

UNIVERSITE SAAD DAHLAB DE BLIDA

Faculté des Sciences de l'Ingénieur

Département d'Aéronautique

MEMOIRE DE MAGISTER

Spécialité : Aéronautique

OPTIMISATION DES PARAMETRES D'UN CONTROLEUR NEURO-FLOU PAR LES ALGORITHMES GENETIQUES. APPLICATION A LA COMMANDE DU CAP D'UN BALLON DIRIGEABLE PAR LA VISION.

Par

MELBOUS Abdelkader

Devant le jury composé de :

SALAH BOUKRAA	Professeur, U. de Blida	Président
TAHAR REZOUG	Maître de Conférence, U. de Blida	Examineur
HASSEN SALHI	Maître de Conférence, U. de Blida	Examineur
ABDERREZAK GUESSOUM	Professeur, U. de Blida	Rapporteur
M'HAMED HADJ SADOK	Chargé de Cours, U. de Blida	Co-rapporteur

Blida, Octobre 2006

RESUME

Dans ce travail, on présente la commande d'un ballon dirigeable par la vision, on développe une méthodologie de contrôle qui permet au dirigeable d'accomplir une tâche de prospection d'un environnement tel que le suivi d'une trajectoire observée par une caméra, par l'introduction des paramètres visuels dans la boucle de commande ; la structure de la commande proposée est basée sur un contrôleur par logique floue pour la commande du cap. Les meilleurs paramètres du contrôleur sont choisis par les méta-heuristiques tel que les algorithmes génétiques, réseaux de neurones.

ABSTRACT

In this work, we present the control of a blimp by the vision; we develop a methodology of control that allows the dirigible to accomplish a stain of prospecting of an environment as the follow-up of a trajectory observed by a camera, by the introduction of the visual parameters in feedback of control. The structure of the control proposed is based on a controller by fuzzy logic for the yaw. The controller's best parameter is chosen by the méta-heuristic as the genetic algorithms, networks of neurons.

ملخص

نقوم من خلال هذا العمل بعرض قيادة منطاد باستعمال كاميرا تطور طريقة قيادة التي تسمح للمنطاد بقيام بمهمة مراقبة المحيط مثل إتباع مسار مصور بكاميرا بإدخال معلومات النظرية المشتقة من الكاميرا في دارة المراقب. تصميم المراقب المقترح يعتمد على المنطق الغامض لقيادة زاوية التوجيه. نختار أفضل عوامل المراقب باستعمال الخوارزميات الوراثة و استغلال خواص الشبكات العصبونية..

REMERCIEMENT

Je tiens à remercier Monsieur S. BOUKRAA, Professeur à U. de Blida et directeur de département d'aéronautique, qui m'a fait l'honneur de présider ce jury.

Je remercie également H.SALHI , Maître de Conférence, U. de Blida d'avoir accepté de juger ce travail

Je tiens à remercier particulièrement Monsieur T. REZOUG, Maître de conférence au département d'Aéronautique de l'Université de Blida pour son aide précieuse à l'accomplissement de ce mémoire et faisant partie de jury de ma soutenance.

Il m'est difficile de dissocier dans ces remerciements Monsieur le professeur A.Guessoum et Monsieur M.Hadj Sadok chargé de cours ,d'avoir proposé un sujet de thèse passionnant et n'ont pas épargné leurs efforts pour m'aider tout au long de cette thèse. Je leur dois beaucoup et même davantage.

Je remercie également tous les enseignants de la post graduation du département d'aéronautique de l'université de Blida.

J'adresse mes amitiés à tous mes collègues du département de mécanique, je les remercie de leur soutien et leur aide pendant la préparation de cette thèse.

En dernier, je remercie chaleureusement mon collègue D. Amalou chargé de cours de l'université de Blida pour son aide efficace.

TABLE DES MATIERES

RESUME	
REMERCIEMENT	
TABLE DES MATIERES	
LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX	
INTRODUCTION	
1. MODELISATION MATHEMATIQUE DU BALLON DIRIGEABLE	13
1.1. Description du ballon dirigeable	13
1.2. Modélisation cinématique.	14
1.3. Modélisation dynamique.	18
1.4. Synthèse de la dynamique du dirigeable	27
2. ASSERVISSEMENT VISUEL	28
2.1. Introduction	28
2.2. Les différents types de commandes par la vision	28
2.3. Matrice d'interaction	31
2.4. Cinématique de la caméra	32
2.5. Modélisation de la camera	33
2.6. Calcul de la matrice d'interaction d'un point	36
3. COMMANDE FLOU ET NEURO-FLOU	38
3.1. Introduction	38
3.2. Sous ensemble flous	39
3.3. Raisonnement en logique flou	41
3.4. Commande flou	45
3.5. Commande hybride neuro-flous	52
4. ALGORITHMES GENETIQUES	57
4.1. Introduction	57
4.2. Principe de l'algorithme génétique	58
4.3. Mécanisme de sélection	60
4.4. Croisement	61
4.5. Mutation	62

5. SYNTHESE DES COMMANDES ET SIMULATION	63
5.1. Les équations décrivant la dynamique du vol	63
5.2. Stratégie de vol du dirigeable	64
5.3. Simulation de la navigation	65
5.4. Synthèse d'un Contrôleur flou pour la commande du cap	66
5.5. Synthèse d'un contrôleur flou par Algorithme Génétique	74
5.6. Synthèse d'un contrôleur flou par les algorithmes génétiques hiérarchiques.	85
5.7. Synthèse d'un contrôleur neuro-flou par l'algorithme génétique	94
CONCLUSION	100
APPENDICE A	103
APPENDICE B	106
APPENDICE C	111
REFERENCES	113

LISTE DES ILLUSTRATIONS, GRAHIQUES ET TABLEAUX

Figure 1.1	Dimension du dirigeable et position des propulseurs	13
Figure 1.2	Représentation des repères	14
Figure 1.3	Vue du profil ; rotation autour de l'axe y	15
Figure 1.4	Vue de l'avant ; le cap, rotation autour de l'axe z	15
Figure 1.5	Vue de dessus rotation autour de l'axe x, angle de roulis	16
Figure 1.6	Représentation en 3ddl d'un corps rigide	18
Figure 2.1	Structure Position-Based Servoing	29
Figure 2.2	Structure images Based servoing	31
Figure 2.3	Position du repère de la camera	32
Figure 2.4	Projection d'un point dans le plan image	33
Figure 2.5	Modèle de la caméra	36
Figure 3.1	Représentation d'un sous ensemble flou	40
Figure 3.2	Variable linguistique décrivant la température	42
Figure 3.3	Valeur de vérité d'un terme linguistique	43
Figure 3.4	Configuration générale d'un contrôleur flou	46
Figure 3.5	L'inférence floue mini	48
Figure 3.6	Inférence floue prod	48
Figure 3.7	fonctions d'appartenance des termes linguistiques	49
Figure3.8	Résultat d'inférence	50
Figure 3.9	Neurone artificiel	52
Figure 3.10	Réseaux de neurones non bouclés	53
Figure 3.11	Réseaux de neurones bouclés	54
Figure 3.12	Réseau de neurones à multicouche	55
Figure 3.13	Configuration d'un réseau de neuro-flou multicouche	56
Figure 4.1	Structure d'un algorithme génétique	58
Figure 4.2	Roulette proportionnel	60
Figure 4.3	Croisement en un point	61
Figure 4.4	Croisement multipoints	62
Figure 4.5	Principe de mutation	62

Figure 5.1	Synoptique du mode longitudinal	63
Figure 5.2	Synoptique du mode latéral	64
Figure 5.3	Le synoptique de commande de navigation (architecture de contrôle)	65
Figure 5.4	Synoptique d'un contrôleur flou pour la commande du cap	66
Figure 5.5	Trajectoire à suivre	66
Figure 5.6	Fenêtre de Blackman	67
Figure 5.7	Synoptique de la commande du cap par la vision	68
Figure 5.8	L'apparition du repère sur le plan image	69
Figure 5.9	Contrôleur flou adopté	70
Figure 5.10	Fonctions d'appartenance du contrôleur	71
Figure 5.11	Variation du cap en fonction du temps	72
Figure 5.12	Le suivi de la trajectoire	72
Figure 5.13	Navigation du dirigeable le long d'une courbe fermée	73
Figure 5.14	Forme de la fonction d'appartenance triangulaire	75
Figure 5.15	Forme de la fonction d'appartenance trapézoïdale	75
Figure 5.16	Forme de la fonction d'appartenance gaussienne	76
Figure 5.17	Exemple d'un chromosome codé en base cinq	77
Figure 5.18	Structure du chromosome codé en base sept	78
Figure 5.19	Représentation des piques et des ouvertures des fonctions d'appartenance	79
Figure 5.20	Structure de l'algorithme génétique	81
Figure 5.21	Les fonctions d'appartenance	82
Figure 5.22	Variation du cap	83
Figure 5.23	Navigation du dirigeable	83
Figure 5.24	Variation de la fonction d'adaptation en fonction du nombre de générations	84
Figure 5.25	Chromosome de l'algorithme génétique hiérarchique	85
Figure 5.26	Structure de l'algorithme génétique hiérarchique	86
Figure 5.27	Structure du chromosome adopté	87
Figure 5.28a	Univers de discours avec des termes linguistique invalides	87
Figure 5.28b	Procédure de recouvrement des fonctions d'appartenance	88
Figure 5.29	Fonctions d'appartenance du contrôleur optimisé par les HGA	89
Figure 5.30	Variation du cap obtenue par les HGA	90

Figure 5.31	Navigation du CF optimisé par les HGA	90
Figure 5.32a	Séquences d'images obtenues	91
Figure 5.32b	Séquences d'images obtenues	92
Figure 5.33	Superposition des images	93
Figure 5.34	Structure d'un réseau neuro-flou à cinq couches	94
Figure 5.35	Forme du chromosome du contrôleur neuro-flou	97
Figure 5.36	Les fonctions d'appartenance des entrées et de sortie du réseau neuro- flou	98
Figure 5.37	Variation du cap en fonction du temps obtenu par réseau neuro- flou	98
Figure 5.38	Navigation obtenue par réseau neuro-flou	99
Tableau 3.1	Les T-normes et T-conormes les plus utilisées	41
Tableau 3.2	Implications floues les plus utilisées	45
Tableau 5.1	Base des règles	71
Tableau 5.2	Tableau des conclusions	77
Tableau 5.3	Base des règles codées en base 5	77
Tableau 5.4	Tableau de la base des règles	82

INTRODUCTION

Le développement réalisé durant ces dernières années dans les domaines de la technologie des capteurs de vision et du traitement d'image a contribué au retour de l'utilisation des ballons dirigeables dans des applications civiles et militaires. Dotés d'un système embarqué, les dirigeables assurent différentes fonctions comme : mission de prospection, surveillance, étude environnementale et recherche climatique,[21],[22]. Par rapport aux autres drones (UAV), les dirigeables ont un très faible coût de se maintenir en vol et ont la possibilité de voler à basse altitude et à faible vitesse.

Commander un dirigeable, c'est lui permettre d'atteindre une configuration désirée ou de réaliser une tâche dans un environnement réel. Cette opération est très délicate, à cause des perturbations dues à l'environnement, et à la structure complexe du dirigeable menant à des équations dynamiques non linéaire fortement couplé avec des paramètres dépendant et souvent inconnus.

Au premier abord de la littérature on peut constater que la modélisation dynamique et cinématique du dirigeable, ainsi que la planification de trajectoire et génération de mouvement sont parmi les thèmes qui ont attiré l'attention des chercheurs en robotique et aboutissant à des résultats intéressants [5][6][7]

Les auteurs décrivent le ballon dirigeable comme une enveloppe de forme ellipsoïde remplie de gaz d'hélium, muni de propulseurs vectorisés pour assurer le contrôle de la force de poussée et de direction, il est doté d'une plate forme permettant d'emporter le système embarqué de contrôle.

Deux types de dirigeables sont utilisés aux laboratoires servant comme application des algorithmes de contrôle (Appendice B)

- des petits ballons utilisés à l'intérieur du laboratoire.
- des grands ballons utilisés à l'extérieur de dimensions réelles

L'approche adoptée par la plupart des chercheurs dans le contrôle des dirigeables consiste à considérer chaque condition de vol comme un problème à part, ce qui amène à découpler le système en deux modes ; mode longitudinal, vol dans le plan vertical et le mode latéral, vol dans le plan horizontal [8] [20]

L'étude du système pour les manœuvres proche du vol plané, est très abordée dans la littérature [1] [2] [3] [4], elle a suscité beaucoup d'intérêt à cause des missions réalisées par le dirigeable tel que le vol stationnaire et le suivi d'une trajectoire.

Dans ce contexte des travaux ont été menés dans le laboratoire des systèmes complexes université d'Evry France sur le thème de la planification du chemin et génération de mouvement [2]. Une étude de la génération d'une trajectoire équilibrante a été réalisée où le ballon se déplace dans le plan horizontal. En première partie la génération de la trajectoire à spirale cubique a été générée; et un algorithme de commande non linéaire basé sur la linéarisation entrée-sortie pour le suivi de la trajectoire a été introduit pour le cas d'un vol à altitude constante. Une simulation sous Matlab a été élaborée pour le déplacement du dirigeable entre deux points où le chemin optimal a été déduit par les algorithmes génétiques, une fois la trajectoire à suivre est connue, la commande non linéaire par feedback a été appliquée au ballon dirigeable. On constate que le contrôle est basé sur la position et non pas sur l'orientation, le dirigeable avance et tourne sur lui-même. D'autres auteurs ont suivi une méthodologie similaire, avec l'introduction d'autres critères d'optimisation tel que le temps nécessaire à l'accomplissement d'une mission ou l'énergie fournie ; le problème consiste à trouver les solutions admissibles pour les commandes qui minimisent une fonction de coût pour amener le système d'un état initial vers l'état final. [9][29]

L'utilisation des paramètres visuels, extraits de la camera, dans la boucle de commande des robots a suscité un intérêt croissant, chez les chercheurs en robotique (option UAV), pour le développement des systèmes intelligents autonomes.

L'approche de la commande par la vision ou l'asservissement visuel a été abordée par François Chaumette [11][12] et Peter Corke [13].

La plupart des recherches réalisées dans ce domaine sont orientées sur deux structures de commande :-Position based servoing - Asservissement visuel [14].

Dans ce contexte, en Pennsylvanie une étude de la commande dynamique du mouvement du dirigeable par la vision a été réalisée au laboratoire GRASP[10] [29]; Dans ces travaux ,on a d'abord adopté la méthodologie utilisée dans l'étude des robots mobiles autonomes ,planification de la trajectoire et génération de mouvement dans l'espace cartésien ,pour un vol plané, en essayant de planifier le chemin optimal , et en minimisant l'énergie consommée entre deux points .Ensuite on a injecté la dynamique du dirigeable dans l'image tout en reprenant l'étude de la planification du chemin et génération du mouvement dans l'espace du plan image, en introduisant les informations visuelles dans la boucle de commande. Enfin pour éviter le problème d'homographie de la camera où les cercles se transforment en ellipses, le plan d'image doit être parallèle au plan de l'objet. Ce dernier est un petit ballon rouge se trouvant à une certaine distance, le modèle du ballon et la camera sont mis sous forme d'une équation mathématique permettant d'appliquer la commande basée sur la technique de linéarisation entrée-sortie de type PID. Des résultats satisfaisants ont été obtenus comparativement avec une simulation expérimentale.

A l'institut supérieur technique de l'université de Lisbonne des travaux ont abordé le problème de l'estimation de la position du dirigeable en tenant compte de l'homographie et de la modélisation de la caméra [27]. La réalisation de cette application est effectuée en deux parties. Tout d'abord la modélisation cinématique et dynamique du dirigeable a été traité où le modèle dynamique est mis sous la forme matricielle puis linéarisé et découplé en deux modes de vol, ensuite les auteurs ont procédé à la modélisation de la camera et à l'étude de l'homographie ;la navigation du dirigeable a été simulée et validée par une expérimentation où le ballon du laboratoire évolue dans le plan horizontal à une altitude constante au dessus d'une maquette simulant une ville observée à partir d'une certaine hauteur.

Dans cette simulation une trajectoire de forme d'une spirale cubique est générée et prise comme une trajectoire de référence à suivre. Les primitives extraites de la camera sont utilisées par l'algorithme d'estimation de la position et un filtre de Kalman est introduit pour minimiser les erreurs de traitement et d'estimation. Deux techniques de commande robuste (Linear Control(LQR) ; Non-linear Control (Robust Sliding Mode Control) ont été testées pour le suivi de la trajectoire dans le plan horizontal ;dans le mode longitudinal une

autre commande de type PID est appliquée pour maintenir le dirigeable à une altitude constante. Dans cette navigation, la trajectoire est subdivisée en plusieurs points entre le point de départ et le point d'arrivée, cette technique s'appelle Point-to-Point Control tirée de la navigation de robots mobiles[23]. Les résultats obtenus de cette étude ont permis l'élaboration d'un logiciel de simulation sous Matlab dénommé : BlimpSim

Au Brésil, au laboratoire de l'université de Compinas des recherches ont été menées en asservissement visuel et sont concrétisés par des articles et des thèses [15][16]. Parmi ces chercheurs on cite G.Silveira, l'auteur de nombreux articles relatifs à l'asservissement visuel des ballons dirigeables [17],[46] ; parmi ses travaux, l'étude de l'asservissement visuel par des primitives de droites simulant les lignes blanches continues des bords de la route ou des rivières, a été réalisée en se basant sur les travaux de Chaumette [11],[12]. Le principe de commande consiste à asservir le dirigeable dans le plan d'image par des primitives de droites. Une commande robuste de type LQG est adoptée, le résultat est mis en évidence par une séquence d'images où la commande tente de positionner la route au milieu de l'image.

A travers cette synthèse nous constatons que les travaux concernant la commande des dirigeables sont abordés par plusieurs auteurs. Dans ces travaux l'étude des commandes de type PID, LQG et LQR est la plus fréquente contrairement à la commande par logique floue. L'objectif de notre travail est l'application de la logique floue pour l'élaboration d'un contrôleur flou, un contrôleur neuro-flou et l'amélioration de ces performances par les algorithmes génétiques. Pour cela on réalise une simulation qui nécessite un modèle dynamique ; pour cette application on a choisi le dirigeable dénommé videoblimp utilisé en [27] dont les paramètres sont mentionnés en annexe.

Le plan suivant est adopté :

- Modélisation cinématique et dynamique du dirigeable.
- Asservissement visuel.
- Commande en logique floue et neuro-flou.
- Synthèse d'une commande du cap d'un ballon dirigeable et l'optimisation des paramètres du contrôleur par l'algorithme génétique et réseau de neurones.

