

APPRENTISSAGE GENETIQUE GLOBAL D'UN CONTROLEUR FLOU A DEUX VARIABLES BASE SUR LA SIMULATION D'UN VEHICULE AUTONOME

L.Gacôgne

LAFORIA CNRS - Université Paris VI place Jussieu 75252 Paris 5°

tel : 44 27 70 02 fax : 44 27 70 00 mail : gacogne@laforia.ibp.fr

Institut d'Informatique d'Entreprise (CNAM) allée J.Rostand 91025 Evry tel : 60 77 97 40

Abstract We are interested in this paper by the functional aspect of the Mamdani-Sugeno algorithm about fuzzy control. Indeed this one gives a way to build a continuous mapping from $[-1, 1]^n$ to $[-1, 1]^p$. With a package of rules relying each one a fuzzy point of R^n to a fuzzy point of R^p we study the link existing between the definition of those "fuzzy points" according to a triangular norm and the norm used on R^n . More practically we axed to fuzzy controller based on rules with two, three or four premises and two conclusions for instance. We use for that the simulation of an autonomous vehicle regulating itself in the same time its direction and speed with only one controller. After several experiments with genetic algorithms to learn not only the number and the numeric values of the rules but all the characteristics of the robot, we conclude on the advantage of fuzzy control generalized with several outputs in spite of the difficulty of the learning and the evaluation in our benchmark.

Résumé On s'intéresse à l'aspect fonctionnel de l'algorithme de Mamdani-Sugeno relatif au contrôle flou. Celui-ci permet en effet une construction d'une fonction continue de $[-1, 1]^n$ dans $[-1, 1]^p$. Grâce à un jeu de règles reliant chacune un point flou de R^n à un point flou de R^p , on étudie le lien qu'il peut y avoir entre la définition de ces "points flous" suivant une norme triangulaire, et la norme employée sur R^n . On a cherché à illustrer plus concrètement des contrôleurs flous basés sur des règles ayant deux, trois ou quatre prémisses et deux conclusions par exemple. On se sert pour cela de la simulation d'un véhicule autonome réglant simultanément sa direction et sa vitesse à l'aide d'un seul contrôleur. Ayant conduit plusieurs expériences d'apprentissages génétiques visant à acquérir non seulement des règles numériques et le nombre de ces règles, mais tous les paramètres décrivant le robot, on conclue sur l'avantage de la commande floue généralisée à plusieurs sorties en dépit de la difficulté de l'apprentissage et de l'évaluation.

I VECTEURS FLOUS ET t-NORMES

Le produit d'espaces normés est défini habituellement par $n(x, y) = \sup(n_1(x), n_2(y))$, cependant il est possible de généraliser par des

définitions topologiquement équivalentes en dimension finie (théorème de Riesz).

Soient (E_1, n_1) et (E_2, n_2) deux espaces normés et S une application de R^{+2} dans R^+ , en définissant n sur le produit $E_1 * E_2$ par $n(x, y) = S(n_1(x), n_2(y))$ on peut montrer certains liens entre les propriétés de n et celle de S .

a) S est sous-additive pour l'addition des couples $\Leftrightarrow n$ vérifie l'inégalité triangulaire

b) Si S est croissante [$a < c$ et $b < d \Rightarrow S(a, b) \leq S(c, d)$] alors :

$S(1, 1) = 1 \Leftrightarrow$ Le produit des boules unités de E_1 et E_2 est la boule unité de $E_1 * E_2$

c) La norme n se projette suivant n_1 sur E_1 et n_2 sur $E_2 \Leftrightarrow [\forall a S(a, 0) = S(0, a) = a]$

d) $[n(x, y) = 0 \Rightarrow x = y = 0] \Leftrightarrow [\forall a, b S(a, b) = 0 \Leftrightarrow a = 0 \text{ et } b = 0]$

S se comporte alors comme une disjonction.

Cependant le fait que n soit une norme n'est pas équivalent au fait que $S/[0, 1]^2$ soit une conorme triangulaire. [Menger 42, Schweitzer 83]

En effet, si nous prenons par exemple dans R^2 la norme définie par $n(x, y) = |x| + |x + y|$, nous avons $S(1, 1) = 2$ et S non associative. En prenant $n(x, y) = |x| + 2|y|$, n est également une norme telle que S n'est pas commutative et $S(1, 1) = 3$. Enfin, réciproquement si S est la conorme algébrique définie par $S(a, b) = a + b - ab$, alors n ne vérifie pas la propriété $[\forall \lambda, X n(\lambda X) = |\lambda| X]$, mais l'inégalité triangulaire.

