EXECUTION DE PROGRAMMES FLOUS DE CONTROLE PAR EVALUATION DES PREDICATS

FLOS:

La méthode consiste en l'exécution d'un programme flou où la condition de branchement est caractérisée par l'évaluation du prédicat flou. Cette évaluation donne une valeur comprise dans l'intervalle $[0,1]$, laquelle sera comparée à une valeur seuil, α , fixée par l'opérateur.

Le résultat de cette comparaison permet d'attribuer, la valeur 0 (faux) ou 1 (vrai) au prédicat flou .

4.1. Interprétation de l'exécution de l'opérateur conditionnel .

Considérons l'opérateur:

IF $L_s = T_i$ THEN $L_d := T_j$ ELSE $T_d := T_k$

en utilisant les résultats de SULTAN.L [8] et [9], nous aurons le schéma de réalisation suivant :

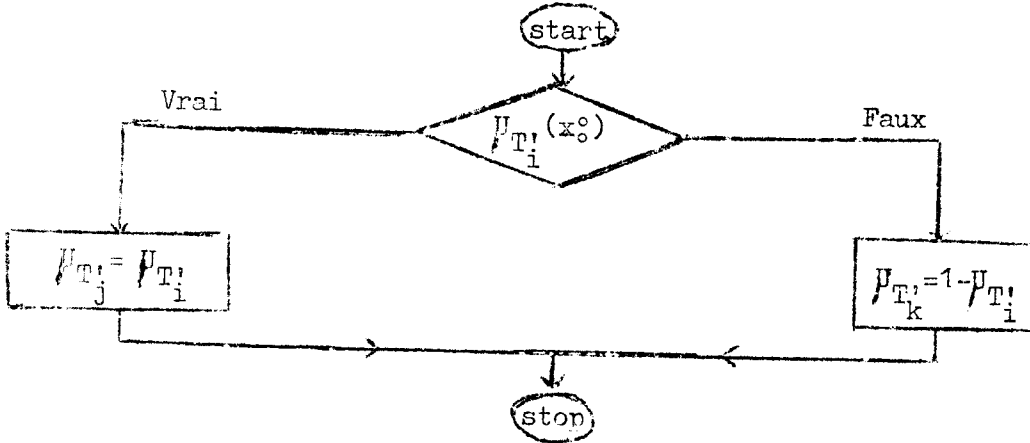


Fig:3

où $\mu_{T_i}(x_0)$ est le degré de vérité utilisé pour tester si le prédicat est vrai ou faux. ON utilise ensuite le principe de la mémoire associative de la façon suivante μ_{T_j} et μ_{T_k} sont des adresses dont le contenu sera la valeur de la variable linguistique de décision. La figure 4 illustre cette technique.

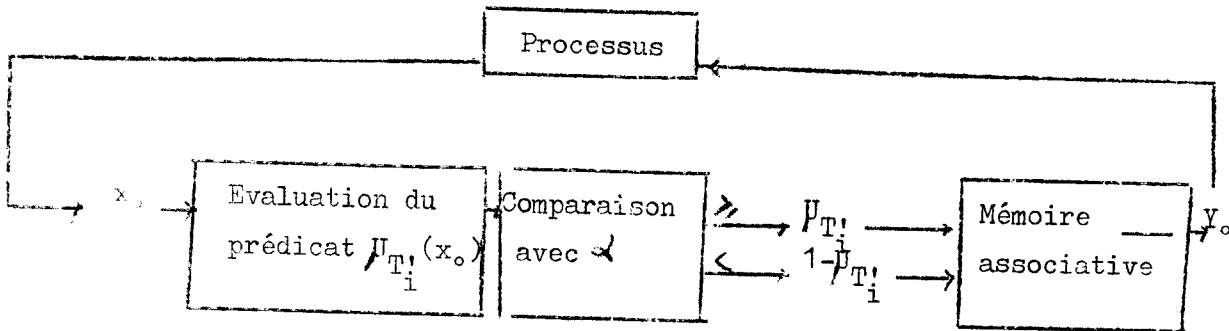
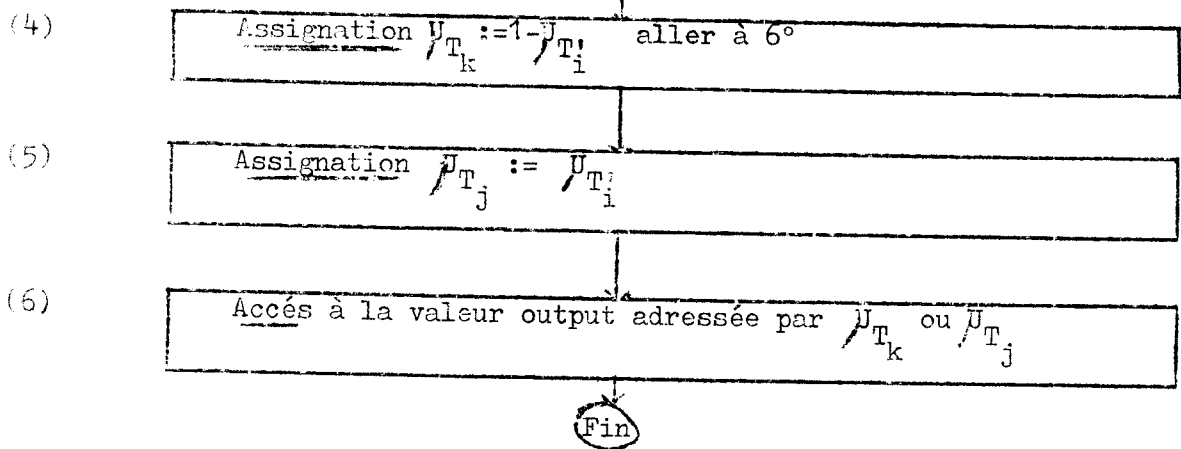