CHAPITRE 1

MODELISATION MATHEMATIQUE DU BALLON DIRIGEABLE

1.1. Description du ballon dirigeable [27][29]

Le ballon dirigeable est une enveloppe de mylar de forme ellipsoïde remplie de gaz d'hélium, muni de propulseurs vectorisés pour assurer le contrôle de la force de poussée et de direction ,il est doté d'une plate forme permettant d'emporter le système embarqué de contrôle.

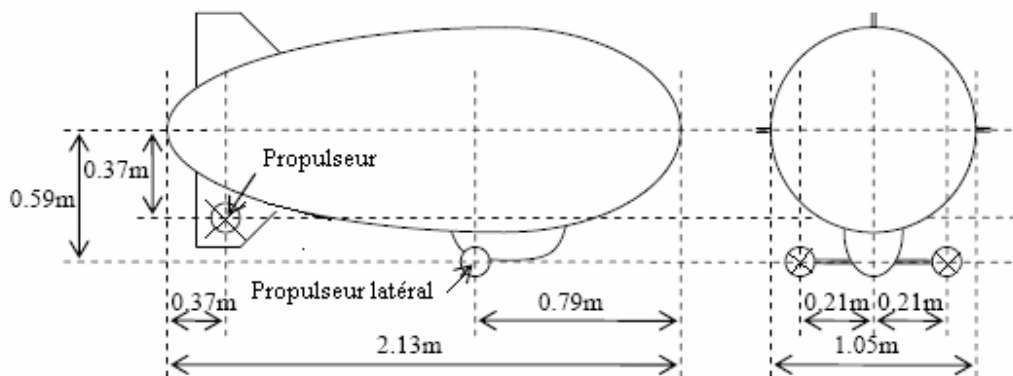


Figure 1.1 : Dimension du dirigeable et position des propulseurs [27]

1. 2. Modélisation cinématique [42]

L'étude et l'analyse du mouvement d'un ballon dirigeable se fait dans deux repères, un repère lié au ballon, l'autre lié à la terre que l'on appelle repère référentiel selon la formulation de Fossen[32] :

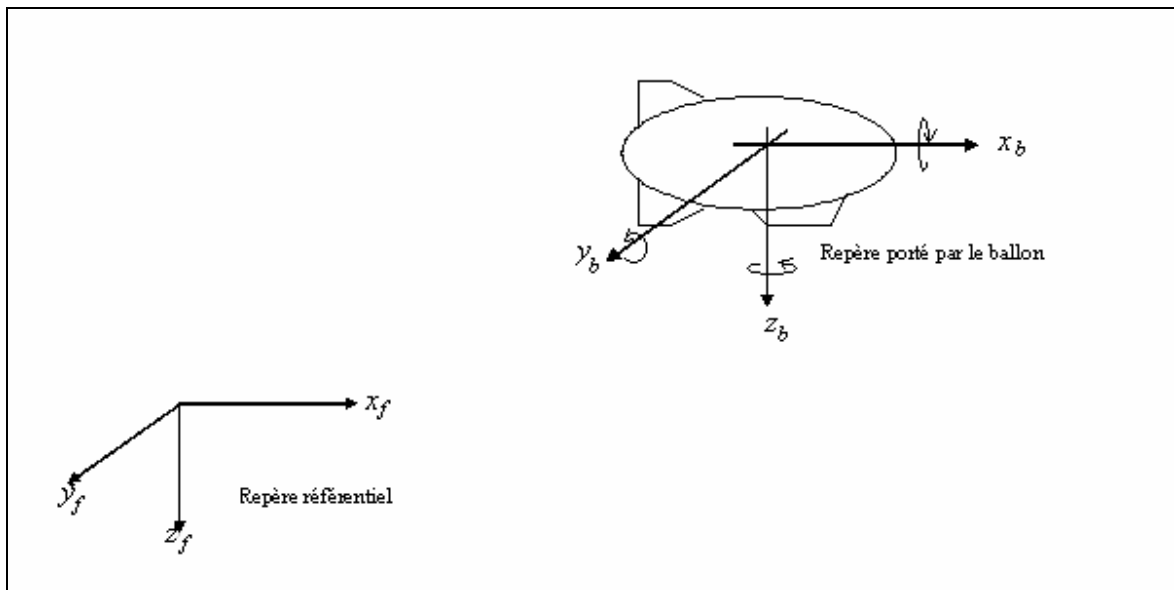


Figure 1.2 : Représentation des repères

La position et l'orientation du ballon sont définies dans le repère référentiel où la rotation de la terre est négligée ; sa surface est supposée plane.

Ils sont exprimés par le vecteur $\boldsymbol{\eta} = [\boldsymbol{\eta}_1, \boldsymbol{\eta}_2]^T$

Avec $\boldsymbol{\eta}_1 = [x \ y \ z]^T$ représente les trois composantes de position

$\boldsymbol{\eta}_2 = [\phi \ \theta \ \psi]^T$ représente les trois composantes d'orientation que l'on appelle respectivement angle de roulis, de tangage et de lacet

Les vitesses linéaires et angulaires seront définies dans le repère lié au ballon par le vecteur $\boldsymbol{v} = [\boldsymbol{v}_1, \boldsymbol{v}_2]^T$

où $\boldsymbol{v}_1 = [v_x \ v_y \ v_z]^T$ représente les trois composantes du vecteur vitesses linéaires.

$\boldsymbol{v}_2 = [\omega_x \ \omega_y \ \omega_z]^T$ représente les vitesses angulaires autour des trois axes du repère lié au ballon.

L'étude de l'orientation du ballon dirigeable utilise la formulation des angles d'Euler où le passage d'une configuration angulaire à une autre se fait par la décomposition de la matrice de passage, comme indiquée ci-dessous :

-Rotation autour de l'axe y

$$\mathbf{R}(y, \theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad (1.1)$$

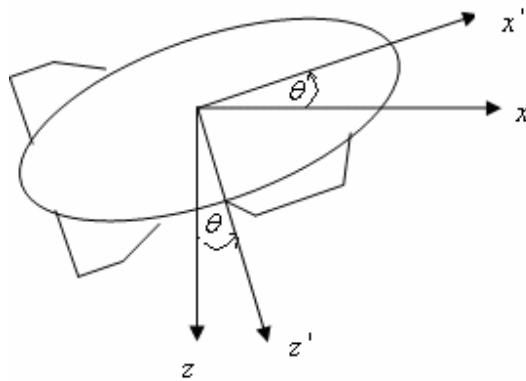


Figure 1.3 : Vue du profil ; rotation autour de l'axe y (angle de tangage)

-Rotation autour de l'axe x :

$$\mathbf{R}(x, \phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \quad (1.2)$$

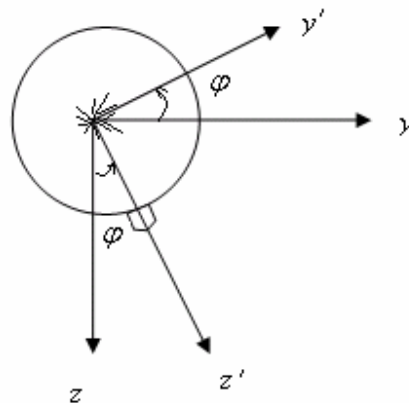


Figure 1.4 : Vue de l'avant, rotation autour de l'axe x, angle de roulis

-Rotation autour de l'axe z :

$$\mathbf{R}(z, \psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.3)$$

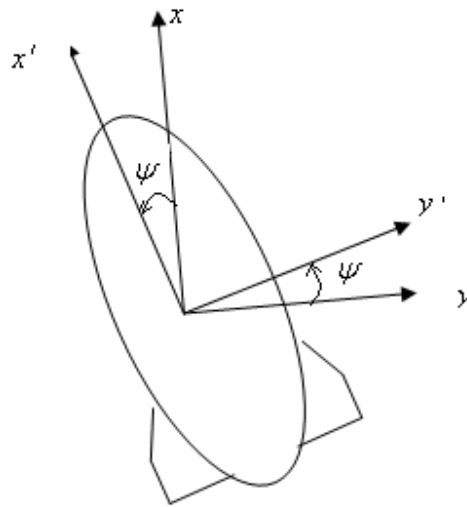


Figure 1.5: Vue de dessus ; le cap, rotation autour de l'axe z (angle de lacet)

La matrice de passage est donnée par l'expression :

$$\mathbf{R}(\varphi, \theta, \psi) = \mathbf{R}(y, \theta) \cdot \mathbf{R}(x, \varphi) \cdot \mathbf{R}(z, \psi)$$

$$\mathbf{R}(\varphi, \theta, \psi) = \begin{bmatrix} \cos \psi \cos \theta & -\sin \psi \cos \phi + \cos \psi \sin \theta \sin \phi & \sin \psi \sin \phi + \cos \phi \cos \psi \sin \theta \\ \sin \psi \cos \theta & \cos \psi \cos \phi + \sin \phi \sin \theta \sin \psi & -\cos \psi \sin \phi + \sin \theta \sin \psi \cos \phi \\ -\sin \phi & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix} \quad (1.5)$$

1.2.1 Transformation des vitesses linéaires

La relation des vitesses linéaires dans les deux repères est donnée par :

$$\dot{\eta} = J_1(\eta_2) \nu_1 \quad (1.5)$$

Cette relation décrit la position du ballon dirigeable dans le repère inertiel où $J_1(\eta_2) = R(\varphi, \theta, \psi)$ représente la matrice de passage du repère inertiel au repère porté par le ballon dirigeable, c'est une matrice de rotation orthogonale, de déterminant unité ayant pour inverse sa transposée

1.2.2. Transformation des vitesses angulaires

La relation liant les vitesses angulaires dans les deux repères est de la forme suivante :

$$\dot{\eta}_2 = J_2(\eta_2) \nu_2 \quad (1.6)$$

où

$$J_2(\eta_2) = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix}$$

Avec $\theta \neq \pi/2 + k\pi$ (1.7)

En résumé le modèle cinématique du ballon dirigeable est représenté par la relation suivant :

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} [J_1(\eta_2)] & [0_{3 \times 3}] \\ [0_{3 \times 3}] & [J_2(\eta_2)] \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ w_x \\ w_y \\ w_z \end{bmatrix} \quad (1.8)$$

Cette représentation a une singularité pour $\theta=90$.

L'intégration de l'équation $\dot{\eta} = J(\eta) \nu$ nous donne la position et l'orientation du véhicule volant à chaque instant dans le repère inertiel avec $\eta = [x \ y \ z \ \phi \ \theta \ \psi]^T$;

$$\nu = [v_x \ v_y \ v_z \ w_x \ w_y \ w_z]^T$$

$$J(\eta) = \begin{bmatrix} [J_1(\eta_2)] & [0_{3 \times 3}] \\ [0_{3 \times 3}] & [J_2(\eta_2)] \end{bmatrix} \quad (1.9)$$

1.3 Modélisation dynamique

Les équations qui décrivent la dynamique du ballon dirigeable sont déduites des lois de Newton de la mécanique classique et des lois de Kirchoff relatives à la mécanique des fluides. Pour obtenir la dynamique du système, on applique les principes du corps rigide à 6 ddl par rapport au repère référentiel [28],[30] :

Soit un corps rigide de forme quelconque avec comme centre de gravité le point G , le centre du repère lié à ce corps est le point O figure 1.6.

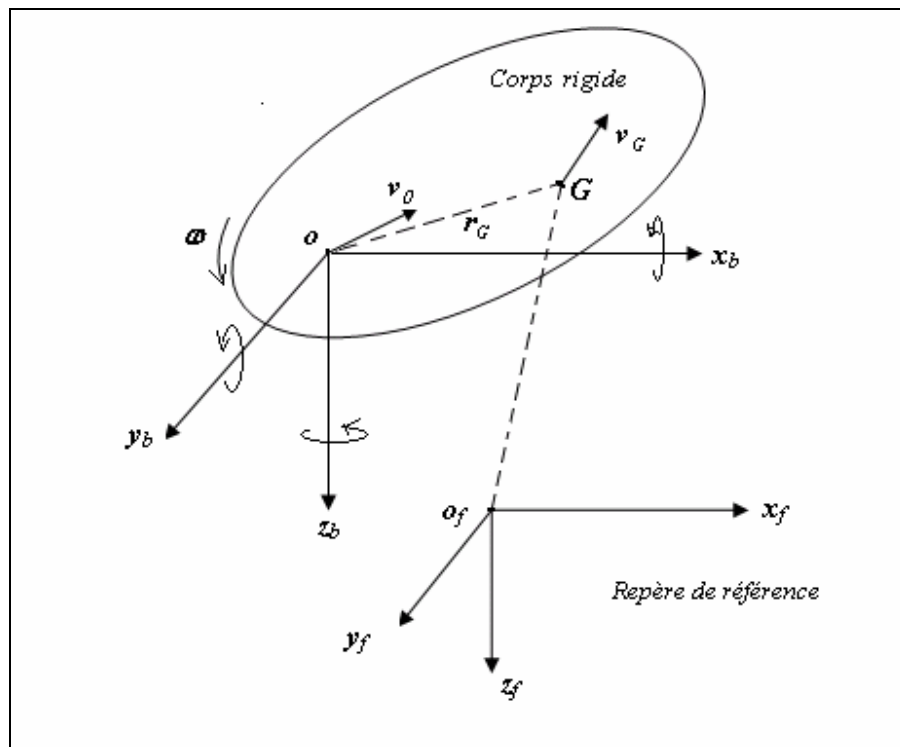


Figure 1.6 : Représentation en 3ddl d'un corps rigide

Le principe de la dynamique du solide décrit l'ensemble des équations suivantes :

La vitesse et l'accélération du corps sont données par :

$$\mathbf{v}_G = \mathbf{v}_0 + \boldsymbol{\omega} \wedge \mathbf{r}_G \quad (1.10)$$

$$\mathbf{a}_G = \mathbf{a}_0 + \boldsymbol{\omega} \wedge \mathbf{v}_0 + \dot{\boldsymbol{\omega}} \wedge \mathbf{r}_G + \boldsymbol{\omega} \wedge (\boldsymbol{\omega} \wedge \mathbf{r}_G) \quad (1.11)$$

Le moment cinétique par rapport à o est donné par :

$$\mathbf{H} = \mathbf{I}_0 \cdot \boldsymbol{\omega} + m \mathbf{r}_G \wedge \mathbf{v}_0 \quad (1.12)$$

D'après la loi de la dynamique la force et le couple sont données par :

$$m \mathbf{a}_G = \mathbf{f}_0 \text{ et } \frac{d\mathbf{H}}{dt} = \mathbf{M}_0 \quad (1.13)$$

Alors, l'équation du mouvement de translation est

$$m(\mathbf{a}_0 + \boldsymbol{\omega} \wedge \mathbf{v}_0 + \dot{\boldsymbol{\omega}} \wedge \mathbf{r}_G + \boldsymbol{\omega} \wedge (\boldsymbol{\omega} \wedge \mathbf{r}_G)) = \mathbf{f}_0 \quad (1.14)$$

Et l'équation du mouvement de rotation est

$$\mathbf{I}_0 \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \wedge (\mathbf{I}_0 \boldsymbol{\omega}) + m \mathbf{r}_G \wedge (\mathbf{v}_0 + \boldsymbol{\omega} \wedge \mathbf{r}_G) = \mathbf{M}_0 \quad (1.15)$$

Où

$$\begin{aligned} \mathbf{f}_0 = \mathbf{F}_1 &= [X \quad Y \quad Z]^T && \text{Forces externes} \\ \mathbf{M}_0 = \mathbf{F}_2 &= [K \quad M \quad N]^T && \text{Moments externes} \\ \mathbf{v}_0 = \mathbf{v}_1 &= [v_x \quad v_y \quad v_z]^T && \text{Vitesses linéaires} \\ \boldsymbol{\omega} = \mathbf{v}_2 &= [\omega_x \quad \omega_y \quad \omega_z]^T && \text{Vitesses angulaires} \\ \mathbf{r}_G &= [x_G \quad y_G \quad z_G]^T && \text{Centre de gravité} \end{aligned}$$

\mathbf{I}_0 est la matrice d'inertie du solide

$$\mathbf{I}_0 = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{xy} & I_{zz} \end{bmatrix} \text{ où } \mathbf{I}_0 = \mathbf{I}_0^T > 0 \quad (1.16)$$

Si le solide a un plan de symétrie

$$\mathbf{I}_0 = \text{diag}[I_{xx} \quad I_{yy} \quad I_{zz}] \quad (1.17)$$

Pour l'étude de la dynamique du ballon dirigeable, on se base sur les hypothèses suivantes :

- la masse et le volume du ballon sont des grandeurs invariables
- les phénomènes d'aéroélasticités sont négligés
- les phénomènes liés au gaz d'hélium sont négligés
- le centre de poussée est supposé au centre du volume

- le ballon comporte des plans de symétries, le centre de gravité et le centre de flottabilité se trouvent dans l'intersection de ces deux plans
- le centre de gravité coïncide avec le centre du repère lié au ballon ($r_G=0$)
- le nombre de Mach est faible ($<3m/s$) et le nombre de Réynolds est faible ($\sim 10^4$)
- la viscosité du fluide (de l'air) est supposée constante
- le centre de flottabilité est au dessus de centre de gravité

En tenant compte de ces hypothèses les expressions (1.14) et (1.15) deviennent

$$m(\dot{\mathbf{v}}_0 + \boldsymbol{\omega} \wedge \mathbf{v}_0) = \mathbf{f}_0 \quad (1.18)$$

$$I_0 \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \wedge (I_0 \boldsymbol{\omega}) = \mathbf{M}_0 \quad (1.19)$$

En projetant ces deux équations sur les différents axes du repère lié au dirigeable on obtient :

$$\begin{aligned} m \left[\dot{v}_x - v_y \omega_z + v_z \omega_y \right] &= X \\ m \left[\dot{v}_y - v_z \omega_x + v_x \omega_z \right] &= Y \\ m \left[\dot{v}_z - v_x \omega_y + v_y \omega_x \right] &= Z \\ I_{xx} \dot{\omega}_x + (I_{zz} - I_{yy}) \omega_y \omega_z &= K \\ I_{yy} \dot{\omega}_y + (I_{xx} - I_{zz}) \omega_z \omega_x &= M \\ I_{zz} \dot{\omega}_z + (I_{yy} - I_{xx}) \omega_x \omega_y &= N \end{aligned} \quad (1.20)$$

Ces équations (1.20) peuvent se mettre sous la forme matricielle selon la formulation de Fossen[30]

$$M_{RB} \dot{\boldsymbol{\nu}} + C_{RB}(\boldsymbol{\nu}) \boldsymbol{\nu} = \boldsymbol{\Gamma}_{RB} \quad (1.21)$$

Avec $\boldsymbol{\nu} = [v_x \quad v_y \quad v_z \quad w_x \quad w_y \quad w_z]^T$

M_{RB} : Matrice d'inertie du système, c'est une matrice symétrique qui s'exprime ainsi

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{bmatrix} \quad (1.22)$$

m : masse du ballon

C_{RB} est la matrice des forces de Coriolis et des forces centrifuges qui s'appliquent au ballon dirigeable. C'est une matrice anti-symétrique de dimension 6x6 et d'expression :

$$C_{RB} = \begin{bmatrix} 0 & 0 & 0 & 0 & mv_x & -mv_y \\ 0 & 0 & 0 & -mv_z & 0 & mv_x \\ 0 & 0 & 0 & mv_y & mv_x & 0 \\ 0 & mv_z & -mv_y & 0 & I_{zz}\omega_z & -I_{yy}\omega_y \\ -mv_z & 0 & mv_x & -I_{zz}\omega_z & 0 & I_{xx}\omega_x \\ mv_y & -mv_x & 0 & I_{yy}\omega_y & -I_{xx}\omega_x & 0 \end{bmatrix} \quad (1.23)$$

1.3.1. Les efforts hydrodynamiques [51][52]

Le dirigeable est considéré comme un corps submergé dans un fluide, lors de son déplacement il apparaît des efforts hydrodynamiques, ils sont dus aux échanges d'énergie cinétique entre le ballon et l'air environnant. Les efforts s'appliquent au centre de poussée ou de flottabilité, ils dépendent de la densité du fluide et de la géométrie du mobile, on suppose que le dirigeable est submergé dans un fluide parfait, volume d'air infini, incompressible, non visqueux irrotationnel et à l'état de repos à l'infini.