Cependant, si nous considérons les trois normes les plus utilisées pour le produit d'espaces normés, nous retrouvons comme applications S associées :

Conorme de Zadeh $S(a, b) = \max(a, b)$ pour la norme "sup"

Conorme de Lukasiewicz $S(a, b) = \min(1, a + b)$ pour la norme de Hamming $N(x, y) = n_1(x) + n_2(y)$

Conorme de Yager $S(a, b) = (a^2 + b^2)^{1/2}$ c'est la norme euclidienne.

La question du choix d'une norme triangulaire pour la conjonction des prémisses d'une règle floue, correspond donc partiellement à celle du choix de la norme dans R^n à propos de la définition des voisinages flous qui va suivre.

II MODELISATION D'UNE FONCTION VECTORIELLE PAR L'ALGORITHME DE MAMDANI-SUGENO

II - 1° Modélisation numérique

Considérons une fonction f entre deux compacts de deux espaces normés de dimensions finies, pour simplifier nous nous placerons de $[-1, 1]^n$ dans $[-1, 1]^p$

Nous notons $B(a, r)$ la "boule floue" ou "hypercône" centrée en $a \in [-1, 1]^n$ et de rayon $r \in \mathbb{R}^+$ le sous-ensemble flou définie par la fonction d'appartenance $\mu_{B(a, r)}(x) = \max(0, 1 - N(a - x) / r)$

Si T est une t -norme et N la norme associée (un des trois cas précédent), on aura donc :

$\mu_{B(a, r)}(x) = T(\mu_{B(a_1, r)}(x_1), \dots, \mu_{B(a_n, r)}(x_n))$ où $a = (a_1, \dots, a_n)$ et $x = (x_1, \dots, x_n)$

On peut maintenant formaliser les règles du type : Si $x = (x_1, x_2, \dots, x_n) \in B(a_i, r_i)$ alors $f(x)$ est $c_i = (c_1, \dots, c_p)$ en posant :

Définition : f est une fonction de Sugeno définie par un ensemble fini de "règles"

$\{(a_1, r_1, c_1), \dots, (a_k, r_k, c_k)\}$ où chaque $a_i \in [-1, 1]^n$, $r_i \in \mathbb{R}^+$ et $c_i \in [-1, 1]^p$, si :

$$\forall x \in [-1, 1]^n \quad f(x) = (\sum \alpha_i * c_i) / \sum \alpha_i \text{ avec}$$

$\alpha_i = \mu_{B(a_i, r_i)}(x)$ pour i allant de 1 à k .

Définition : f est une fonction de Sugeno non lacunaire sur K si $\forall x \in K$, x appartient au support d'au moins une des boules (a_k, r_k) $1 \leq i \leq k$ définissant f .

Définition : g est une restriction de Sugeno de pas r pour la fonction f si g est une fonction de Sugeno définie par un ensemble $\{(a_1, r_1, f(a_1)), \dots, (a_k, r_k, f(a_k))\}$ avec $r = \max(r_1, \dots, r_k)$.

Théorème : Si K est un compact de \mathbb{R}^n et si f est définie et continue sur K à valeurs dans \mathbb{R}^p , toute suite $(g_k)_{k \in \mathbb{N}}$ de restrictions de Sugeno de f de pas r_k , non lacunaires et dont la suite des pas converge vers 0, converge uniformément vers f .

En effet $\forall \epsilon > 0$ il existe un rang k à partir duquel $r_k = \text{pas}(g_k) < \epsilon$ alors $\forall x \in K$:

$N(f(x) - g_k(x)) = N(f(x) - (\sum \alpha_i * g_k(x)) / \sum \alpha_i) = N(\sum \alpha_i (f(x) - g_k(x))) / \sum \alpha_i$ pour $1 \leq i \leq q_k$ si g_k est définie par q_k boules floues, et donc :

$N(f(x) - g_k(x)) \leq (\sum \alpha_i N(f(x) - g_k(x))) / \sum \alpha_i \leq \epsilon$ car f est uniformément continue sur le compact K . (Le système du "contrôle flou" constitue un approximateur universel).

Théorème : Si f est lipshitzienne de rapport q sur E et g une restriction de Sugeno non lacunaire de f , de pas r , alors on dispose d'une majoration $N(f - g) \leq qr$ sur E .

Résultat trivial.

II - 2° Application au suivi de trajectoire par un véhicule autonome contrôlant sa direction et sa vitesse

L'idée de contrôler simultanément deux ou plusieurs paramètres ne semble pas encore avoir été exploitée. On peut signaler [Watanabe 92], mais il s'agit en fait de plusieurs contrôleurs à une sortie tous placés en parallèle.

A l'aide de quelques règles, il est relativement facile de contrôler la variation qu'un mobile doit apporter simultanément à sa direction et à sa vitesse afin de suivre au mieux une trajectoire.