Fig:4. Schéma de réalisation d'un opérateur utilisant le principe de la mémoire associative .

L'algorithme de réalisation sera le suivant :

- (1) Lecture de l'input x_0 et de la valeur seuil
 - (2) Accès dans le sous_ensemble flou T_i à μ correspondant à x_0 .
 - (3) Comparaison de $\mu_{T_i}(x_0)$ et α si $\mu_{T_i}(x_0) < \alpha$ aller à 5°
- (A)

(A)



4.2. Réalisation d'un système de contrôle à 1 input et 1 output :

Le système est représenté par un ensemble d'opérateurs de type (2.1). L'organigramme du programme flou de ce système sera le même que celui de la figure 3 à la seule différence que lorsque le prédicat pour un terme n'est pas satisfaisant on passe au terme suivant.

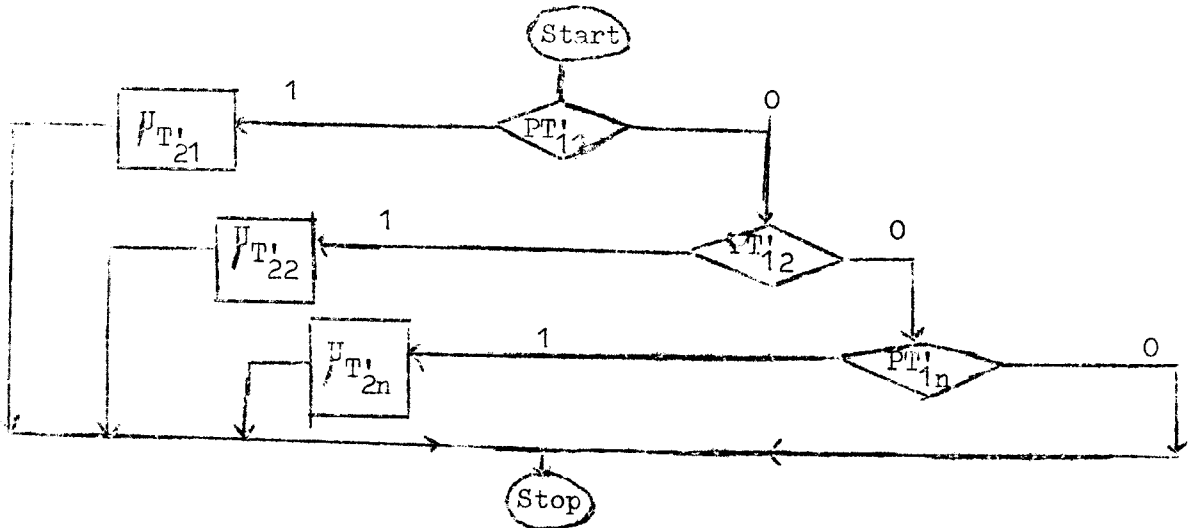
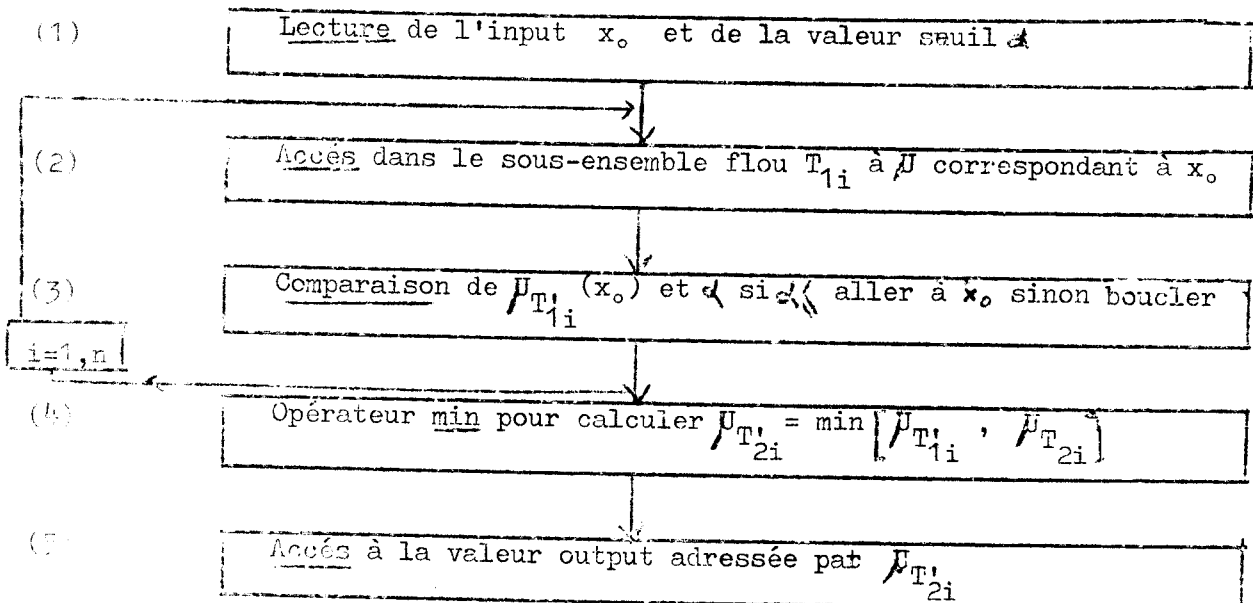