Soit (x,y,z) les coordonnées d'un point situé sur le véhicule ou dans le fluide, correspondant aux système d'axe lié au véhicule ,on suppose qu'on peut caractériser l'écoulement du fluide par une fonction scalaire $\phi(x,y,z)$ appelée potentiel des vitesses, définie par

$$\mathbf{u} = -\nabla\phi = \left(-\frac{\partial\phi}{\partial x}, -\frac{\partial\phi}{\partial y}, -\frac{\partial\phi}{\partial z} \right). \quad (1.24)$$

Où \mathbf{u} est la vitesse locale du fluide

Comme le fluide est incompressible l'équation de continuité devient :

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} + \frac{\partial^2\phi}{\partial z^2} = 0 \quad (1.25)$$

Soit \mathbf{n} vecteur normal à la surface du dirigeable orienté vers le fluide, alors, en tout point de la surface on a :

$$-\nabla\phi \cdot \mathbf{n} = v_n \quad (1.26)$$

Où v_n est la composante de la vitesse de la surface au point (x, y, z) dans la direction de \mathbf{n}

Comme le fluide est supposé au repos à l'infini on a

$$\frac{\partial\phi}{\partial x} = 0, \quad \frac{\partial\phi}{\partial y} = 0, \quad \frac{\partial\phi}{\partial z} = 0 \quad (1.27)$$

Étant donné les hypothèses précédentes, Kirchhoff montra qu'il existe des fonctions de (x,y,z) noté $\phi_1, \phi_2, \phi_3, \chi_1, \chi_2, \chi_3$ déterminée d'une façon unique par la configuration de la surface du véhicule tel que

$$\phi = v_x\phi_1 + v_y\phi_2 + v_z\phi_3 + \omega_x\chi_1 + \omega_y\chi_2 + \omega_z\chi_3 \quad (1.28)$$

En utilisant le théorème de Green, l'énergie cinétique du fluide T_f est donnée par

$$2T_f = \rho_0 \iiint \left(\left(\frac{\partial\phi}{\partial x} \right)^2 + \left(\frac{\partial\phi}{\partial y} \right)^2 + \left(\frac{\partial\phi}{\partial z} \right)^2 \right) dx dy dz = -\rho_0 \iint \phi(\nabla\phi \cdot \mathbf{n}) dS \quad (1.29)$$

Avec ρ_0 la densité du fluide, $-\nabla\phi \cdot \mathbf{n}$ représente la vitesse normale du fluide à la surface et $\rho_0\phi$ pression nécessaire pour générer le mouvement ; soit $\mathbf{v} = [v_x \quad v_y \quad v_z \quad \omega_x \quad \omega_y \quad \omega_z]^T$ le vecteur vitesse du dirigeable.

En substituant (1.28) dans (1.29) l'énergie cinétique est

$$\mathbf{T}_f = \frac{I}{2} \mathbf{v}^T \mathbf{M}_a \mathbf{v} \quad (1.30)$$

Où \mathbf{M}_a une matrice diagonale.

\mathbf{M}_a est une matrice diagonale dans le cas où le corps a des plans de symétrie dont les éléments représentent les masses et les inerties de la partie du fluide induite en mouvement avec la même vitesse que le corps du solide et qui dépendent de la forme de la surface de ce corps et de la densité du fluide .

Généralement dans l'étude de la dynamique des corps solides au milieu d'un fluide tel que le sous marin, torpille, ballon dirigeable, l'existence du fluide est ignorée et la masse et l'inertie du fluide induit en mouvement sont ajoutés à la masse et à l'inertie du corps du solide ; \mathbf{M}_a est dite matrice de masses ajoutées. En résumé les efforts hydrodynamiques sont dus aux échanges d'énergie cinétique entre le corps du dirigeable et les particules du fluide induites en mouvement.

Donc l'énergie cinétique des efforts de la masse ajoutée peut être défini comme suit :

$$\mathbf{T}_a = \frac{I}{2} \mathbf{v}^T \mathbf{M}_a \mathbf{v} \quad (1.31)$$

Les forces et les couples générés par l'échange d'énergie cinétique du fluide sont déterminés par les équations de Kirchhoff [28].

$$\frac{d}{dt} \left(\frac{\partial \mathbf{T}_a}{\partial \mathbf{v}_1} \right) + \mathbf{v}_2 \wedge \frac{\partial \mathbf{T}_a}{\partial \mathbf{v}_1} = -\mathbf{f}_a \quad \text{Force} \quad (1.32)$$

$$\frac{d}{dt} \left(\frac{\partial \mathbf{T}_a}{\partial \mathbf{v}_{21}} \right) + \mathbf{v}_2 \wedge \frac{\partial \mathbf{T}_a}{\partial \mathbf{v}_2} + \mathbf{v}_1 \wedge \frac{\partial \mathbf{T}_a}{\partial \mathbf{v}_1} = -\mathbf{n}_a \quad \text{Couple} \quad (1.33)$$

Ces deux équations nous donnent l'expression du torseur de la force et du couple induit par l'inertie et la masse ajoutées.

$$\Gamma_a = \left(M_a \dot{\mathbf{v}} + C_a(\mathbf{v})\mathbf{v} \right) \quad (1.34)$$

Où M_a est la matrice d'inertie ajoutée dont les termes sont constants

$$M_a = \begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{66} \end{bmatrix} \quad (1.35)$$

Avec les a_{ij} sont données en APPENDICE C

C_a est la matrice de force de Coriolis et de forces de centrifuges hydrodynamiques et peut se mettre sous la formes

$$C_a = \begin{bmatrix} 0 & 0 & 0 & 0 & a_{33}v_z & -a_{22}v_y \\ 0 & 0 & 0 & -a_{33}v_z & 0 & a_{11}v_x \\ 0 & 0 & 0 & a_{22}v_y & -a_{11}v_x & 0 \\ 0 & a_{33}v_z & -a_{22}v_y & 0 & a_{66}\omega_z & -a_{55}\omega_y \\ -a_{33}v_z & 0 & a_{11}v_x & -a_{66}\omega_z & 0 & a_{44}\omega_x \\ a_{22}v_y & -a_{11}v_x & 0 & a_{55}\omega_y & -a_{44}\omega_x & 0 \end{bmatrix} \quad (1.36)$$

1.3.2 .Les forces d'amortissements

Les forces d'amortissement sont générées par un transfert d'énergie entre le fluide et le solide immergé, leur point d'application est le centre de flottabilité du ballon dirigeable.

Elles sont représentées par une matrice $\mathbf{D}(\mathbf{v})$ strictement négative, signifiant que les forces et les couples sont appliqués au corps par l'environnement, ces efforts sont proportionnels à la vitesse

$$\boldsymbol{\tau}_D = \mathbf{D}(\mathbf{v}) \cdot \mathbf{v} \quad (1.37)$$

Ces efforts sont dus principalement aux trois facteurs suivants :

- la force d'amortissement due à la rugosité du solide.
- la force d'amortissement induite lors de la formation des tourbillons à l'arrière d'un solide.
- la force d'amortissement due à la forme du dirigeable ,cette force est modélisé comme suit :

$$\mathbf{D}_f = -\frac{1}{2} \rho C_D A |\mathbf{v}| \mathbf{v} \quad (1.38)$$

Où

ρ : densité du fluide.

C_D :coefficient de traînée .

A : surface du corps en incidence.

\mathbf{v} : vecteur vitesse .

La matrice d'amortissement $\mathbf{D}(\mathbf{v})$ est donnée par [27] :

$$\mathbf{D}(\mathbf{v}) = \mathit{diag} \{ D_{v_x} + D_{v_x v_x} |\mathbf{v}_x|, D_{v_y} + D_{v_y v_y} |\mathbf{v}_y|, D_{v_z} + D_{v_z v_z} |\mathbf{v}_z|, \\ D_{\omega_{xx}} + D_{\omega_x \omega_x} |\omega_x|, D_{\omega_{yy}} + D_{\omega_y \omega_y} |\omega_y|, D_{\omega_{zz}} + D_{\omega_z \omega_z} |\omega_z| \} \quad (1.39)$$

Les coefficients d'amortissement sont déterminés expérimentalement

1.3.3. Les efforts hydrostatiques, poussée d'Archimède et force de pesanteur

Ces efforts se présentent quand le dirigeable est à l'état de flottement sous l'effet de la combinaison des actions de la force de poussée qui s'applique au centre de poussée et à la force de pesanteur qui agit au centre de gravité

La poussée d'Archimède est égale à la force opposée au poids que subit le volume du fluide

$$F_B = \rho V_f g \quad (1.40)$$

ρ : la densité du fluide

V_f : le volume du fluide déplacé

g : L'accélération gravitationnelle

La force de pesanteur est exprimée par $F_G = mg$ où m est la masse du solide

La résultante des forces hydrostatiques est donnée par : [25][27]

$$\mathbf{g}(\boldsymbol{\eta}) = \begin{bmatrix} (W - B)\sin \theta \\ -(W - B)\cos \theta \sin \phi \\ -(W - B)\cos \theta \cos \phi \\ -(y_G W - y_B B)\cos \theta \cos \phi + (z_G W - z_B B)\cos \theta \sin \phi \\ (z_G W - z_B B)\sin \theta + (x_G W - x_B B)\cos \theta \cos \phi \\ -(x_G W - x_B B)\cos \theta \sin \phi - (y_G W - y_B B)\sin \theta \end{bmatrix} \quad (1.41)$$

Avec $W = \|F_G\|$ et $B = \|F_B\|$

1.4 .Synthèse de la dynamique du dirigeable [27] [30]

Finalement l'équation dynamique du ballon dirigeable est la suivante :

$$\mathbf{M} \dot{\mathbf{v}} + \mathbf{C}(\mathbf{v})\mathbf{v} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \quad (1.42)$$

Où

$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_a$: matrice 6x6 contenant les termes de masse et d'inertie

\mathbf{M}_{RB} est définie en (1.22)et \mathbf{M}_a en (1.35)

$\mathbf{C}(\mathbf{v}) = \mathbf{C}_{RB}(\mathbf{v}) + \mathbf{C}_a(\mathbf{v})$: matrice 6x6 des termes de forces de Coriolis et de centrifuges

\mathbf{C}_{RB} est déterminée en (1.23) et \mathbf{C}_a en (1.36)

$\mathbf{D}(\mathbf{v})$: représente une matrice contenant des termes de forces d'amortissements ,est donnée par (1.39)

$\mathbf{g}(\boldsymbol{\eta})$: matrice définie en (1.41) englobe les termes des forces hydrostatiques

$\boldsymbol{\tau}$: forces et couples des propulseurs

\mathbf{v} : torseur cinématique du ballon dirigeable.

Avec $\mathbf{v} = [v_x \ v_y \ v_z \ w_x \ w_y \ w_z]^T$

CHAPITRE 2

ASSERVISSEMENT VISUEL

2.1. Introduction:

L'asservissement visuel consiste à la mise en œuvre des trois sous systèmes suivants :

- capteurs visuels, généralement des caméras
- un dispositif d'acquisition et traitement d'image
- un système mécanique actionné , tel que les robots mobiles, les manipulateurs, les drones.

Le but de l'asservissement visuel est d'intégrer en temps réel les informations visuelles dans la boucle de commande des robots .Les techniques d'utilisation d'une caméra embarqué permettent de fournir une interaction entre la position du robot et son environnement, il en résulte une commande plus précise et plus robuste en présence d'incertitudes et d'erreurs de modélisation.

La principale tâche de l'asservissement visuel consiste à contrôler de façon stable et robuste l'attitude (position et orientation) du robot (le dirigeable) ,potentiellement variable dans le temps ,à l'aide des informations visuelles .

2.2.Les différents types de commande par la vision

La première contribution importante au domaine de la commande par la vision est due aux chercheurs Chaumette et P.Corke [12] [13] ; ils ont proposé plusieurs schémas de commandes permettant de réaliser des tâches de positionnement dans le domaine de la robotique, ils sont arrivé à distinguer deux approches concernant l'utilisation des information visuelles en commande ; Ces deux approche sont détaillées dans [12], [13] [14].

-la première est basé sur un asservissement en situation de la caméra par rapport à son environnement et dénommé Postion-Based visualisation ou asservissement dans l'espace cartésien.

-la seconde approche est basée sur une régulation dans le plan image, cette approche trouve encore plusieurs qualificatifs tel que image based look, visual servoing.

2.2.1..Asservissement dans l'espace cartésien *Position-Based Servoing*

Dans cette structure, on emploie des techniques basées sur un asservissement sur la situation de la caméra par rapport à la scène, elle nécessite une phase d'interprétation des informations visuelles, de manière à reconstruire la situation de la caméra et son environnement.

Ce mode de fonctionnement consiste en un enchaînement séquentiel de deux étapes : dans la première étape, la caméra acquiert une image et un algorithme extrait les informations visuelles de la scène ; interprète l'estimation en 3D de la position relative entre la caméra et son environnement .Cette interprétation utilise dans la plupart des cas une connaissance à priori de la géométrie de la scène ; en suite ,une loi de commande doit calculer directement les actions à envoyer au robot pour effectuer les déplacements à partir de l'erreur entre la situation estimée et la situation désirée .

La figure suivante décrit le schéma de principe d'un asservissement dans l'espace cartésien.

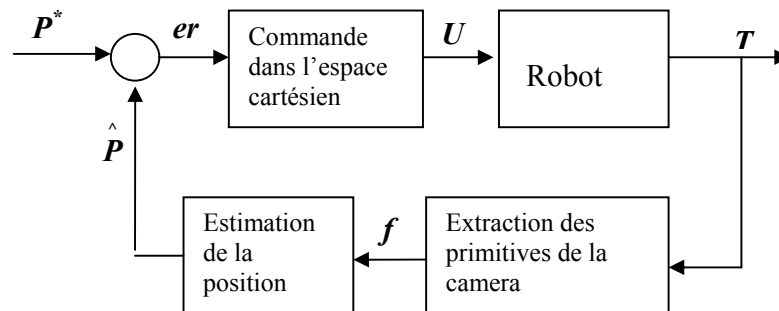


Figure 2.1: Structure Position-Based Servoing

La consigne P^* est un vecteur donnant les coordonnées d'attitude désirée de l'objet par rapport à la caméra, cette consigne est comparée à la mesure estimée issue d'un algorithme de calcul de reconstruction (estimation de la position), le correcteur fournit au robot les commandes adéquates pour que \hat{P} estimé converge vers P^* désiré ($er = P^* - \hat{P}$).

U représente la commande du robot ; T le torseur cinématique; f les informations visuelles.

Trois types d'erreurs peuvent intervenir dans ce processus :

- les erreurs d'extraction des informations visuelles inhérentes à tout traitement d'images.
- les erreurs d'estimations de la situation entre la caméra et son environnement dues aux inévitables erreurs de calibrage de la caméra et aux erreurs d'interprétation.
- les erreurs de modélisation.

2.2.2. Asservissement visuel *Image Based servoing*

Dans cette structure, les informations visuelles extraites de l'image sont directement utilisées dans la boucle de commande. Elle permet de s'affranchir totalement de la phase d'interprétation et par conséquent on évite les problèmes de calibrage et de modélisation. L'originalité de cette technique réside dans le fait que la tâche à réaliser est spécifiée directement dans le plan d'image.

Dans cette approche on utilise une relation entre le mouvement de la caméra et la variation des primitives dans l'image. Cette relation différentielle caractérisant l'interaction entre la position de la caméra et la scène est connue sous le nom Jacobien d'image ; Cette relation est donnée par :

$$\dot{f} = LT \quad (2.1)$$

Où

f : représente le vecteur de variation des primitives (informations visuelles).

T : est le torseur cinématique de la caméra (vitesses linéaires et angulaires).

L : s'appelle matrice d'interaction.

La relation (2.1) exprime la variation des primitives visuelles dans le plan d'image en fonction du mouvement de la camera. Toute fois l'évaluation de la matrice Jacobien n'est pas toujours facile à déterminer, dépend du type des informations visuelles considérées. La figure (3.2) décrit le schéma de principe de la structure *Image Based servoing* :

Les primitives f extraites de l'image, par procédure de traitement d'image, sont comparées au primitives f^* désirées ; à partir de l'erreur le contrôleur élabore une loi de commande pour faire converger les primitives extraites vers les primitives désirées .

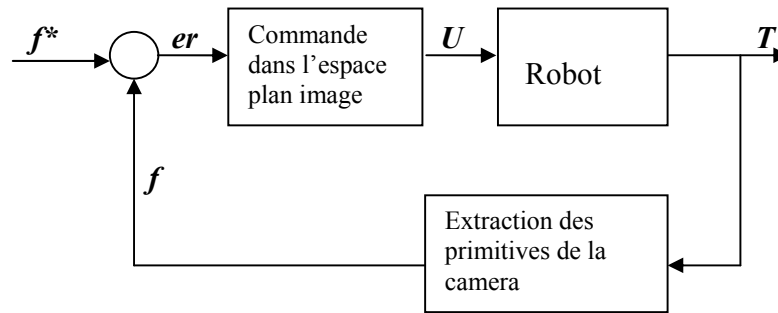


Figure 2.2: Structure image Based servoing

2.3 .Matrice d'interaction

La vitesse des information visuelles (par exemple les coordonnées d'un point) dans l'image peut s'écrire :

$$\dot{\mathbf{f}} = \mathbf{L}_s^T \mathbf{T}$$

$\mathbf{T} = (\mathbf{V}, \boldsymbol{\omega})$ est le torseur cinématique de la caméra représentant le mouvement relative entre la caméra et la scène avec $\mathbf{V} = (v_x, v_y, v_z)^T$ et $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)^T$ sont respectivement les composantes de translation et de rotation de \mathbf{T} dans le repère caméra .