Pour une largeur d'écran de 500, et un pas minimum $ds=8$, les paramètres sont un les maximums du pas pm , de l'accélération acc et de la variation de direction am . Le contrôle s'effectue au regard de la distance algébrique x à la courbe, que le robot mesure suivant deux ouvertures de l'ordre de $ouv=\pm 60^\circ$ par rapport à sa direction, et de la variation de cette distance dx . Il s'agit donc de la mise en oeuvre d'une fonction $[-1, 1]^2$ vers $[-1, 1]^2$. Plus précisément: $x, dx \in [-dm, dm] * [-vm, vm] \rightarrow da, dv \in [-am, am] * [-acc, acc]$

La ligne à suivre est décrite par $f(x) = M + \text{Hexp}(-x/H) \sin(\pi \cdot \text{nsi} \cdot x / L)$ où $\text{nsi} = 7$ dans la fenêtre $[0, L] * [0, H]$ où M est de l'ordre de $H/2$.

Une solution est du type $(ouv \ dm \ vm \ am \ acc \ pm \ R1 \ R2 \ R3 \ \dots)$ où chaque règle (en nombre indéterminé) possède la forme $(a \ b \ r \ c \ d)$ exprimées en centièmes.

Si $T = \text{sup}$, on a donc pour $x, y \in [-1, 1]^2$ le coefficient de satisfaction de la règle $\alpha = \max(1 - \text{sup}(|x - a|, |y - b|) / r, 0)$ et (c, d) aura le poids α dans l'examen de toutes les règles.

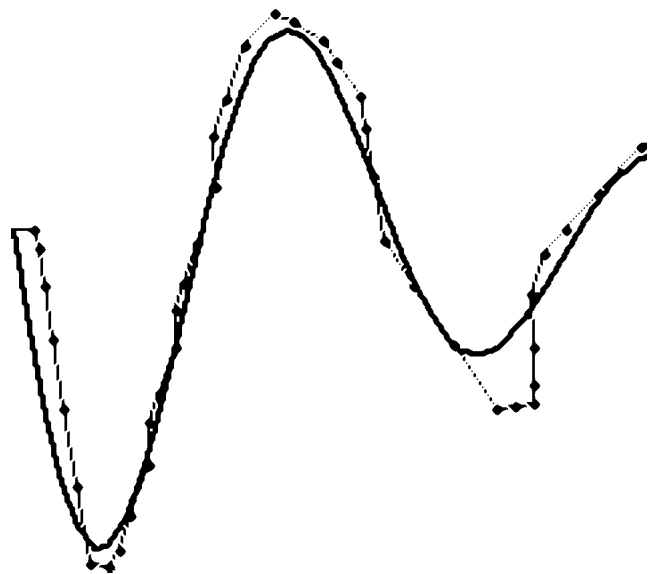


Figure 1 : un robot accélérant dans les lignes droites. Pour $ouv=60^\circ$, $dm=25$, $vm=60$, $am=50^\circ$ $acc=25$ et $pm=30$. Règles $(-100 \ -100 \ 100 \ 140 \ -100)$, $(0 \ 25 \ 100 \ 20 \ 25)$ et leurs symétriques.

Un jeu de règles est stable par symétrie avec la définition $\text{sym}(a, b, r, \text{al}, \text{ac}) = (-a, -b, r, -\text{al}, -\text{ac})$. Intuitivement ces règles doivent bien sûr être du type : (NB NB r PB NB) ou (NB PB r PS NS) et leurs symétriques.

II - 3° Modélisation symbolique

En fixant les boules floues par des centres -1, -1/2, 0, 1/2, 1, et un rayon $r = 1/2$, (et leur donnant les appellations courantes respectivement de NB, NS, ZE, PS, PB), il est possible de reprendre le même algorithme en ajoutant toutefois l'utilisation d'un prédicat ANY (toujours vrai, c'est à dire une boule de centre 0 et de rayon infini) qui a l'avantage de pouvoir décrire en même temps des règles de R^m dans RP pour des m tels que $m \leq n$, ce qui peut aider à écrire des règles dont certaines prémisses n'ont pas d'importance.

Exemple 1 On considère le cercle $t \in [-1, 1] \rightarrow (x = \cos(\pi t), y = \sin(\pi t)) \in [-1, 1]^2$

Le test est fait sur 41 valeurs régulièrement distribuées dans $[-1, 1]$, la valeur d'un jeu de règles étant défini par l'erreur moyenne mesurée comme distance de Hamming entre la sortie réelle et la sortie désirée. Celle-ci est rapportée en pourcentage par rapport au diamètre du cube unité. Nous utilisons un algorithme génétique sur une population d'une dizaine de jeux de règles initialement aléatoires. Le résultat de 200 générations donne un jeu de règles assez médiocre (5% d'erreur), le "ou" signifiant que l'on a plusieurs règles de mêmes prémisses.