Fig 5: Organigramme de réalisation d'un système à 1 input et 1 output.

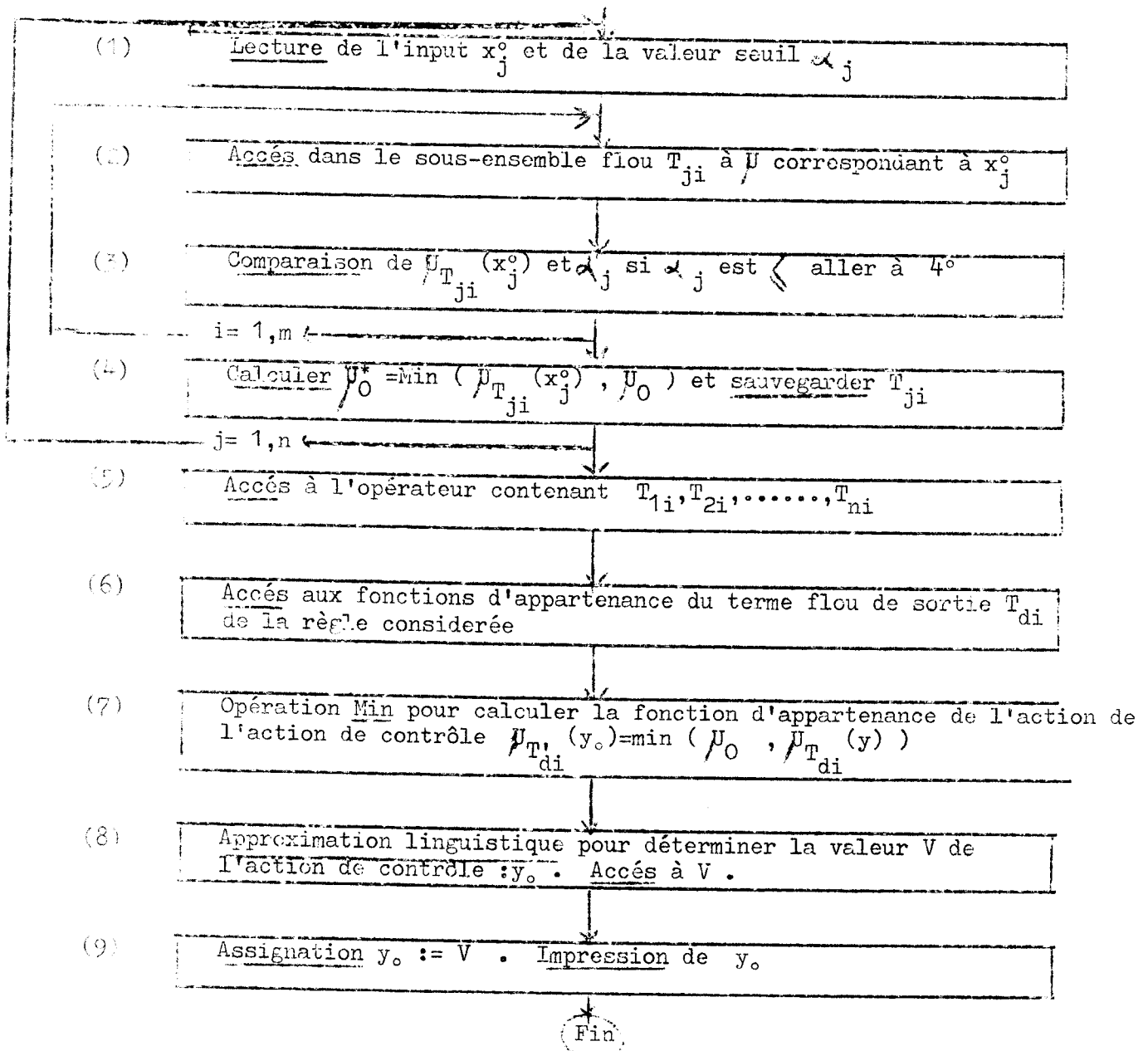
L'algorithme de réalisation sera :