\mathbf{L}_s^T matrice d'interaction caractérisant complètement l'interaction entre le capteur et son environnement . Cette matrice équivalente à une matrice Jacobienne peut être calculée explicitement pour les paramètres représentant la projection dans le plan d'image des primitives géométriques paramétrables telle que les points, les segments de droites, des droites, cercle, sphère et cylindre.

2.4. Cinématique de la caméra : [12] [27]

La camera est placée sous le ballon dirigeable, pour superviser et collecter les informations visuelles de la scène à prospecter. La camera est munie d'un servomoteur pour la commande de son orientation. Elle est choisie à priori par l'opérateur.

On adopte un repère rigidement lié à la camera, son centre est confondu avec le centre optique et il est exprimé dans le repère ballon par les coordonnées (x'_c, y'_c, z'_c)

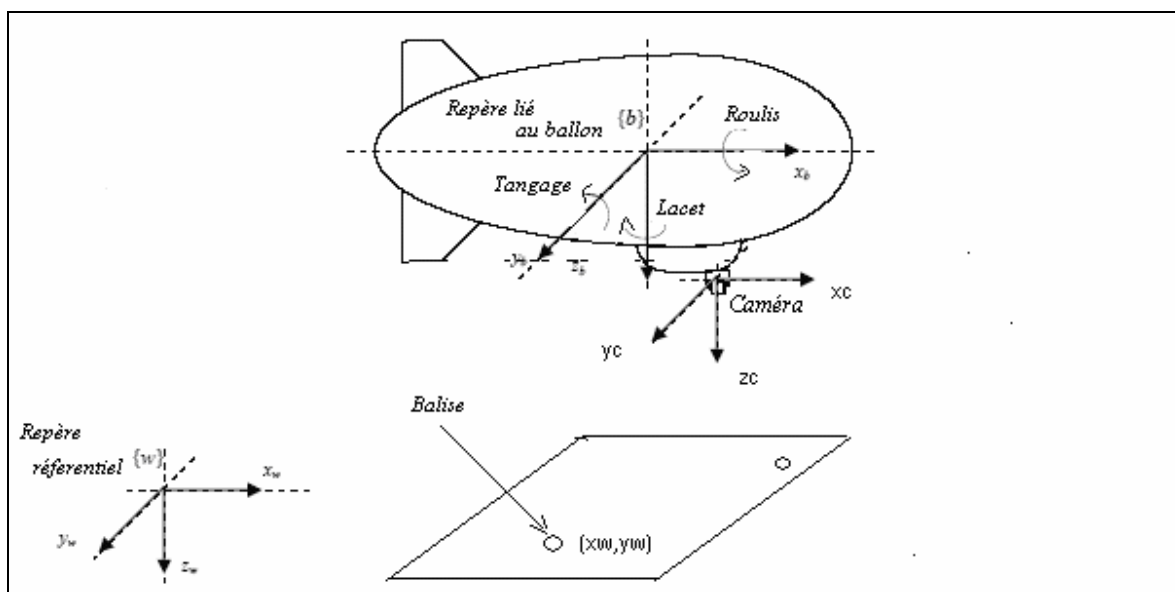


Figure 2.3 : Position du repère de la camera

L'orientation de la camera est définie par les angles

α autour de l'axe x est appelé pan.

β autour de l'axe y est dit tilt .

γ autour de l'axe z est nommé swing.

On choisit à priori l'angle tilt variable, l'angle pan nul et l'angle swing est égale $\pi/2$

Le passage du repère ballon au repère camera se fait par la matrice suivante :

$${}^bT_c = \begin{bmatrix} {}^bR_c & {}^b\mathbf{t}_c \\ \mathbf{0}_{1 \times 3} & \mathbf{I} \end{bmatrix} \quad (2.2)$$

Avec ${}^b_c\mathbf{R}$ matrice de rotation

$${}^b_c\mathbf{R} = \begin{bmatrix} -\cos \gamma \cos \beta & -\sin \gamma \cos \beta + \cos \gamma \sin \beta \sin \alpha & \sin \gamma \sin \beta + \cos \alpha \cos \gamma \sin \beta \\ \sin \gamma \cos \beta & \cos \gamma \cos \beta + \sin \gamma \sin \beta \sin \alpha & -\cos \gamma \sin \beta + \sin \beta \sin \gamma \cos \alpha \\ -\sin \beta & \cos \beta \sin \alpha & \cos \beta \cos \alpha \end{bmatrix} \quad (2.3)$$

Et ${}^b_c\mathbf{t} = [x'_c \quad y'_c \quad z'_c]^T$ Vecteur de translation

2.5. Modélisation de la camera : [12]

La camera est un capteur qui réalise une projection d'un ensemble de points d'une scène tridimensionnelle dans un espace à deux dimensions (plan image). Le cadre naturel pour étudier de manière simple de telles transformations est la géométrie projective.

Trois systèmes de coordonnées sont définis selon la figure :

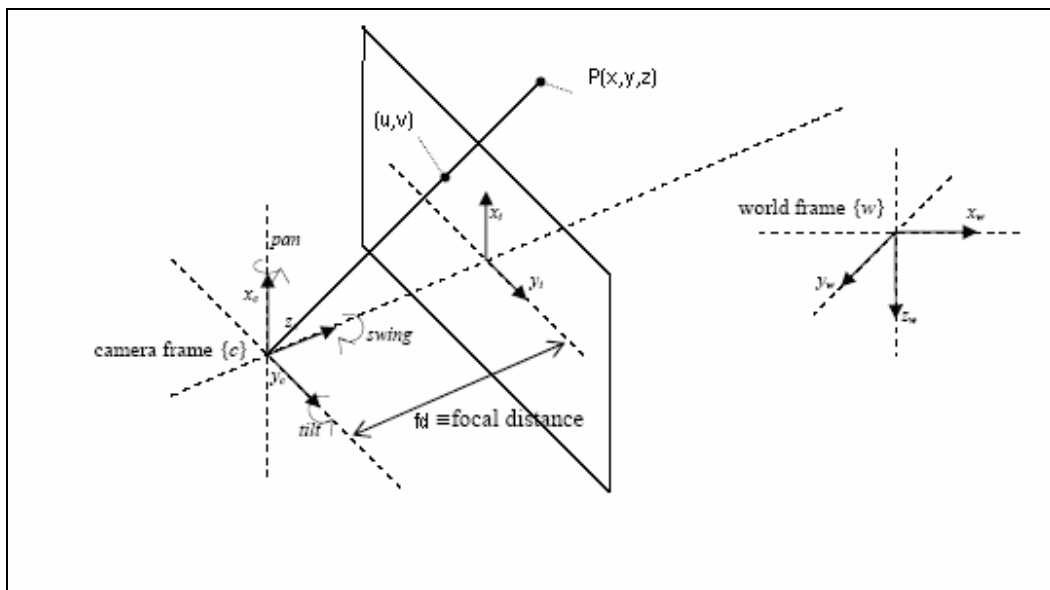


Figure 2.4 : Projection d'un point dans le plan image

- Un repère 3D appelé repère objet (x_w, y_w, z_w) dans lequel sont exprimés les points d'un objet dans une scène.

-Un repère (x_c, y_c, z_c) , lié à la camera de telle façon que l'origine soit le centre optique et l'axe z_c est confondu avec le rayon optique.

-Un repère 2D lié au plan d'image est situé à une distance f (distance focale) de $\mathbf{o}_c(x_i, y_i)$.

Le model projectif est dit sténopé et la représentation en coordonnées homogènes est utilisée.

Un point P d'un objet de la scène de coordonnées ${}^w\mathbf{P} = (x, y, z, 1)^T$ dans le repère objet est exprimé dans le repère liée à la camera par ${}^c\mathbf{P} = (x_c, y_c, z_c, 1)^T$.

$${}^c\mathbf{P} = \mathbf{G} \cdot {}^w\mathbf{P}$$

\mathbf{G} : C'est la matrice homogène qui exprime le changement de repère :

$$\mathbf{G} = \begin{bmatrix} {}^c\mathbf{R}_w & {}^c\mathbf{t}_w \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \quad (2.4)$$

Ou ${}^c\mathbf{R}_w$ et ${}^c\mathbf{t}_w$ sont respectivement la matrice de rotation et le vecteur de translation entre \mathbf{R}_c et \mathbf{R}_w

Le point ${}^c\mathbf{P}$ exprimé dans le repère caméra est projeté en un point \mathbf{m}' dans l'espace image $\mathbf{m}' = (u_m, v_m, 1)$ en coordonnées métriques.

$$\mathbf{m}' = \mathbf{T}_p \cdot {}^c\mathbf{P} \quad (2.5)$$

$$\text{Avec} \quad \frac{z_c}{f_d} \mathbf{m}' = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot {}^c\mathbf{P} \quad (2.6)$$

$$\text{Et} \quad \mathbf{T}_p = \frac{f_d}{z_c} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.7)$$

f_d : distance focale

Enfin, la caméra effectue également une transformation affine des coordonnées homogènes métriques $\mathbf{m}' = (u_m, v_m, 1)$ en coordonnées homogènes en pixels

$$\mathbf{p} = \mathbf{K}\mathbf{m}' \quad (2.7)$$

Où \mathbf{K} matrice des paramètres intrinsèques de la camera (3x3) dont les coefficients peuvent être déterminés par une procédure de calibrage

$$\mathbf{K} = \begin{bmatrix} -f_d k_u & f_d k_u \cot \theta' & u_0 \\ 0 & -f_d \frac{k_v}{\sin \theta'} & v_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_u & \alpha_{uv} & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

k_u, k_v sont les facteurs d'échelles horizontal et vertical des pixels

u_0, v_0 sont les coordonnées du point principal défini comme étant l'intersection entre l'axe optique et le plan image

θ' est l'angle entre les deux axes du repère image, puisque en pratique les vecteurs de ce repère ne sont pas tout à fait perpendiculaire

En résumé, on obtient la transformation réalisée par la caméra d'un point \mathbf{P} dont les coordonnées sont exprimées dans le repère de la scène en un point \mathbf{p} du plan image dont les coordonnées sont exprimées en pixels.

$$\mathbf{p} = \mathbf{K}\mathbf{T}_p\mathbf{G}\mathbf{P} \quad (2.9)$$

Avec $\mathbf{P} = (x, y, z, 1)$ $\mathbf{p} = (u, v, 1)$

2.6. Calcul de la matrice d'interaction d'un point[12]

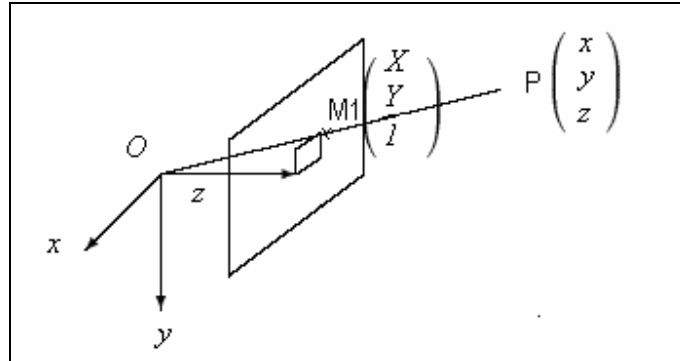


Figure 2.5 : Modèle de la caméra

On suppose que la distance focale de la caméra égale à 1 de telle sorte que tout point P du repère lié à la caméra et ayant pour coordonnées $P(x, y, z, 1) = \bar{x}$ se projette par projection perspective sur le plan image en M_I de coordonnées métriques $\bar{X} = (X, Y, 1)$.

On suppose que $\theta' = \frac{\pi}{2}$.

$$\bar{X} = \frac{f_d}{z_c} \bar{x} \quad z_c = z, \quad f_d = 1 \quad (2.10)$$

Le point m' est observé par une caméra en mouvement de translation et de rotation avec des vitesses linéaires et angulaires $V = (v_x, v_y, v_z)^T$ et $\omega = (\omega_x, \omega_y, \omega_z)^T$

La vitesse relative du point m' par rapport à la caméra s'exprime par la relation suivante []

$$\frac{dm'}{dt} = -V - \omega \wedge om' \quad (2.11)$$

$$\dot{\bar{x}} = -V - \omega \wedge \bar{x} \quad (2.12)$$

D'où

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = - \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \wedge \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (2.13)$$

En dérivant par rapport au temps la relation (2.10) on obtient

$$\dot{\bar{X}} = -\frac{\dot{z}}{z^2} \bar{x} + \frac{1}{z} \dot{\bar{x}} \quad (2.14)$$

En combinant les équations (2.10) et (2.13)

On obtient

$$\begin{pmatrix} \dot{\bar{X}} \\ \dot{\bar{Y}} \end{pmatrix} = \begin{pmatrix} -\frac{1}{z} & 0 & \frac{X}{z} & XY & -(1+X^2) & Y \\ 0 & -\frac{1}{z} & \frac{Y}{z} & (1+Y^2) & -XY & -X \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \quad (2.15)$$

La forme de la matrice d'interaction L_s^T dans le cas d'un point est :

$$L_s^T = \begin{pmatrix} -\frac{1}{z} & 0 & \frac{X}{z} & XY & -(1+X^2) & Y \\ 0 & -\frac{1}{z} & \frac{Y}{z} & (1+Y^2) & -XY & -X \end{pmatrix} \quad (2.16)$$

Pour plus de détail, la méthode générale pour calculer la matrice d'interaction associée aux paramètres représentant les primitives géométriques paramétrables est décrite dans [12]. Les résultats explicites y sont donnés pour les primitives les plus usuelles : points, segments, droites, cercles, sphères et cylindres.

CHAPITRE 3

LA COMMANDE FLOUE ET NEURO-FLOU

3.1. Introduction

De nos jours, la logique floue est un axe de recherche important sur lequel se penchent de nombreux chercheurs [34] [36]. Des retombées technologiques sont déjà disponibles, dans le domaine du grand public (appareils photos, machines à laver, fours à micro onde), et dans le domaine industriel (réglage et commande de processus complexes liés à l'énergie, aux transports, à la transformation de la matière, à la robotique et aux machines-outils).

Les concepts théoriques de la logique floue ont été formulés en 1965 par le professeur Zadeh. Il a introduit la notion de sous ensemble flous, qui s'appuie sur le degré d'appartenance d'un élément à un sous ensemble. Cette approche permet de modéliser des connaissances imparfaitement décrites ou imprécises.

Dès 1975, Mamdani et Assilian ont publié [53] les premiers résultats permettant une exploitation de cette théorie dans des systèmes de réglage. En utilisant une structure de contrôleur relativement simple, ils ont obtenu de meilleurs résultats lors de la commande de certains processus que ceux fournis par un régulateur standard de type PID.

Peu de temps après, en 1977, les danois ont appliqué la logique floue à la commande des tubes broyeurs pour la fabrication de ciment.

A partir de 1985, les japonais ont appliqué la logique floue dans des produits à usage public (machine à laver)

3.2. Sous ensemble flou[24] [36]

Etant donné un ensemble de référence X dénombrable ou non, dénoté par ses éléments $\{x\}$. Un sous ensemble classique A de X caractérisé par une fonction caractéristique

μ_A prenant simplement deux valeurs 0 ou 1 est défini par :

$$\mu_A : X \rightarrow \{0,1\} \quad \text{tel que} \quad \mu_A = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$$

Si l'appartenance de certains éléments n'est pas absolue (l'élément appartient partiellement au sous ensemble), la fonction caractéristique est remplacée par une fonction d'appartenance $\mu_A(x)$ qui prend ses valeurs dans l'intervalle $[0,1]$. Cette classe est appelée sous ensemble flou de X . l'ensemble X sera appelé ensemble de l'univers du discours.

Un sous ensemble flou A dans un univers du discours X est caractérisé par sa fonction d'appartenance $\mu_A(x)$ qui associe à chaque élément x de X une valeur dans l'intervalle des nombre réels $[0,1]$

$$\mu_A(x) : X \rightarrow [0,1] \quad (3.2)$$

Ainsi le sous ensemble flou A de X peut être défini comme étant l'ensemble des couples ordonnés

$$A : \{(x, \mu_A(x)) | x \in X\} \quad (3.3)$$

Les sous ensemble flous A de X est aussi représenté par la notation suivante :

$$A = \int_X \mu_A(x)/x \quad \text{si } X \text{ est continue} \quad (3.4)$$

$$A = \sum_{x_i \in X} \mu_A(x_i)/x_i \quad \text{si } X \text{ est discret} \quad (3.5)$$

La fonctions d'appartenance est généralement de forme triangulaire, trapézoïdale ou gaussienne.

Exemple : (figure 3.1)

En logique classique toute les températures $<15^\circ\text{C}$ appartiennent à l'ensemble (moins 15°C) les autre n'appartiennent pas .En logique flou $T=15^\circ$ appartient au sous ensemble « froids » avec un degré d'appartenance égale à 0.10 et appartient aussi au sous ensemble « chaud » avec un degré d'appartenance égale à 0.70

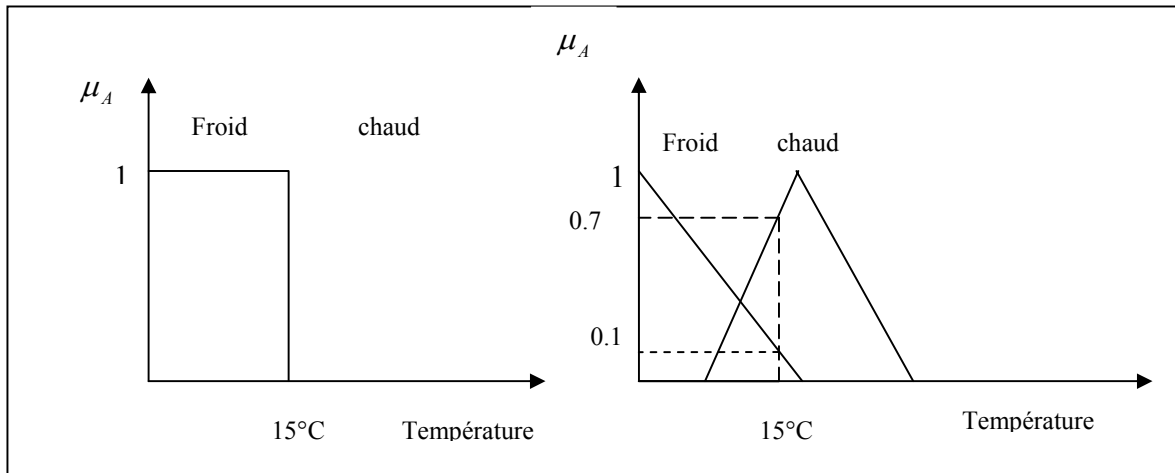


Figure 3.1 : Représentation d'un sous ensemble flou

3.2.1. Opérations sur les sous-ensembles flous

Supposons que A et B sont deux sous-ensembles flous définis dans un univers du discours X par les fonctions d'appartenance μ_A et μ_B . On peut définir des opérations telles que l'égalité, l'inclusion, l'intersection, l'union et le complément grâce à des opérations sur les fonctions d'appartenance.