NB \rightarrow (NB NB) ou (NB PS) NS \rightarrow (ZE NB)
 ZE \rightarrow (PB ZE) PS \rightarrow (ZE PB)
 PB \rightarrow (NB PB) ou (NB ZE)

Exemple 2 Soit le paraboloïde hyperbolique $(x, y) \in [-1, 1]^2 \rightarrow z = x^2 - y^2 \in [-1, 1]$. La table de z en fonction de x et y donne une erreur de 4% :

PB	ZE	PS	PB	PS	ZE
PS	NS	ZE	PS	ZE	NS
ZE	NB	NS	ZE	NS	NB
NS	NS	ZE	PS	ZE	NS
NB	ZE	PS	PB	PS	ZE
x / y	NB	NS	ZE	PS	PB

Exemple 3 Soit l'hélice $t \in [-1, 1] \rightarrow (x = \cos(\pi t), y = \sin(\pi t), z = t) \in [-1, 1]^3$

La meilleure solution (2% d'erreur moyenne) en 150 générations est :

NB \rightarrow (NB NB PS) (3 fois)
 NS \rightarrow (PB PB PB) ou (NS ZE PB)
 ZE \rightarrow (PB NS NB) ou (PB NB NB) ou 4 fois (PB NS PB)
 PS \rightarrow (PS NB PB)
 PB \rightarrow (NB NS NS) ou (NB ZE PB)

On remarquera que l'évolution de la population étant laissée à elle-même, plusieurs règles de même prémisses ont des conclusions qui se nuancent l'une l'autre, et que le fait qu'apparaissent plusieurs fois la même règle revient à donner un coefficient plus fort à celles-ci.

Exemple 4 Soit la sphère inscrite dans le cube unité : $(\theta, \phi) \in [-1, 1]^2 \rightarrow (x = \cos(\pi\theta)\cos(\pi\phi), y = \sin(\pi\theta)\cos(\pi\phi), z = \sin(\pi\phi)) \in [-1, 1]^3$

En 300 générations nous arrivons à une solution qui rejoint la solution intuitive formée par 17 règles : ANY NB \rightarrow ZE ZE NB et ANY PB \rightarrow ZE ZE PB ainsi que :

PB	NS ZE NS	NB ZE ZE	NS ZE PS
PS	ZE PS NS	ZE PB ZE	ZE PS PS
ZE	PS ZE NS	PB ZE ZE	PS ZE PS
NS	ZE NS NS	ZE NB ZE	ZE NS PS
NB	ZE ZE NS	NB ZE ZE	ZE ZE PS
θ/ϕ	NS	ZE	PS

Celle-ci représentant encore 10% d'erreur en moyenne.

Ces approximations sont certes grossières, mais rappelons que cet algorithme de contrôle flou n'est pas destiné à approcher des fonctions bien connues et que d'autre part on s'est arrêté à une centaine de générations, mais plutôt à être utilisé lors de processus non supervisés et seulement évalués globalement par une heuristique pour laquelle on ne connaît pas a priori de solution optimale.

III EXPERIENCES DE SUIVI DE TRAJECTOIRE

III - 1° La stratégie génétique utilisée pour la synthèse du contrôleur

Lors d'expériences précédentes [Gacogne 93], réalisées sur des contrôleurs flous définis de manière symbolique par une suite de 9 ou 25 labels de prédicats, nous étions parti d'une population aléatoire de n_p individus. A chaque génération, l'ensemble de la population mère et de ses images par les transitions génétiques était triée afin de n'en retenir que les n_p meilleurs en s'écartant légèrement de ce qui est fait habituellement [Goldberg 89] [Holland 93]. En s'inspirant de différentes heuristiques de la recherche opérationnelle [Kirkpatrick 83], on va chercher à associer les méthodes de recherches individuelle et génétique. Ici, les contrôleurs flous sont des suites finies de réels, et nous obtenons rapidement avec cette stratégie une population formée d'éléments voisins (des suites finies ne différant par exemple que sur une ou deux composantes). C'est pourquoi, partant initialement d'une population aléatoire, à chaque génération, lors de l'application d'une transition sur un individu, nous éliminons immédiatement le moins bon entre le père et le fils ou les deux moins bons s'il s'agit d'un croisement opéré sur deux individus. Le tri et la troncature de la population ne sont plus nécessaires et nous conservons donc de la sorte n_p lignées d'autant plus indépendantes que le "cross-over" sera peu employé ou, comme nous l'avons fait, employé tardivement après que la descente vers différents minimums locaux ait été observé. De cette façon

lorsqu'une descente est amorcée vers un minimum, seul le dernier descendant y amenant est conservé, les éléments ascendants n'encombrent pas la population et nous augmentons la probabilité d'avoir plus d'un minimum local et nous sommes de ce fait plus à même d'analyser les résultats.