4.3. Réalisation d'un système de contrôle à n inputs et 1 output.

Le système est représenté par un ensemble d'opérateurs du type (2.4). La méthode consiste à trouver un chemin solution en passant par chaque variable linguistique. Un terme, par variable linguistique, est retenu. L'ensemble des termes retenus détermine, en utilisant le principe de la mémoire associative, l'action de contrôle à entreprendre.

L'algorithme de réalisation sera:



μ_0 est initialisée à 1.

5. CONCLUSION.

Dans cet article nous avons donné une classification des systèmes de contrôle puis fait une étude comparative, des méthodes d'exécution de programmes flous de contrôle, basée sur l'optimisation des ressources machine.

Nous avons par ailleurs développé la technique de formalisation pour l'application de la méthode de réalisation des programmes flous de contrôle par évaluation des prédicats flous. Cette* présente l'avantage d'être plus générale.

Dans les algorithmes nous avons appliqué le principe de la mémoire associative afin d'optimiser le temps de calcul.

Les résultats obtenus peuvent être utilisés comme base pour l'organisation des systèmes flous basés sur des modèles fonctionnels constructifs pour des systèmes à microprocesseurs.

* méthode

REFERENCES.

- [1] C.L.CHANG, "Interprétation and éxécution of fuzzy programs" in ZADEH.L.A, TANAKA.K, SHIMURA.M and others "Fuzzy Sets and Their Application to Cognitive and Décision Processes", Academic press, New York, pp 191-217, 1975.
- [2] W.J.M.KICKERT and H.R.VAN NAUTA LEMKE, "application of a fuzzy controller in a warm water plant", Automatica 12, 301-308, 1976.
- [3] W.J.M.KICKERT "an example of linguistic modelling: the case of mulder's theory of power" in MADAN .M.GUPTA and others "advances in fuzzy set theory and applications " north-Holland, pp 519-540, 1979.
- [4] P.J.KING and E.H.MAMDANI , "The application of fuzzy control systems to industrial processes". Automatica, 13, pp 235-242, 1977 .
- [5] E.H.MAMDANI , "Application of fuzzy logic to approximate reasoning using linguistic synthesis ". IEEE Transactions on computers. Vol C26, N°12, Dec 1977.
- [6] PARNEGSHVILLI.E.M. "Microelectronic Associative Processors", Moscou, Sovetskya Radio, 1975 (in RUSSIAN).
- [7] POSPELOV.D.A "Situational control Theory and Practics", Moscou, Sovetskya Radio 1977.(in RUSSIAN).
- [8] SULTAN.L.H. "A Constructive approach to the realisation of fuzzy programs", to appear in Fuzzy Sets and Systems.
- [9] SULTAN.L.H. "Fuzzy logic implementation in the realisation of fuzzy algorithms" Internal Report, University of ORAN, Dept; d'Informatique.
- [10] L.A.ZADEH. "Outline of a new approach to the analysis of complex systems and decision processes", IEEE Trans.Syst.Man.Cybern.Vol SMC -3, pp 28-44, Jan 1973.