-Egalité

A et B sont dits égaux si : $\forall x \in X \quad \mu_A(x) = \mu_B(x)$ (3.6)

-Complément

Le complément A par rapport à X est défini par la fonction d'appartenance suivante

$$\mu_{A^c}(x) = 1 - \mu_A(x) \quad \forall x \in X \quad (3.6)$$

-Inclusion

A est dit inclus dans B, que l'on note $A \subseteq B$, si et seulement si :

$$\forall x \in X \quad \mu_A(x) \leq \mu_B(x) \quad (3.8)$$

-Union

L'union de A et B, que l'on note $A \cup B$ est le sous-ensemble flou ayant la fonction d'appartenance suivante :

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x)\} \quad \forall x \in X \quad (3.9)$$

-Intersection

L'intersection de A et B, que l'on note $A \cap B$ est sous-ensemble flou constitué des éléments de X tel que

$$\forall x \in X \quad \mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x)\} \quad (3.10)$$

3.2.2. Normes et conormes triangulaires

D'après le paragraphe précédent, les définitions des opérations entre ensembles flous sont basées sur une combinaison des fonctions d'appartenance utilisant les opérations de minimum, maximum et de complément à 1. Il existe en fait, une multitude d'opérateurs que l'on peut utiliser pour mettre en oeuvre les opérations entre ensembles flous. Ces opérateurs sont regroupés en deux familles. Les normes triangulaires, notées T-normes, qui définissent les opérateurs d'intersection ou de conjonction et les conormes triangulaires, notées T-conormes, définissant les opérateurs d'union ou de disjonction :

t-norme	t-conorme	nom
$\min(x, y)$	$\max(x, y)$	Zadeh
$x \cdot y$	$x + y - x \cdot y$	probabiliste
$\max(x + y - 1, 0)$	$\min(x + y, 1)$	Lukasiewicz
$\frac{xy}{\gamma + (1-\gamma)(x+y-xy)}$	$\frac{x+y-xy-(1-\gamma)xy}{1-(1-\gamma)xy}$	Hamacher ($\gamma > 0$)
$\begin{cases} x & \text{si } y = 1 \\ y & \text{si } x = 1 \\ 0 & \text{sinon} \end{cases}$	$\begin{cases} x & \text{si } y = 0 \\ y & \text{si } x = 0 \\ 1 & \text{sinon} \end{cases}$	Weber

Tableau 3.1 : Les T-normes et T-conormes les plus utilisées[24]

3.3 .Raisonnement en logique flou

3.3.1.Variables linguistiques

Les variables linguistiques permettent de décrire la connaissance acquise par un expert sur une variable, même lorsqu'elle est vague ou imprécise. Une variable linguistique est définie par un ensemble de termes de la forme $(V, T(V), X, M)$.

V : représente le nom de la variable définie sur l'univers du discours X .

$T(V)$ est un ensemble des termes linguistiques qui sont des nombres flous,

M est l'ensemble des règles sémantiques qui permettent de définir les termes linguistiques.

Exemple :

Afin de décrire la température d'une pièce par une variable linguistique, on peut utiliser l'ensemble des termes suivants :

$$T(V) = \{\text{froid, tiède, chaud}\}.$$

En considérant que l'univers de discours est l'intervalle de température $[0,45]$, on peut utiliser les règles sémantiques suivantes :

- le terme linguistique froid représente une température d'environ 10°C .
- le terme tiède : «une température d'environ 17°C »
- le terme chaud « une température supérieur à 24°C . »

Ces termes peuvent être caractérisés par les fonctions d'appartenance suivantes :

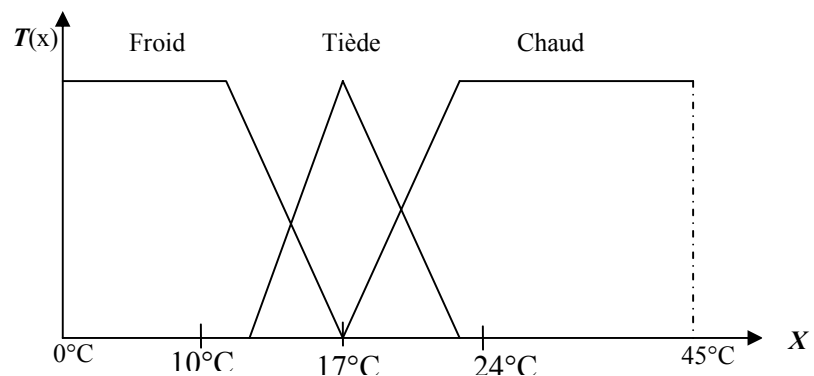


Figure 3.2 : Variable linguistique décrivant la température

3.3.2. Propositions floues élémentaires

Une proposition floue est définie à partir d'un ensemble de variables linguistiques afin de représenter une connaissance. Par exemple, «la température de la pièce est froide». Une forme élémentaire de proposition floue est définie à partir d'une seule variable linguistique ($V, T(V), X, M$) et exprimée simplement par la phrase :

$$P : V \text{ est } A \quad (3.11)$$

où V est une variable qui prend sa valeur dans l'univers du discours X , et A est l'un des termes linguistiques de $T(V)$.

Supposons que l'on veuille définir la température de $T_1=15^\circ\text{C}$ d'une pièce en utilisant la variable linguistique de la figure 3.2 la proposition floue correspondante est exprimée par ($p_1 : T_1 \text{ est tiède}$). A partir de la fonction d'appartenance définissant «tiède», nous trouvons que le degré d'appartenance d'une température de 15°C au sous ensemble flou «tiède» est de 0.7, comme on le voit sur la figure 3.3.

On peut aussi exprimer une autre proposition ($p_2 : T_2 \text{ est froide}$), cependant la valeur de vérité est égale à 0.3.

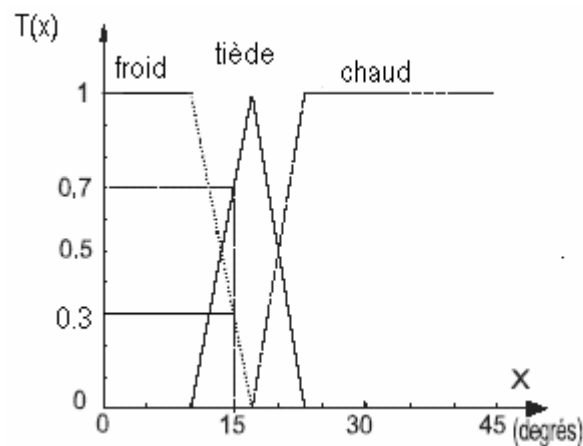


Figure 3.3 : Valeur de vérité d'un terme linguistique

3.3.4. Propositions floues générales

Une proposition floue générale est obtenue par la composition de propositions élémentaires.

La conjonction de propositions floues élémentaires

$$p : (X_1 \text{ est } A_1) \text{ et } \dots \text{ et } (X_n \text{ est } A_n) \quad (3.12)$$

Sa valeur de vérité est alors définie sur l'univers de discours par $X_1 \times X_2 \times \dots \times X_n$

$$V(p) = \min(\mu_{A_1}(X_1), \dots, \mu_{A_n}(X_n)) \quad (3.14)$$

La disjonction de propositions floues élémentaires :

$$p : (X_1 \text{ est } A_1) \text{ ou } \dots \text{ ou } (X_n \text{ est } A_n) \quad (3.15)$$

La valeur de vérité de la disjonction sur les univers du discours $X_1 \times X_2 \times \dots \times X_n$ est

$$V(p) = \max(\mu_{A_1}(X_1), \dots, \mu_{A_n}(X_n)) \quad (3.16)$$

3.3.5. Règles linguistiques

L'idée principale basée sur la logique floue est d'exprimer la connaissance humaine sous forme des règles linguistiques de la forme « si.....alors..... » , chaque règle a deux parties ; la partie antécédente décrivant l'état du système constitue la prémisse de la règle ; la partie conséquente exprime l'action que l'opérateur doit exécuter ; c'est une implication floue .

3.3.6. Implications floues

L'implication floue permet de définir une liaison entre la prémisse « $X \text{ est } A$ » et la conclusion « $Y \text{ est } B$ » de cette règle. Elle peut se ramener à une fonction d'appartenance. L'implication floue est notée : $A \Rightarrow B$

En utilisant la fonction d'appartenance $\mu_A(x)$ intervenant dans la prémisse et celle $\mu_B(x)$ intervenant dans la conclusion de la règle, la fonction d'appartenance de l'implication floue s'exprime :

$$\mu_R(x, y) = I(\mu_A(x), \mu_B(x)) \quad \text{pour tout } (x, y) \text{ de } X \times Y, \quad (3.17)$$

Les implications floues le plus souvent employées sont :

Valeur de vérité	$I(\mu_A(x), \mu_B(x))$	nom
I_m	$\min(\mu_A(x), \mu_B(x))$	Mamdani
I_l	$\mu_A(x) \times \mu_B(x)$	Larsen
I_r	$1 - \mu_A(x) + \mu_A(x) \times \mu_B(x)$	Reichenbach
I_w	$\max(1 - \mu_A(x), \min(\mu_A(x), \mu_B(x)))$	Willmott
I_{rg}	$\begin{cases} 1 & \text{si } \mu_A(x) \leq \mu_B(x) \\ 0 & \text{sinon} \end{cases}$	Rescher-Gaines
I_{kd}	$\max(1 - \mu_A(x), \mu_B(x))$	Kleene-Dienes
I_{bg}	$\begin{cases} 1 & \text{si } \mu_A(x) \leq \mu_B(x) \\ \mu_B(x) & \text{sinon} \end{cases}$	Brouwer-Gödel
I_g	$\begin{cases} \min(\mu_B(x)/\mu_A(x), 1) & \text{si } \mu_A(x) \neq 0 \\ 0 & \text{sinon} \end{cases}$	Goguen
I_l	$\min(1 - \mu_A(x) + \mu_B(x), 1)$	Lukasiewicz

Tableau 3.2 : Implications floues les plus utilisées [24]

Parmi les implications définies dans ce tableau les implications de Mamdani et de Larsen sont les plus connues en raison de leurs applications dans la commande floue.

3.4. Commande floue

Lorsqu'un opérateur humain commande manuellement un système, les actions qu'il réalise sont dictées par une connaissance subjective du fonctionnement de ce système. Par exemple, s'il fait froid dans une pièce, on augmente le chauffage; s'il fait très froid, on chauffe plus. Cette commande du système peut être envisagée de façon différente selon la personne qui la réalise : la sensation de froid n'est pas directement liée à une mesure de la température. Ce principe est à la base de la commande floue. La mesure réalisée sur le système (température) est prise en compte par l'intermédiaire d'une variable linguistique (froid, tiède, chaud), qui est issue d'une analyse par un expert humain. Ensuite, l'action à réaliser est déduite à la fois d'un ensemble de règles de commande et de l'état du système, qualifié par la variable linguistique. Enfin, la commande finale du système est obtenue en utilisant les conclusions des implications floues.

3.4.1. La structure générale d'un contrôleur flou [36]

Un contrôleur basé sur la logique floue comporte les éléments décrits dans le synoptique suivants :

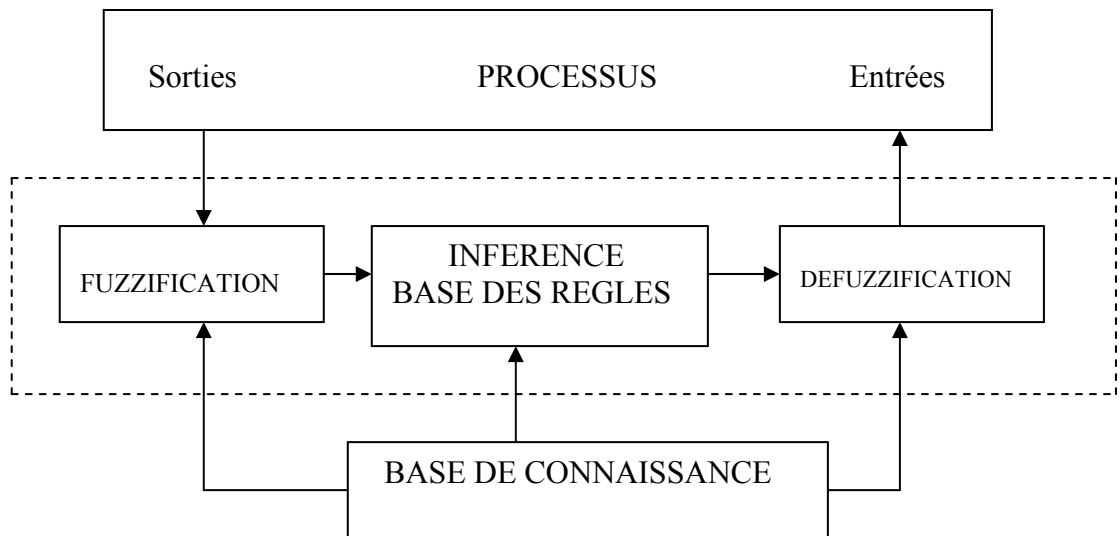


Figure 3.4 : Configuration générale d'un contrôleur flou

- Un bloc d'interface avec le flou ou interface de fuzzification.
- Une base de connaissances (base de données constituée de paramètres des fonctions d'appartenance et base de règles linguistiques de commande).
- Un sous-système réalisant un raisonnement en utilisant des méthodes issues de la logique floue ou moteur d'inférence.
- Un sous-système de défuzzification ou interface avec le non flou, qui fournit les sorties

3.4.2. La fuzzification :

Les valeurs délivrées par les capteurs de mesures, décrivant l'état du système à contrôler, sont traduites en un sous ensemble flou, autrement dit chaque grandeur ou variable d'entrée se voit attribuée un degré d'appartenance d'un sous ensemble flou dans un univers de discours.

La variable linguistique, qui est définie par une expertise, doit respecter un certain nombre de critères :

- Les fonctions d'appartenance doivent respecter l'ordre linguistique.
- Les fonctions d'appartenance ne doivent pas trop se superposer.

3.4.3. Base de connaissance

Les règles floues constituent la base de connaissance décrivant des connaissances de l'état du système en fonction des qualifications linguistiques déduites des expériences acquises par les opérateurs ou les experts. Ces connaissances sont traduites en règles simples pouvant être utilisées dans un système d'inférence floue.

Aujourd'hui, il est cependant possible de constituer une base de règles floues grâce à des méthodes d'apprentissage, sans avoir nécessairement besoin d'un expert humain tel que les réseaux de neurones et algorithmes génétiques.

- Ces règles se présentent sous la forme [Si - prémisses - alors - conclusion -]; les prémisses constituent les entrées du contrôleur flou et les conclusions correspondent sa sortie. Ces règles sont en fait des propositions floues incluant un opérateur d'implication .

Il existe deux types de règles:

Les règles de type Mamdani : $Si\ x\ est\ A\ et\ y\ est\ B\ ,\ alors\ z\ est\ C$

Les règles de type Sugeno : $Si\ x\ est\ A\ et\ y\ est\ B\ ,\ alors\ z = f(x)$

3.4.4. Inférences floues (activation des règles linguistiques)

Une inférence floue est une relation floue définie entre deux sous-ensembles flous. Le mécanisme d'inférence consiste à déterminer des règles floues activées pour déterminer une valeur floue pour les variables de sorties. Les deux méthodes d'inférence les plus couramment utilisées sont le mini et le prod (figure ci-dessous)

Exemple 1: Règle1 : $si\ X_1\ est\ A_1\ et\ X_2\ est\ A_2\ alors\ Y\ est\ B_1$

Règle2 : $si\ X_1\ est\ A_3\ et\ X_2\ est\ A_4\ alors\ Y\ est\ B_2$

L'inférence mini consiste à tronquer à la valeur μ la fonction caractéristique associée à la variable de sortie

L'inférence prod revient à affecter coefficient d'échelle μ_p à μ la fonction caractéristique associée à la variable de sortie

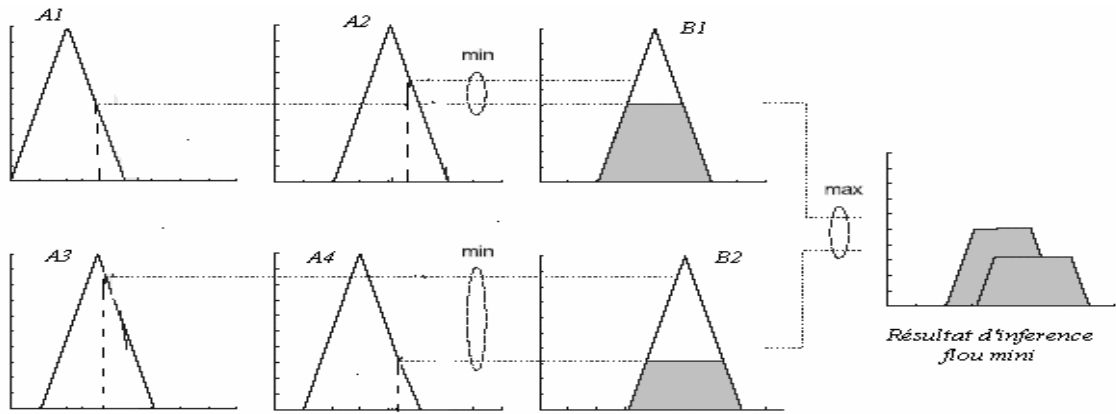


Figure 3.5 : L'inférence flou mini

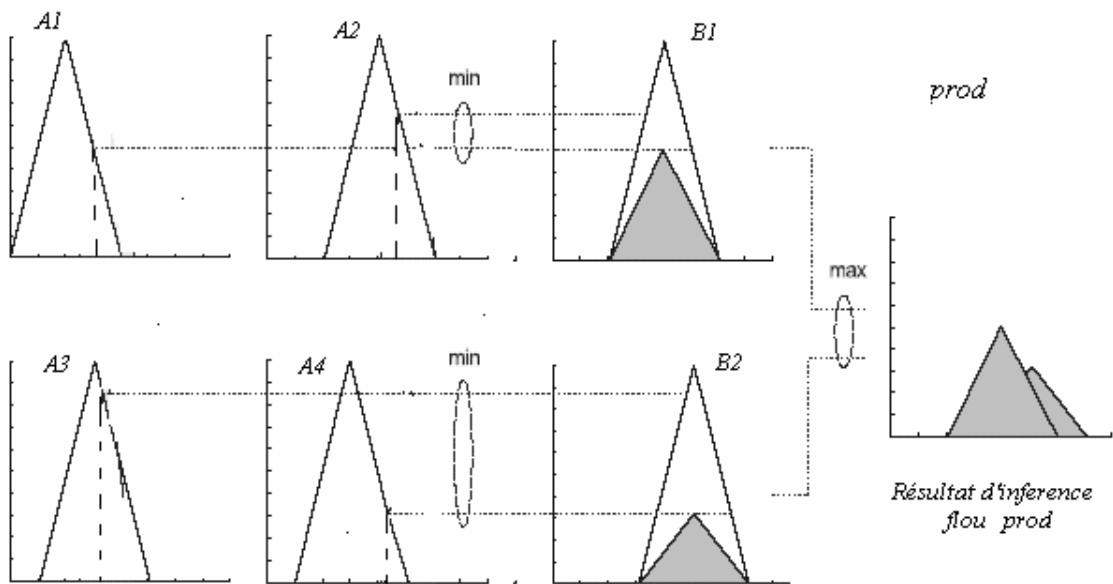


Figure 3.6 : Inférence flou prod

Exemple 2:

Soit un contrôleur à deux entrées ; l'erreur E et le variation d'erreur ΔE et une seule sortie S ; admettons la base de règles floues suivantes :

- Règle 1 : Si (Eest EZ) et (ΔE est PP) ; alors (Sest EZ)
 Règle 2 : Si (Eest EZ) et (ΔE est EZ) ; alors (Sest EZ)
 Règle 3 : Si (Eest NP) et (ΔE est EZ) ; alors (Sest PG)

Avec les termes linguistiques suivants :

EZ : environ de zéro ;

PP positif petit ;

NP négatif petit ;

PG positif grand

Prenant comme entrées : $E=-1$, $\Delta E =0.5$

Les valeurs d'entrée du contrôleur sont transformées en termes linguistiques dont les fonctions d'appartenance sont représentées sur la figure ci dessous.