Les transitions choisies sont différents bruits gaussiens, le croisement mais aussi la création aléatoire de règle, la suppression d'une règle choisie aléatoirement et la migration pour laquelle environ le tiers des coefficients est modifié en leur complément [Franz 72]. Ces opérateurs sont aléatoirement brouillés chaque fois qu'on les réinitialise. Au cours des générations ils sont notés suivant leur capacité à faire descendre l'évaluation des chromosomes et triés suivant cette capacité pour la génération suivante. L'idée d'attribuer un mérite aux transitions a déjà été exploitée [Grefenstette 87] y compris dans des expériences de simulation [Grefenstette 89]. Au cas où aucun chromosome n'aurait pu donner naissance à un plus performant, alors l'ensemble des opérateurs est réinitialisé, ceci ayant pour effet de ne pas faire converger cet ensemble vers un seul type de transition.

Remarque Si dans un intervalle I une fonction réelle possède n minimums, en partant d'un point donné par une variable aléatoire uniformément répartie sur I, et considérant k variables de même loi indépendantes, la probabilité d'obtenir des descentes vers X parmi ces n minimums est donnée par :

$$\text{pr}(X = p) = \sum_{i=0}^p (-1)^{p+i} C_p^i \left(\frac{i}{n}\right)$$

Par exemple si $n = p = k = 3$, on obtient une probabilité de 0.222 mais 0.617 si $k = 5$, 0.825 si $k = 7$ et 0.922 si $k = 9$. Dans la pratique une population de l'ordre d'une dizaine d'individus autorise donc une bonne probabilité d'obtenir 3 minimums.

Exemple, si $f(x)$ est la partie entière de $\text{Hexp}(-x/H)\sin(\pi.nsi.x/L)$ où $nsi = 7$ est le nombre de "sommets" de la sinusoïde amortie dans l'intervalle $[0, L]$ ($L = 495$, $H = 150$ sont les dimensions de l'écran utilisé). Avec $k = 10$ tirages aléatoires dans l'intervalle, en 4 générations seulement on obtient : (-37, -191) (106, -75) (247, -30) (389, -12) (519, -5)

Ce sont les trois minimums attendus pour $x = L(3 + 4k)/2.nsi$ avec $k \in \mathbb{Z}$, qui se trouvent encadrés par les deux minimums immédiatement situés de part et d'autre des bornes de $[0, L]$, ceux-ci étant obtenus par le jeu des "bruits" affectés à la population des 10 abscisses.

La population initiale est fixée à une quinzaine de chromosomes aléatoires.

III - 2° Critères examinés pour apprécier une trajectoire

But de l'expérience : on cherche à expérimenter un apprentissage par algorithme génétique de l'ensemble des données liées au problème du suivi de ligne par un robot. Le contrôleur va chercher à suivre cette ligne par un ensemble de règles destinées à déterminer la variation de direction ainsi que sa variation de vitesse, en prenant comme arguments la distance à la courbe et la variation de cette distance. Cependant les paramètres tels que la distance maximale, le braquage maximal, l'accélération ou encore l'ouverture des capteurs de distance seront sujets eux aussi à modifications par apprentissage. L'apprentissage n'étant pas supervisé, on peut s'interroger sur le problème de "boot-strap" lié à l'évolution d'une population initialement aléatoire et à celle des transitions aléatoires à propos de la question capitale de l'évaluation des individus de cette population.

Nous avons défini plusieurs critères lors des deux types d'expériences réalisées : suivi d'une ligne et suivi d'un chemin tracé par deux lignes.

a) L'écart moyen tx est la moyenne de la distance non algébrique mesurée à chaque pas suivant un angle d'ouverture "ouv" par rapport à sa direction et propre au robot, moyenne effectuée suivant le nombre de pas. A lui seul ce critère ne peut pas traduire un bon suivi car les pas effectués doivent être variables.

b) L'écart angulaire moyen td mesure l'angle géométrique entre la direction du robot et celle de la courbe à suivre au point de celle-ci où la distance est mesurée. Ces deux derniers indices sont significatifs pris ensemble, mais même nuls il peut y avoir une grande distance entre certains points de la trajectoire réalisée et la ligne à suivre.

c) La longueur du pas moyen tp , nous avons cherché à la maximiser de façon à favoriser les solutions les plus rapides à accomplir leur suivi.

d) L'accélération algébrique moyenne nc : cet indice est la moyenne des écarts de pas en valeur absolue.

On a alors cherché à minimiser une fonction d'évaluation du type $\alpha tx + \beta td + \gamma tp + \delta nc + \epsilon \text{len}(C)$ en faisant intervenir de plus le nombre de règles du chromosome C. Nous remarquons que pour $\gamma = -1$, la valeur $\alpha = 4$ est discriminante, (convergence vers une trajectoire rectiligne si $\alpha < 4$ et vers une trajectoire qui suit assez bien la ligne, mais à petits pas pour $\alpha > 4$). Si $\gamma = -1$ et $\epsilon = 1$, l'importance a donc été donnée à δ de -50 à -70. Les meilleurs résultats tenant en compte l'écart angulaire moyen ont été obtenus pour $\beta = 9$. Cependant, il apparait que l'apprentissage est très lent et peu efficace, c'est pourquoi nous avons remplacé tx par la norme L1, et nc par un indice de tenue de route, puis au lieu de chercher à réduire simultanément une fonction résultant d'une combinaison linéaire de paramètres à minimiser, nous avons cherché à les réduire

successivement en imposant des conditions plus fortes sur la trajectoire suivie.

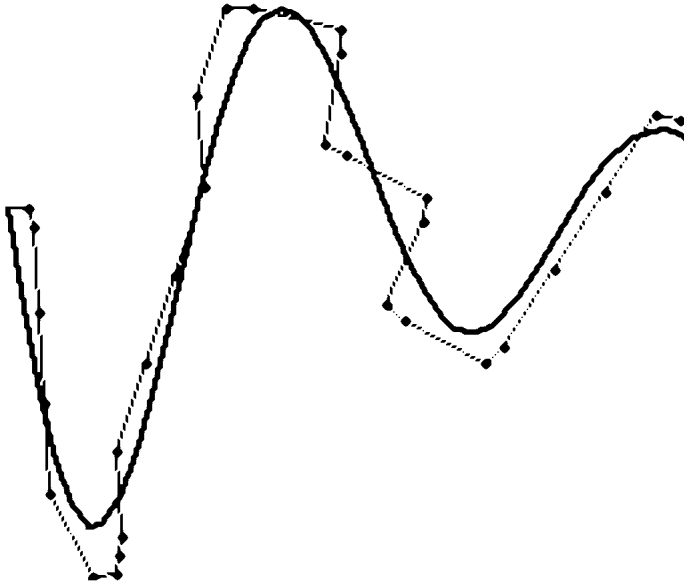


Figure 2 : Exemple de trajectoire où seule l'accélération positive est favorisée. Cette trajectoire est obtenue pour $ouv=60^\circ$, $dm=25$, $vm=85$, $am=50^\circ$, $acc=25$, $pm=35$ et les règles $(-104 -97 89 143 -86)$ $(0 51 102 29 97)$ et leurs symétriques.

Norme Dans la mesure où le robot décrit une trajectoire dont les abscisses sont croissantes, elle représentera une fonction ϕ dont la distance au sens de l'espace L1 avec la courbe à suivre f , donnera un meilleur indicateur.

$$\text{Ecart} = \frac{1}{x} \int_0^x |f - \phi|, \text{ si le robot passe de } (u_0, v_0)$$

à (u_1, v_1) sans couper la courbe, avec $u_0 < u_1$, l'écart est alors :

$$\int_{u_0}^{u_1} |f - \phi| = \left| \frac{1}{2}(u_1 - u_0)(v_0 + v_1) - F(u_0) + F(u_1) \right|$$

Sinon, l'intersection est calculée par dichotomies et on somme les deux écarts.

En 30 générations environ cet écart est ramené entre 5 et 10 (distance qu'il faut comparer avec L et H), mais ce seul indice favorise également une trajectoire en petits pas, c'est pourquoi nous avons cherché à définir :

III - 3° Tenue de route

Si "dp" est la différence algébrique entre deux pas consécutifs et "da" la variation de direction, on peut remarquer que $(1 + dp / acc) / 2$ est un indicateur d'accélération égal à 0 pour le freinage maximal et à 1 pour l'accélération maximale.

Si, maintenant, on regarde $|da| / am$, c'est un indicateur du braquage, minimal pour 0 et maximal pour 1. C'est pourquoi nous définissons la "bonne tenue de route" lorsque leur somme est voisine de 1 et nous chercherons donc à minimiser :

$$tr = \left| \frac{|da|}{am} + \frac{dp}{2acc} - \frac{1}{2} \right| \in [0, 1]$$

La moyenne ramenée en pourcentage, nous servira de fonction d'évaluation pour les trajets entièrement accompli dans un écart fixé. La meilleure valeur de cet indicateur a été 6%.

Ci dessous, on essaie de contraindre $|f - \phi| < 8ds$ par exemple, puis on se porte naturellement vers le suivi d'une route entre deux bordures.