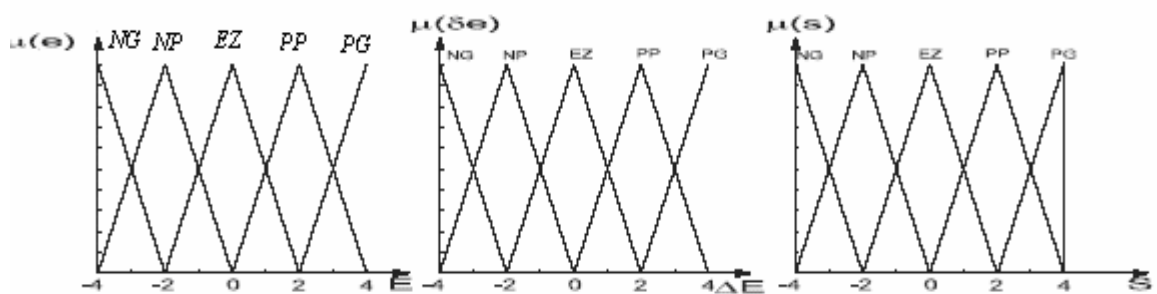


Figure 3.7 : Fonctions d'appartenance des termes linguistiques.

En utilisant l'inférence de Mamdani, l'inférence des règles est schématisée entrées comme suit :

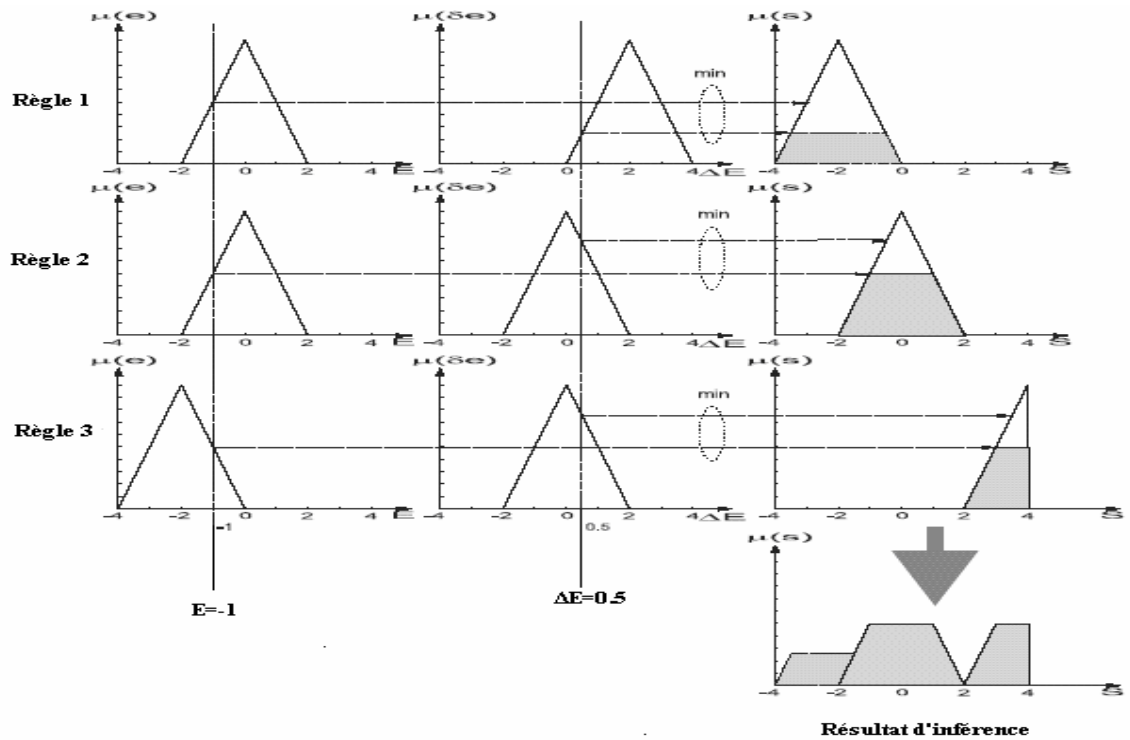


Figure 3.8 : Résultat d'inférence

3.4.5. La défuzzification

La partie inférence fournit un résultat qui est une fonction d'appartenance. Or, la sortie du contrôleur est en général une grandeur physique. La défuzzification est le traitement qui permet le passage du résultat de l'inférence à la grandeur fournie en sortie. Parmi les méthodes les plus utilisées on cite :

- Defuzzification par centre de gravité

La défuzzification par centre de gravité consiste à calculer l'abscisse du centre de gravité de la fonction d'appartenance issue de l'inférence selon

$$y_{cg} = \frac{\int y \cdot \mu_B(y) dy}{\int \mu_B(y) dy} \quad (3.18)$$

En pratique, on estime le centre de gravité en calculant la moyenne d'un certain nombre de points échantillonnés sur la fonction :

$$y_{cg} = \frac{\sum y_i \mu_B(y_i)}{\sum \mu_B(y_i)} \quad (3.19)$$

-Defuzzification par centre maximum

Dans cette méthode, la valeur de sortie est estimée par l'abscisse du point correspondant au centre de l'intervalle pour lequel la fonction d'appartenance est maximale. Cette valeur est donnée par l'expression suivante

$$y_{cm} = \frac{\inf M + \sup M}{2} \quad (3.20)$$

Où M est l'ensemble des points pour lesquels la fonction d'appartenance est maximale. Dans le cas discret, on explore en fait la liste de tous les points pour lesquels la fonction d'appartenance est maximale afin de trouver le plus petit et le plus grand de ces points

- Defuzzification par la valeur maximum

Cette méthode ne s'utilise que dans le cas discret, on prend la moyenne des abscisses du maximum :

$$y_m = \frac{\sum_{y_i \in M} y_i}{|M|} \quad (3.21)$$

3.5. Commande hybride neuro-flous

La commande neuro-flou utilise une combinaison des techniques des réseaux de neurones artificiels et celle de la logique flou ; plusieurs auteurs ont essayé de réaliser l'hybridation de ces deux approches [43] [44].

3.5.1. Réseaux de neurones. [33][37]

Le principe des réseaux de neurones artificiels est inspiré du système nerveux humain. Les différents modèles appliqués en commande et traitement de signal utilisent des unités, appelés neurones artificiels, capables de réaliser un traitement et d'échanger des informations entre eux. On associe généralement aux RNA un algorithme d'apprentissage permettant de modifier les poids des connexions afin de réaliser une tâche donnée.

3.5.2. Le neurone artificiel

La structure du neurone artificiel est inspirée du neurone biologique qui peut être décrite par la figure suivante :

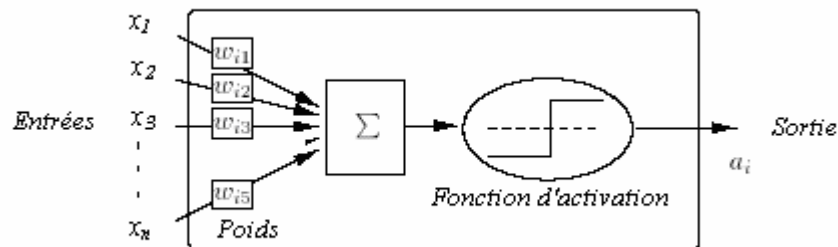


Figure 3.9 : Neurone artificiel

La sortie du neurone est une fonction non linéaire d'une combinaison des entrées $e_i=x_i$ pondérées par des paramètres w_{ij} (poids synaptiques).le neurone est alors considéré comme une fonction algébrique non linéaire paramétré à valeur bornées .

$$y = f\left(w_0 + \sum_{i=1}^n w_i x_i\right) \quad (3.22)$$

f fonction d'activation du neurone.

Plusieurs fonctions peuvent être utilisées comme fonction d'activation, parmi lesquelles, on peut citer : fonction seuil, fonction, tangente hyperbolique, gaussienne, linéaire ou fonction sigmoïde qui est la plus utilisée.

3.5.3. Réseau de neurones artificiel

Un réseau de neurones artificiel est un ensemble de neurones reliés entre eux par des connexions pondérées, il se caractérise par le type des unités utilisées, sa topologie et son algorithme d'apprentissage. On distingue deux types de réseaux de neurones, les réseaux de neurones non bouclés et les réseaux de neurones bouclés dit récurrent.

Le réseau de neurones non bouclé est composé des neurones d'entrées recevant les données du monde extérieur et des neurones de sortie fournissant le résultat du traitement effectué, les unités intermédiaires sont qualifiées de couches cachées comme le montre cette figure :

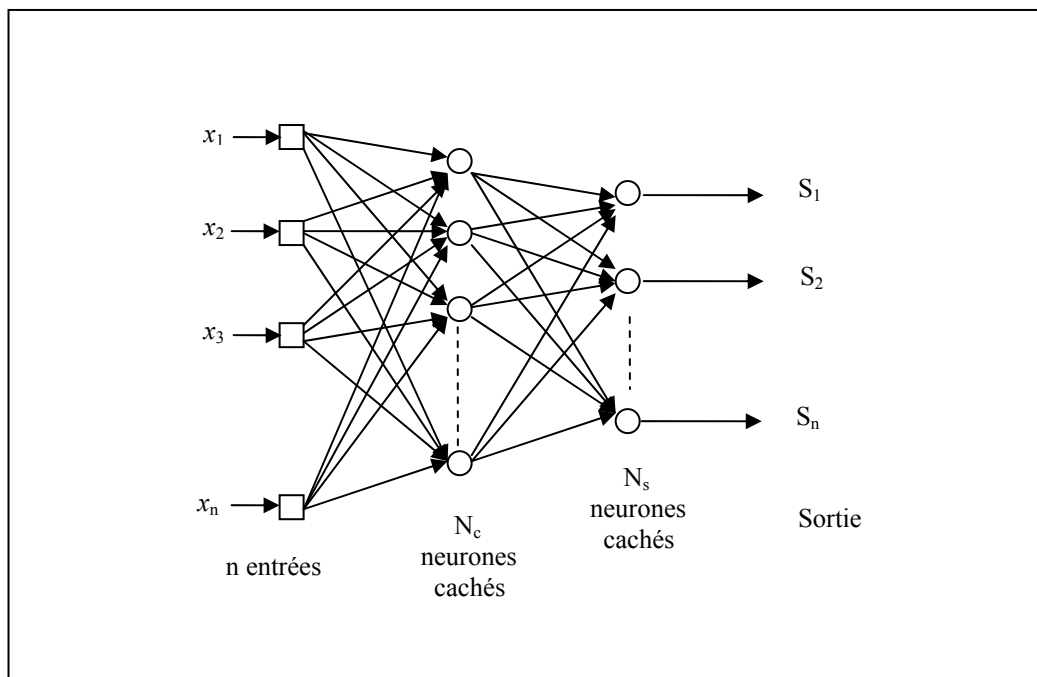


Figure 3.10 : Réseaux de neurones non bouclés

Le réseau de neurones bouclé (récurrent):

Ce réseau possède des boucles de retour sur lui-même comme le montre la figure ci-dessous.

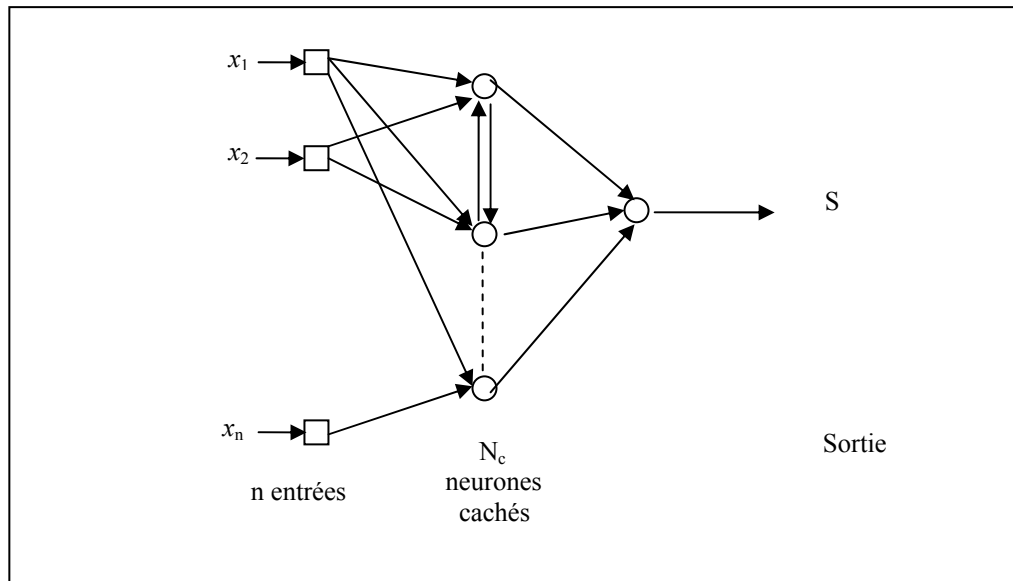


Figure 3.11 : Réseaux de neurones bouclés

L'un des principaux avantages des réseaux de neurones réside dans leur capacité d'apprentissage; on entend par apprentissage la modification automatique des poids des connexions ; il existe trois grandes classes d'apprentissage ;Supervisé,semi supervisé et un apprentissage non supervisé .

- Dans l'apprentissage supervisé on donne au réseau neuronal des exemples résolus ou sortie souhaitée. A partir de la différence entre la sortie du réseau et la sortie souhaitée on effectue l'adaptation en modifiant les poids de connexions.

Les règles d'apprentissage appliquées, dont le rôle consiste à trouver l'optimum des poids synaptiques sont basées les techniques des recherches du minimum, sont citée ci dessous.

La descente du gradient

La descente du gradient avec inertie

La descente stochastique

Le recuit simulé

Les algorithmes génétiques

Dans l'apprentissage semi supervisé la valeur exacte de la sortie désirée n'est pas disponible

Dans l'apprentissage non supervisé il n'y a pas de réponse désirée, la tâche du réseau peut être par exemple la classification des données.

3.5.4. Réseau de neurones à multicouche

C'est la structure la plus utilisée; elle correspond à une organisation des neurones en n couches successives la première couche constitue la couche d'entrée et la dernière s'appelle couche de sortie, la propagation de l'information se déroule en sens unique depuis la couche d'entrée vers la couche de sortie.

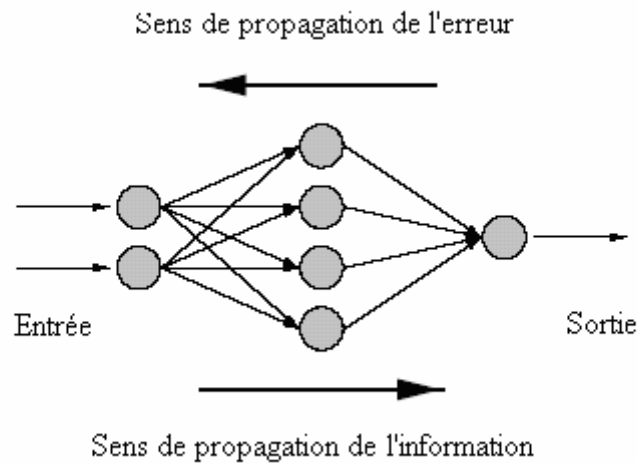


Figure 3.12 : Réseau de neurones à multicouche

La fonction d'activation utilisée pour les neurones peut être n'importe quelle fonction continue et dérivable, on utilise souvent une fonction sigmoïde telle que :

$$g(x) = \frac{1}{1 + \exp(-x)} \quad (3.23)$$

Elle prend pour paramètre la somme pondérée des entrées du neurone

$$e_i = \sum_j W_{ij} a_j + b_i \quad (3.24)$$

La méthode d'apprentissage la plus utilisée est l'algorithme de la rétropropagation du gradient ; il s'agit d'une méthode d'apprentissage supervisée, fondée sur la modification des poids synaptiques du réseau dans le sens contraire à celui du gradient de l'erreur par rapport à ces poids.

3.5.5. Système hybride neuro-flou[42][43][44]

Le système hybride est une représentation d'un système d'inférence flou sous forme d'un réseau de neurones artificiels ; dans cette technique on utilise des fonctions d'activation particulières pour les unités du réseau et une organisation spécifique afin de reproduire les différents éléments constitutifs d'un système d'inférence flou. La structure du réseau est choisie en fonction de la forme de la fonction que l'on cherche à approcher .les dépendances entre les variables d'entrée et de sortie sont déterminées par le choix des règles floues .Les différents paramètres de ces règles constituent des poids synaptiques du réseau et peuvent être modifiés par un algorithme d'apprentissage supervisé, tel que l'algorithme de la rétro- propagation ou Les algorithmes génétiques.

Deux modèles de RNA utilisés pour le codage des systèmes d'inférence flou ; les réseaux à fonction de base radiale et les réseaux à multicouches. Ces derniers sont fréquemment utilisés.