Figure 3 : Un bon résultat avec départ lent, lors d'un apprentissage en difficulté croissante. $ouv=36^\circ$, $dm=9$, $vm=9$, $am=36^\circ$, $pm=27$ et les 5 règles $(24 39 49 -33)$ $(-85 -36 131 51 34)$ $(85 -59 132 -87 -56)$ $(-24 68 108 11 -93)$ $(-16 94 77 -21 44)$ et leurs symétriques.

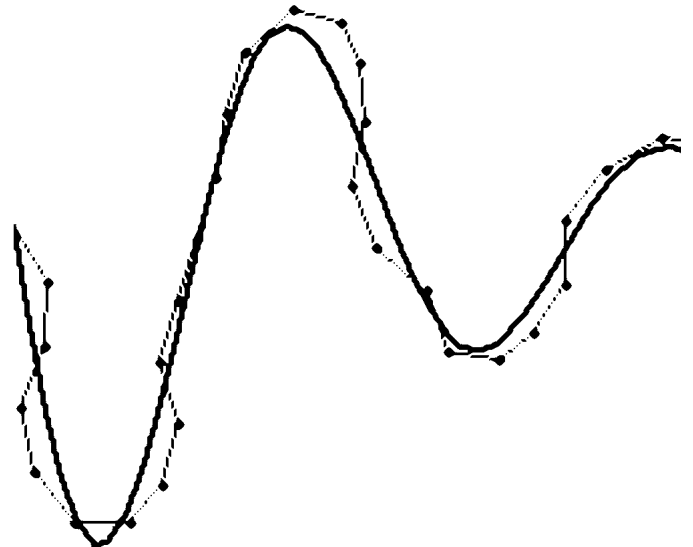


Figure 4 : Pour un départ rapide, apprentissage en difficulté décroissante. Pour $ouv=60^\circ$, $dm=23$, $vm=30$, $am=73$, $acc=11$, $pm=25$ et les règles $(-100 -103 100 93 -80)$ $(-112 -80 100 75 -88)$ $(-74 50 100 30 27)$ $(0 0 100 20 100)$ $(82 -44 94 -83 -31)$ $(67 -83 36 33 15)$ $(86 72 104 -74 15)$ $(-43 95 16 -17 -95)$ tenue de route 13%.

III - 4° Suivi d'une route avec quatre capteurs de distance

Nous disposons d'une route matérialisée par deux bords. Dans un premier temps on utilise comme paramètres d'entrée les distances aux parois mesurées suivant les ouvertures $\pm\pi/2$ et $\pm\pi/4$, puis $\pm\pi/2$ et $\pm\pi/8$.

Cette fois $(x_1, x_2, x_3, x_4) \in [-dm, dm]^4 \rightarrow (da, dv) \in [-am, am] * [-acc, acc]$

Les caractéristiques du robot sont fixées comme suit : vitesse minimale $ds = 8$, vitesse maximale $pm = 30$, accélération maximale $acc = 5$, braquage maximal $am = 50^\circ$, distance maximale de reconnaissance des capteurs $dm =$

35 et enfin $d(f1, f2) = 40$ l'écart entre les deux courbes.

Nous avons calculé la fonction d'évaluation d'une liste de règles C par le parcours restant exprimé en % + tenue de route moyenne en %.

Les jeux de règles sont également symétriques avec : $\text{sym}(x_1, x_2, x_3, x_4, r, \text{al}, \text{ac}) = (x_3, x_4, x_1, x_2, r, -\text{al}, \text{ac})$ Les résultats sont tout à fait analogues à ceux obtenus avec trois capteurs de distance.

III - 5° Trois capteurs

En chaque point, relativement à la direction du véhicule, sont mesurées les distances aux parois suivant des ouvertures respectives $-\pi/3, 0, \pi/3$, le contrôle flou réalise alors une fonction :

$$(x_1, x_2, x_3) \in [-dm, dm]^3 \rightarrow da, dv \in [-am, am]^*[-acc, acc]$$

Nous avons cette fois fixé $dm = 40$ (c'est la distance entre les deux courbes), $pm = 35$, $acc = 10$ et $am = 45^\circ$. Les règles sont du type : (PB PB PB r ZE PB) (ZE PS PS r PS ZE) (PS ZE ZE r PB NB) et leurs symétriques avec $\text{sym}(a, b, c, r, \text{al}, \text{ac}) = (c, b, a, r, -\text{al}, \text{ac})$

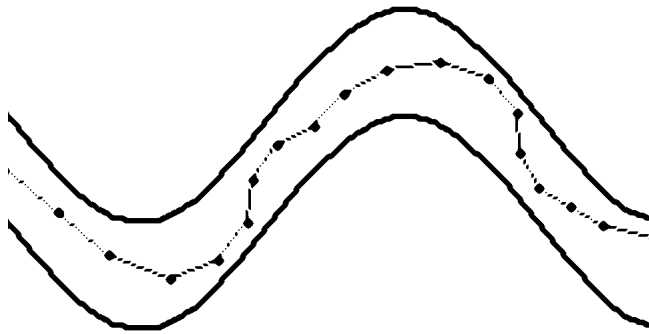


Figure 5 : Tenue de route 9% avec un départ rapide, 6 règles (93 115 24 123 41 32 69) (54 110 54 45 53 -3 47) (79 131 69 4 94 -103 -53). On observera des ralentissements chaque fois que le robot s'est approché trop près des parois.