5.5.6. Les réseaux de neuro-flou multicouches

Dans cette structure, on utilise un réseau de neurones artificiels multi couches pour lequel chaque couche correspond à la réalisation d'une étape d'un système d'inférence flou

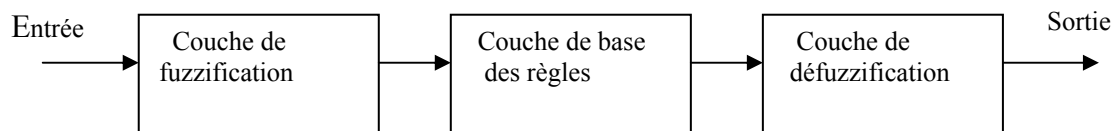


Figure 3.13 : Configuration d'un réseau de neuro-flou multi couche

La couche d'entrée reçoit les valeurs des variables d'entrée, les neurones de la première couche calculent par la fonction de transfert, le degré de vérité des différents sous ensemble flou ; c'est l'étape de fuzzification .La couche suivante correspond au calcul du degré de vérité des prémisses des différentes règles ; les fonctions d'activation utilisée pour ces neurones dépendent des opérateurs présents dans les règles (conjonction ou disjonction) et la logique utilisée. Les neurones de la dernière couche du réseau réalisent la phase de défuzzyfication et fournissent les valeurs de sorties.

CHAPITRE 4 ALGORITHMES GENETIQUES

4.1. Introduction

Les algorithmes génétiques sont des algorithmes d'optimisation et d'apprentissage utilisant des techniques inspirées de la génétique et du processus de l'évolution naturelle (croisements, mutations, sélection, etc.)[33][35].

Les algorithmes génétiques ont la particularité de ne requérir que peu d'information sur la fonction d'évaluation et de pouvoir détecter simultanément plusieurs optima .Ils peuvent traiter des problèmes pour lesquels les solutions potentielles sont situées dans un espace de grande dimension ; et la recherche d'une solution optimale peut être réalisée sans nécessiter de connaissance à priori sur la répartition des solutions dans l'espace d'exploration.

A partir des travaux sur la simulation des structures biologique sur ordinateur réalisés par des biologistes américains, Holland développa les principes fondamentaux des AG dans le domaine d'optimisation mathématique. L'ouvrage de Goldberg [35] a largement contribué à enrichir et mieux faire connaître le domaine des AG.

Le développement de l'informatique a permis d'utiliser des AG dans le cadre de la résolution des problèmes concrets de grande taille suscitant un nouvel intérêt pour cette technique d'optimisation.

Les algorithmes génétiques utilisent une terminologie similaire à celle de la génétique naturelle, ainsi un individu dans une population représente un point de l'espace de données du problème à optimiser, l'individu est identifié par un chromosome constitué d'une chaîne de caractère particulière de taille fixe .Les chromosomes sont eux-mêmes constitués de gènes qui représentent des paramètres ou variables à optimiser .Ces variables sont codées en séquences de bit binaire ou réel pour former d'alphabet similaire à celle du code génétique naturelle que l'on appelle allèle .

Dans le processus de l'évolution naturelle le critère de survie et de reproduction constitue le degré d'adaptation dans le milieu naturel , les algorithmes génétiques associent à chaque solution potentielle la valeur du critère à optimiser que l'on appelle fitness ou critère d'adaptation

4.2. Principe de l'algorithme génétique

La structure générale d'un algorithme génétique est la suivante

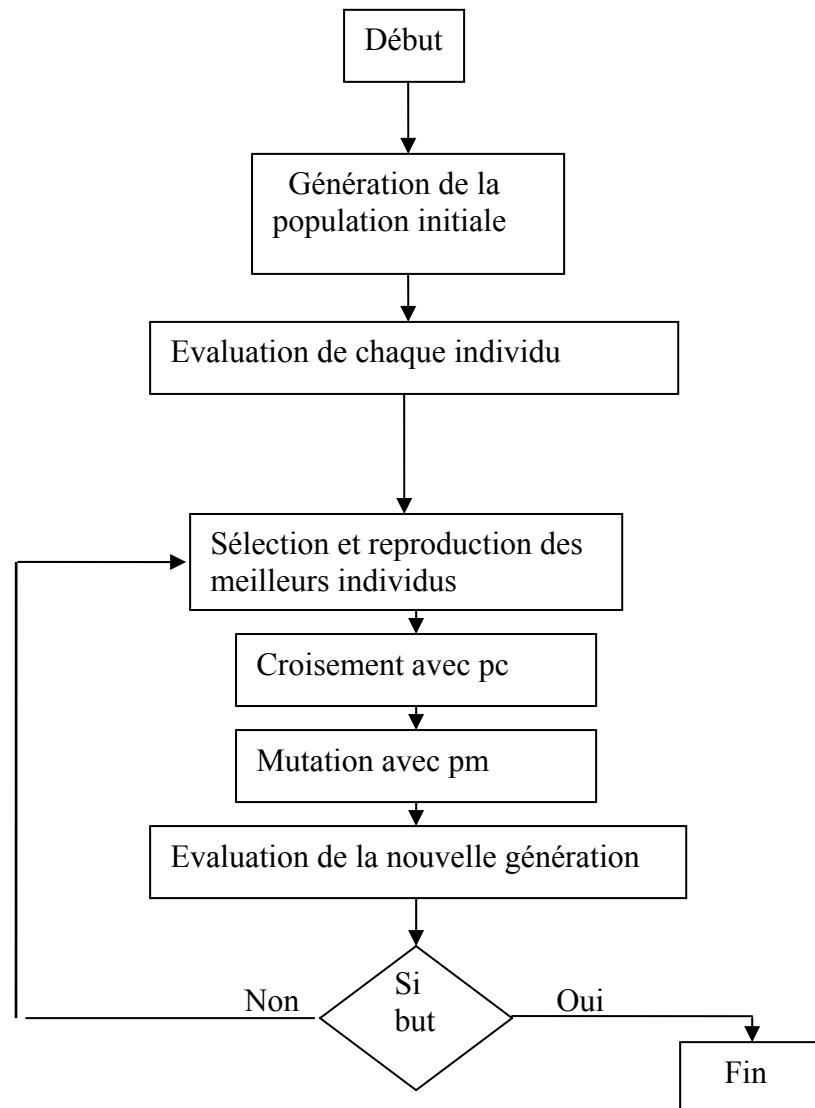


Figure 4.1 : Structure d'un algorithme génétique

On commence par générer une population d'individus de façon aléatoire, on associe à chaque individu la valeur du critère d'optimisation, on applique ensuite les opérateurs de sélection, de croisement et de mutation. La sélection permet de favoriser les meilleurs éléments de la population, le croisement et la mutation assurent une exploration efficace de l'espace de solutions. On génère d'une façon itérative des populations d'individus de cette manière pour atteindre l'optimum. L'utilisation de l'algorithme génétique nécessite les éléments suivants :

- le codage : transformation des données de l'espace de recherche en données utilisables par l'ordinateur en séquence de bit.
- un moyen de créer une population initiale à partir des solutions potentielles à défaut on génère une population aléatoire
- une fonction qui permet d'évaluer l'adaptation d'un chromosome
- les opérateurs de sélection, de croisement et de mutation

4.2.1. Codage des données

Le codage utilisé par les algorithmes génétiques est représenté sous forme de chaînes de bits contenant toute l'information nécessaire à la description d'un point dans l'espace de solutions. En codage classique de Holland, les solutions sont codées par une séquence de caractère binaire, ce codage permet d'utiliser les opérateurs de croisement et de mutation simple (par inversion de bit). Cependant ce type de codage tient généralement peu compte de la structure du problème, alors le codage de données par vecteurs réels est utilisé pour refléter la structure du problème.

4.2.3. Génération de la population initiale

Le choix de la population initiale d'individus aide fortement à la rapidité de l'algorithme. Malheureusement dans de nombreux cas on n'a aucune idée de la position des solutions dans l'espace de recherche, on est contraint alors de générer aléatoirement les individus de la population initiale. Si par contre, on dispose des informations sur la position des solutions du problème, il paraît bien évident de générer les individus dans un sous domaine particulier afin d'accélérer la convergence. La population initiale doit être suffisamment diversifiée et de taille importante pour que la recherche puisse parcourir l'espace de recherche dans un temps limité

c- Evaluation des individus de la population

La fonction d'évaluation est construite à partir du critère que l'on désire optimiser, elle mesure le degré de performance ou le degré d'adaptation de l'individu, elle permet aussi d'avantager ou de désavantager un chromosome vis-à-vis des opérations de sélection et de reproduction, Cette fonction, dont la seule variable est la chaîne de bit caractérisant un individu, est appelée aussi fitness.[33]

d-La sélection

La sélection des parents pour la reproduction consiste à choisir les meilleurs individus selon leurs valeurs sélectives (fitness) ; avant l'application d'opérateurs génétique on procède à l'introduction dans la nouvelle génération par un certain nombre de clones de façon à compléter l'espace des individus éliminés par sélection.

Cette sélection doit tenir compte des individus susceptibles de disparaître, mais porteurs de gènes capables de converger vers des solutions intéressantes après divers croisement et de mutation.

4.3. Mécanisme de sélection

On trouve dans la littérature un nombre important de principe de sélection plus ou moins adapté aux problèmes qu'ils traitent ; les principes de Roulette Wheel selection et de la Stochastic remainder without replacement sélection sont parmi les plus fréquemment appliqués

4.3.1. Roulette proportionnel (Roulette Wheel selection)

Elle consiste à simuler une roulette de casino avec une structure linéaire dont chacun des secteur correspond à un individu de la population, et de surface proportionnelle à sa valeur sélective, ces secteurs sont normalisés sur un segment entre 0 et 1

On se retrouve devant un segment subdivisé en sous segment dont la longueur est proportionnelle au fitness relatives de chaque individu

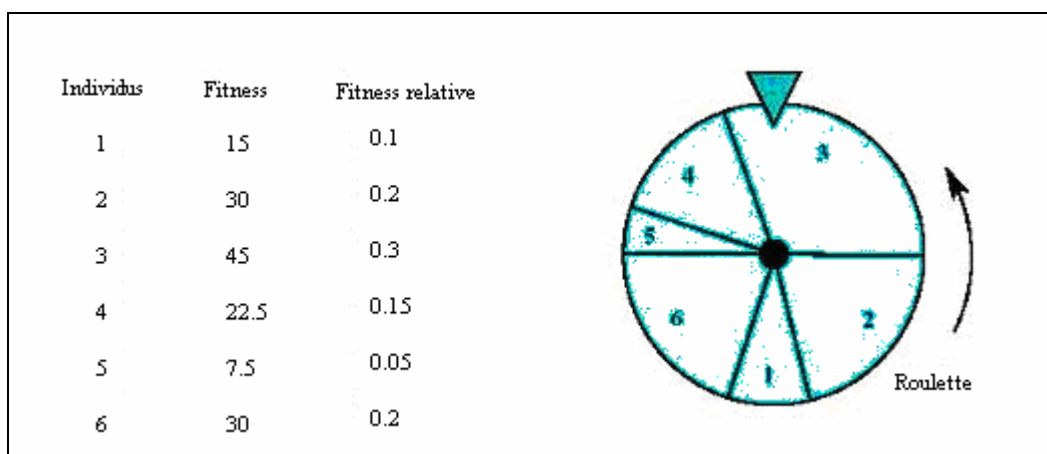


Figure 4.2 : Roulette proportionnel

On tire un nombre aléatoire de distribution uniforme entre 0 et 1, puis on regarde quel est le segment sélectionné, on répète l'opération n fois, n nombre total d'individus

Après chaque tirage, la roue propose un candidat et les individus les plus performants ont plus de chances d'être sélectionnés puisqu'ils occupent une surface plus importante.

4.3.2 Sélection stochastique sans remplacement (Stochastic remainder without replacement)

Le principe de cette sélection consiste à calculer, pour chaque individu de la population le rapport de sa fitness sur la moyenne des fitness r_i puis prendre le nombre de clone égale à la partie entière $e(r_i)$ de r_i , on constate que les individus dont l'adaptation est au dessous de la moyenne ne sont pas retenus, pour compléter la population et garder la diversité on applique le principe de Roulette Wheel sélection sur tous les individus de la population affectés des fitness $r_i - e(r_i)$.

Cette méthode évite que les meilleurs ne prennent rapidement le dessus et garantie une certaine diversité.

4.4.Croisement

Cette étape consiste à choisir aléatoirement deux individus parmi ceux qui ont été sélectionnés (les parents) pour produire deux autres enfants en manipulant la structure des chromosomes

4.4.1. Croisement en un-point

Le croisement d'un point combine deux individus appariés avec une probabilité P_c . Il consiste à tirer au sort une valeur du site de croisement comprise entre 1 et $l-1$ (l étant la longueur d'un chromosome) et à générer deux nouveaux individus en inversant les gènes des deux parents dont le site est compris entre le site du croisement + 1 et l selon la figure .

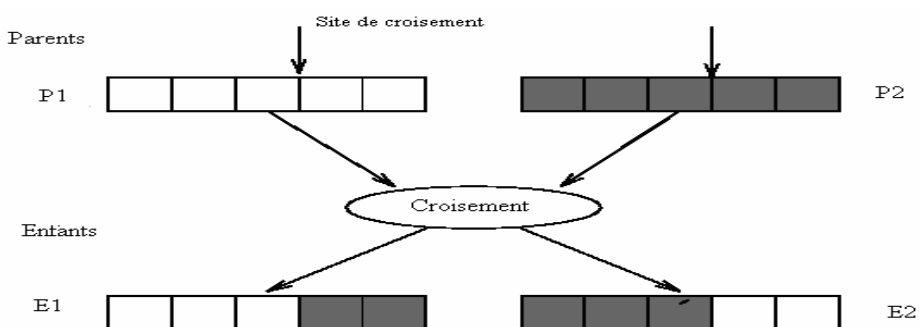


Figure 4.3 : Croisement en un point

Les valeurs de P_c sont classiquement comprises entre 0,6 et 0,9.

4.4.2. Croisement multipoints

Il consiste à déterminer aléatoirement plusieurs sites de croisement

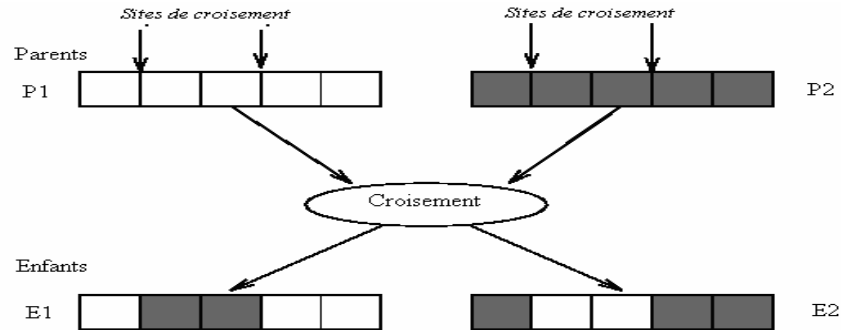


Figure 4.4 : Croisement multipoints

Dans le cas d'un codage par des vecteurs réels, on utilise de préférence d'autres principes, comme le croisement barycentrique. Pour cela, on sélectionne deux gènes $P1(i)$ et $P2(i)$ dans chacun des parents à la même position i que l'on associe en pondération afin de créer deux nouveaux points sur la droite qui relie ces derniers : on crée ainsi $C1(i)$ et $C2(i)$,

$$C1(i) = a P1(i) + (1-a) P2(i)$$

$$C2(i) = (1-a) P1(i) + a P2(i)$$

Où le coefficient a est choisi aléatoirement, par exemple dans $[-0.5, 1.5]$.

4.5. Mutation

Cet opérateur modifie, avec une probabilité P_m faible les allèles des gènes des chromosomes. Lorsque l'alphabet est binaire, la valeur du bit est inversée, dans le cas de données codées par des réels on introduit un bruit aléatoire.

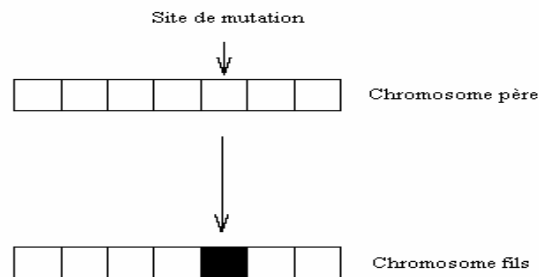


Figure 4.5:Principe de mutation

CHAPITRE 5

SYNTHESE DES COMMANDES ET SIMULATION

La synthèse des commandes de la navigation du dirigeable consiste à considérer chaque condition de vol comme un problème à part, ce qui amène à découpler le système en deux modes ; mode longitudinal, vol dans le plan vertical et le mode latéral, vol dans le plan horizontal.

5.1. Les équations décrivant la dynamique du vol

On découple les équations décrivant la dynamique du dirigeable en deux modes et on néglige la rotation autour de l'axe x ; le roulis: $\varphi = 0$; $\omega_x = 0$.

5.1.1. Mode longitudinal

Les équations décrivant la dynamique du système en mode longitudinal :

$$\begin{aligned}
 m_x \dot{v}_x &= -m_z v_z \omega_y + m_y v_y \omega_z - d_1 v_x + F \cos \alpha \\
 m_z \dot{v}_z &= m_x v_x \omega_y - m_y v_y \omega_x - d_3 v_z + F \sin \alpha \\
 J_y \dot{\omega}_y &= (m_z - m_x) v_z v_x - d_5 \omega_y + l_y F \\
 \dot{x} &= v_x \cos \theta + v_z \sin \theta \\
 \dot{z} &= -v_x \sin \theta + v_z \cos \theta \\
 \dot{\theta} &= \omega_y
 \end{aligned} \tag{5.1}$$

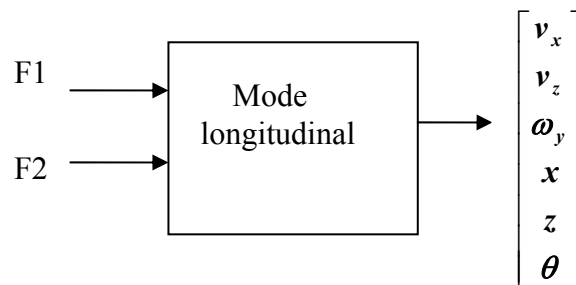


Figure 5.1 : Synoptique du mode longitudinal

$$F1 = F \cos \alpha$$

$$F2 = F \sin \alpha$$

(5.2)

5.1.2. Mode latéral

Les équations décrivant la dynamique du système en mode latéral :

$$\begin{aligned}
 m_y \dot{v}_y &= -m_x v_x \omega_z - d_2 v_y + F_3 \\
 J_z \dot{\omega}_z &= (m_x - m_y) v_x v_y - d_6 \omega_z + F_3 l_x \\
 \dot{x} &= v_x \cos \psi - v_y \sin \psi \\
 \dot{y} &= v_x \sin \psi + v_y \cos \psi \\
 \dot{\psi} &= \omega_z
 \end{aligned} \tag{5.3}$$

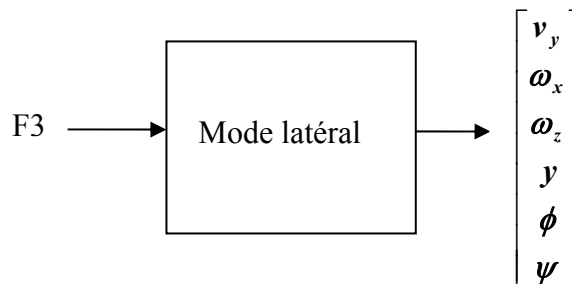


Figure 5.2 : Synoptique du mode latéral

5.2. Stratégie du vol du dirigeable [8]

Le pilotage automatique du dirigeable est assuré par deux commandes une à action latéral et l'autre à action longitudinal ; la stratégie du vol du dirigeable est déduite de la nature de son fonctionnement et se fait comme suit :

a-Phase de décollage :

Pour le décollage ou le changement d'altitude, le dirigeable est commandé en mode longitudinal, le mouvement se produit dans le plan vertical.

b-Phase du maintient :

C'est une étape de préparation à la navigation où deux régulateurs sont mis en œuvre l'un pour contrôler z_d , l'altitude désirée, l'autre pour maintenir la vitesse constante. Ces deux commandes agissent en mode longitudinal

c-Phase de navigation :

On commute vers le mode latéral, la commande agit sur la variation du cap du dirigeable, le mouvement se produit dans le plan horizontal.

d- Phase d'atterrissage :

Pour un cap désiré, la descente est déclenchée, la commande est commutée vers le mode longitudinal.