Nous éprouvons l'une de ces solutions, sur un trajet plus difficile.

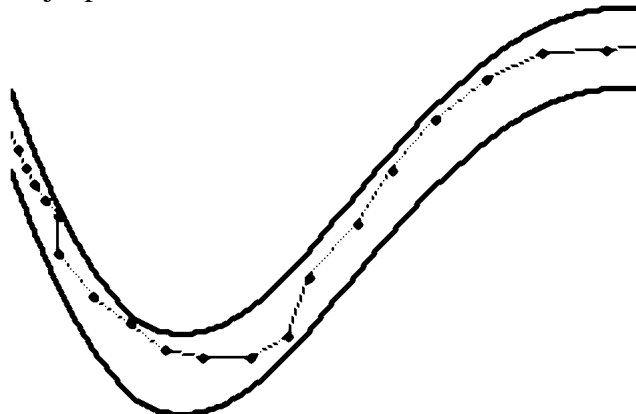


Figure 6 Tenue de route 26% avec les règles (27 104 -19 57 39 -35) (106 59 42 53 -47 -116) (95 54 41 -85 67) (132 3 62 47 -58 -88). On pourra noter dans le détail une explication des changements de direction du robot en sachant qu'il regarde à $\pm\pi/3$ de sa route.

Conclusions et perspectives

Ayant vérifié que l'algorithme de Sugeno du contrôle flou pouvait aisément se généraliser à plusieurs variables simultanément en sortie, grâce à des règles possédant plusieurs conclusions. L'apprentissage par algorithme génétique sur des valeurs numériques s'est révélé évidemment moins satisfaisant et plus lent (plusieurs centaines de générations) que les expériences menées sur des valeurs symboliques. C'est pourquoi, revenir à des valeurs symboliques de prédicats, tout en conservant la généralité sur le nombre d'entrées, de sorties et de règles peut présenter plusieurs avantages : disposer d'un prédicat ANY, donc de règles réalisant des projections sur R^m avec $m < n$, (ce qui peut permettre en outre de différencier les entrées essentielles des entrées secondaires), donner des règles intuitivement plus parlantes et surtout réduire l'espace de recherche.

Références

- Franz Non-linearities in generic adaptive search Doctoral dissertation 1972 in Dissertation Abstracts International 33(11)5240B-5241B University of Michigan
- Holland Une échappatoire à la précarité : les possibilités de l'application des algorithmes généraux d'apprentissage aux systèmes parallèles à base de règles, Apprentissage symbolique pp 523-554 (Cépaduès 1993)
- Gacogne L. Compte rendu sur la recherche de table de contrôleur flou par algorithme génétique, Rapport 93/26 Laforia Université Paris VI
- Gloennec P.Y. Applications des algorithmes génétiques pour l'optimisation des fonctions d'appartenance d'un réseau neuro-flou, Journées sur les applications des ensembles flous, p.219-226 Nîmes 1992
- Grefenstette J.J. Multilevel credit assignment in a genetic learning system, Proceedings of the second conference on genetic algorithms pp 202-209 1987
- Grefenstette J.J. Incremental learning of control strategies with genetic algorithms, Sixth international workshop machine learning p.340-344 1989
- Goldberg David Edward Genetic algorithms in search optimization and machine learning, Addison Wesley 1989
- Kamada H. Yoshida M. A visual control system using image processing and fuzzy system. IEEE round table discussion on vision-based vehicle guidance Tokyo 1990
- Karr Charles L. Design of an adaptive fuzzy logic controller using a genetic algorithm, Fourth International Conf. on genetic algorithm pp 450-457 San Diego 1991
- Kirkpatrick S. Gellat C.D. Vecchi M.P. Optimization by simulated annealing, Sciences 220 p.671-680 1983
- Menger K. Statistical metrics, Proceedings of the national Academy of Sciences USA n°28 p.535-537 1942
- Siler W. Ying H. Fuzzy control theory : the linear case FSS n°33 pp 275 1989
- Sklar A. Schweitzer B. Probabilistic metric spaces, North Holland 1983
- Thrift P. Fuzzy logic synthesis with genetic algorithm, Fourth International Conf. on genetic algorithm p.509-513 San Diego 1991