La phase de navigation correspond à notre objectif de l'étude.

5.3. Simulation de la navigation :[18]

Après stabilisation du dirigeable à une altitude constante z_d et une vitesse de vol appropriée, le dirigeable suit une trajectoire observée par une camera embarquée ; notre trajectoire est repérée par des balises rouges espacées. Ces repères seront captés par la camera et traduites dans le plan image sous forme de points, l'espace entre ces repères est choisi de telle manière que le point suivant apparaît dans l'image pour assurer la navigation par les informations visuelles.

Notre simulation suit les étapes suivantes :

- On génère une trajectoire sous forme d'une courbe fermée, on la discrétise en plusieurs points représentant les balises.
- la technique de commande « Point-to-Point Control » des robot mobiles pour la navigation entre deux points est adoptée [23][47] où chaque point de la trajectoire est considéré comme un sous but, une fois ce point est atteint, ce dernier sera considéré comme point de départ et le point suivant comme but à atteindre et ainsi de suite jusqu'au point final.

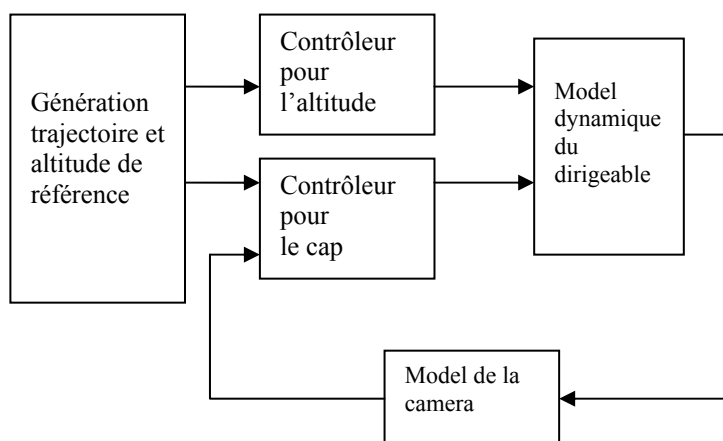


Figure 5.3 : Le synoptique de commande de navigation (architecture de contrôle)

5.4.Synthèse d'un Contrôleur flou pour la commande du cap

La synthèse d'un contrôleur flou dépend principalement des caractéristiques de trois sous systèmes la fuzzification, les règles floues et la défuzzification.

La phase de fuzzification est parfaitement spécifiée lorsque sont définies les fonctions d'appartenance des termes linguistique décrivant les entrées ; les règles floues sont issues d'une analyse des connaissance fournies par un expert humain et de certaines caractéristiques du système commandé ; la défuzzification permet la transformation d'une fonction d'appartenance en une valeur de commande.

Dans notre application nous avons adopté un contrôleur flou à deux entrées et une seule sortie : l'erreur du cap $\Delta\psi$; le taux de variation de l'erreur $\dot{\Delta\psi}$, la sortie agit sur f3, force du propulseur attaché à la queue du dirigeable qui commande l'orientation du cap

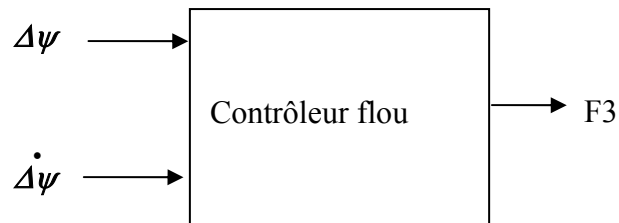


Figure 5.4 : Synoptique d'un contrôleur flou pour la commande du cap

Pour concevoir ce contrôleur, on a simulé une trajectoire de forme droite de longueur finie, cette droite est discrétisée en plusieurs points espacés de telle manière au minimum deux points successifs peuvent être observés par la camera selon la figure suivante :

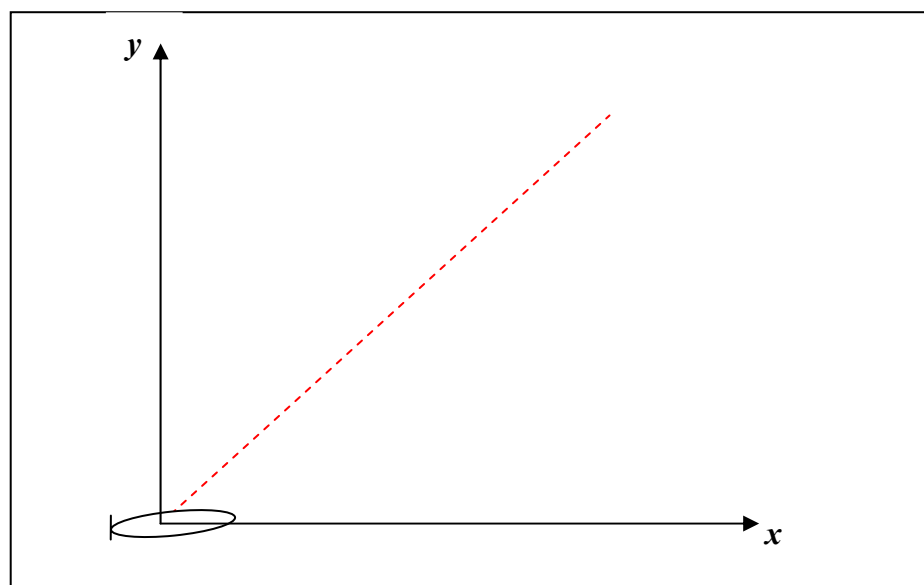


Figure 5.5: Trajectoire à suivre

On désire changer le cap d'une valeur quelconque (ici $\psi = 0$) à une valeur désirée $\psi = \pi/4$ pour suivre une trajectoire de droite ; la vitesse doit être faible lors du changement de cap et en vol de croisière elle atteint la vitesse désirée ; cette technique est décrite en [27] où la valeur de vitesse désirée est appliquée à une fenêtre de Blackman

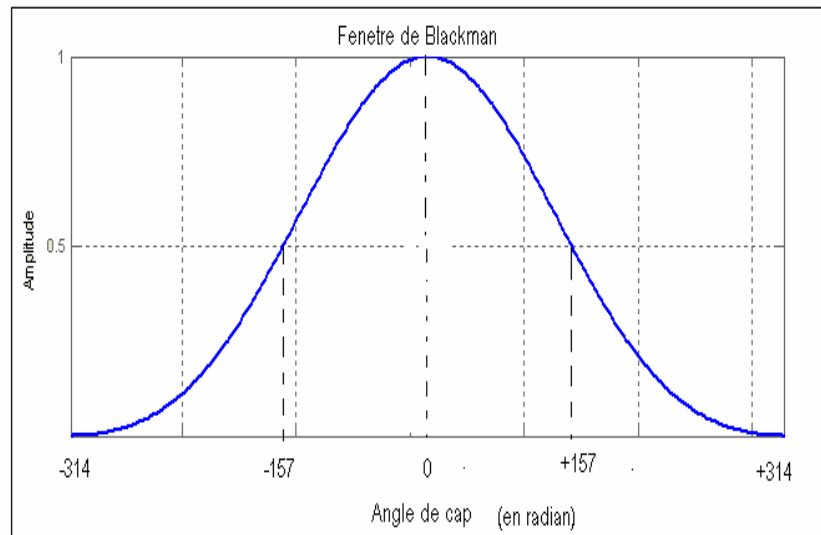


Figure 5.6 : Fenêtre de Blackman

$$\Delta\psi = \arctan g\left(\frac{y_d - y_c}{x_d - x_c}\right) \quad (5.4)$$

$$v_x = v_{xdésirée} W(\Delta\psi) \text{ application de la fenêtre Blackman.} \quad (5.5)$$

x_d, y_d : représentent la position désirée du dirigeable

x_c, y_c : la position courante du dirigeable

W : fenêtre de Blackman

La régulation du suivi de la trajectoire sera réalisée à partir d'informations visuelles où $\Delta\psi$ et $\dot{\Delta\psi}$ seront traduites dans le plan d'image.

5.4.1. Algorithme de commande (figure 5.7)

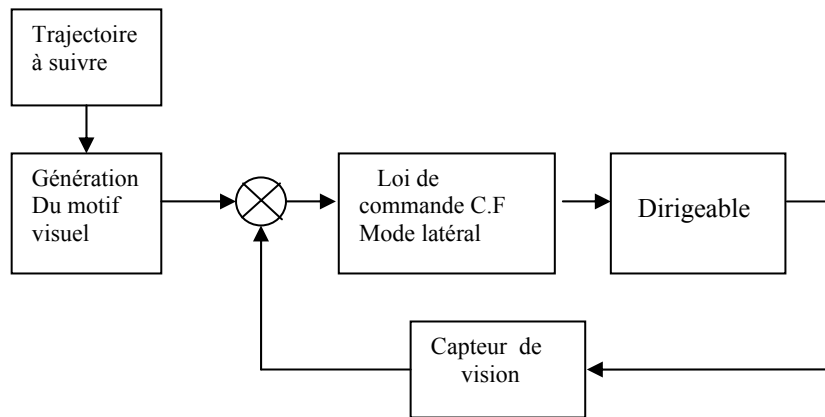


Figure 5.7 : Synoptique de la commande du cap par la vision

a- la trajectoire à suivre et génération du motif visuel :

La trajectoire est représentée par des points (x_f, y_f) simulant la droite à suivre dans le plan du repère référentiel, ces points sont transformés dans le repère lié au ballon par la matrice de changement de repère :

$$T = \begin{bmatrix} R & t \\ [0]_{1 \times 3} & 1 \end{bmatrix} \quad (5.6)$$

Avec R matrice de rotation

$$R = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.7)$$

Et t vecteur de translation

$$t = [x_b \quad y_b \quad z_b]^T \quad (5.8)$$

Les points exprimés dans le repère ballon vont être transformés dans le repère camera puis par la projection se transforment en motif dans le plan image selon (2.9)

$$p = K T_p G P \quad (5.9)$$

Avec K matrice des paramètres intrinsèques de la caméra

$$K = \begin{bmatrix} \alpha_u & 0 & 0 \\ 0 & \alpha_v & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.10)$$

Et

$$\mathbf{T}_P = \frac{f}{z_c} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (5.11)$$

Ces motifs constituent les primitives désirées

b- Module de loi de commande :

Dans cette partie on élabore une force de contrôle du cap à partir de l'erreur issue du comparateur ; le bloc dirigeable est remplacé par le model mathématique du mode latéral. Le mouvement du dirigeable est transféré à la camera embarquée

c- Capteur de vision

A l'aide du torseur cinématique de la camera et la matrice d'interaction, on calcule les primitives actuelles décrivant la position et l'orientation du ballon par rapport à ce point à l'aide de l'expression suivante :

$$\dot{\mathbf{f}} = \mathbf{L}_s^T \mathbf{T} \quad (5.12)$$

où \mathbf{f} : Vecteur des primitives

\mathbf{L}_s^T : Matrice d'interaction

\mathbf{T} : Torseur cinématique de la caméra

A la sortie de ce bloc on récupère les coordonnées du point (u, v) représentant une balise observée par la caméra, qui vont être comparées avec les primitives désirées.

La commande est élaborée à partir des informations visuelles ; le cap du dirigeable est exprimé dans le plan d'image selon l'illustration suivante :

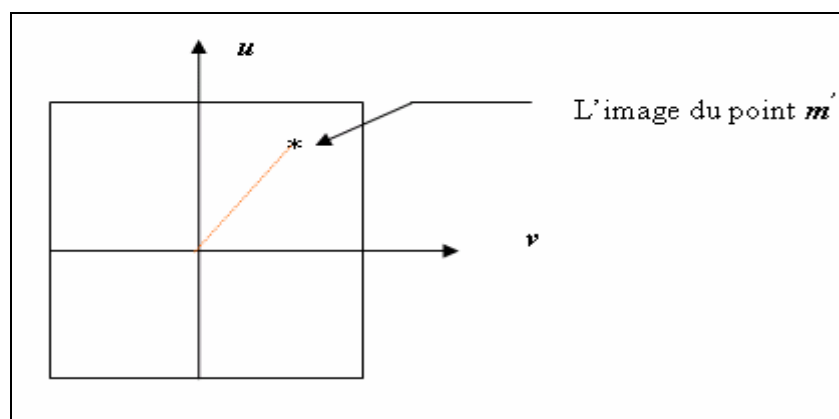


Figure 5.8 : L'apparition du repère sur le plan image

La commande consiste à amener le point m' vers l'axe v , on contrôle ainsi la pente de la droite reliant le point m' à l'origine. Le contrôleur adopté est de type Mamdani à deux entrées et une seule sortie

Les entrées sont des informations visuelles : e et ∇e

e : erreur du cap dans le plan image

$$e = \arctan g \left(\frac{v_d - v_c}{u_d - u_c} \right) \quad (5.13)$$

v_d, u_d : primitives désirées

v_c et u_c : primitives courantes

∇e = taux d'erreur *sur le cap*

S : La sortie constituant la force F_3 charger de la commande de l'orientation du dirigeable.

Lors de la synthèse d'un contrôleur flou, les caractéristiques tel que la position et la forme de termes linguistiques qualifiant les entrées et la sortie, la base des règles floues doivent être choisies de façon intuitive à cause de l'absence d'une expertise.

On se basant sur le fonctionnement de notre système et selon la l'objectif désiré on a simulé un contrôleur flou sous Matlab.

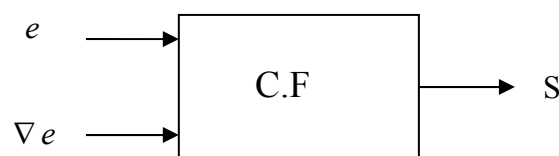


Figure 5.9 : Contrôleur flou adoptée

5.4.2. Résultats de simulation :

a- Les fonctions d'appartenance des différents termes linguistiques qualifiant les entrées et la sortie du contrôleur.

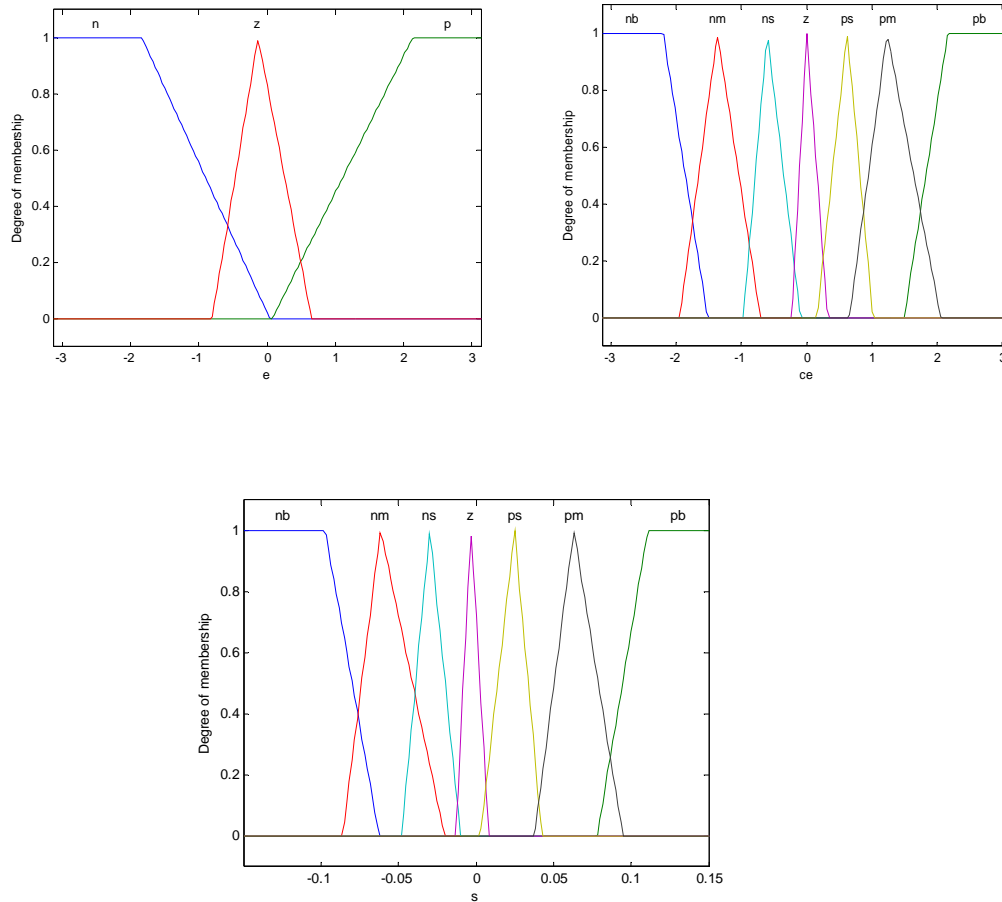


Figure 5.10 : Fonctions d'appartenance du contrôleur

b- Dans les tableaux ci-dessous nous avons représenté les conclusions des règles floues

e\ce	NG	NM	NP	EZ	PP	PM	PG
N	NS	NM	NB	NB	PB	NS	PM
EZ	NS	NM	NB	EZ	PB	NS	PM
P	NM	NS	EZ	PS	PB	NM	PM

Tableau 5.1 : Base des règles

c-La réponse de la commande du cap du ballon dirigeable et le suivi de la trajectoire.

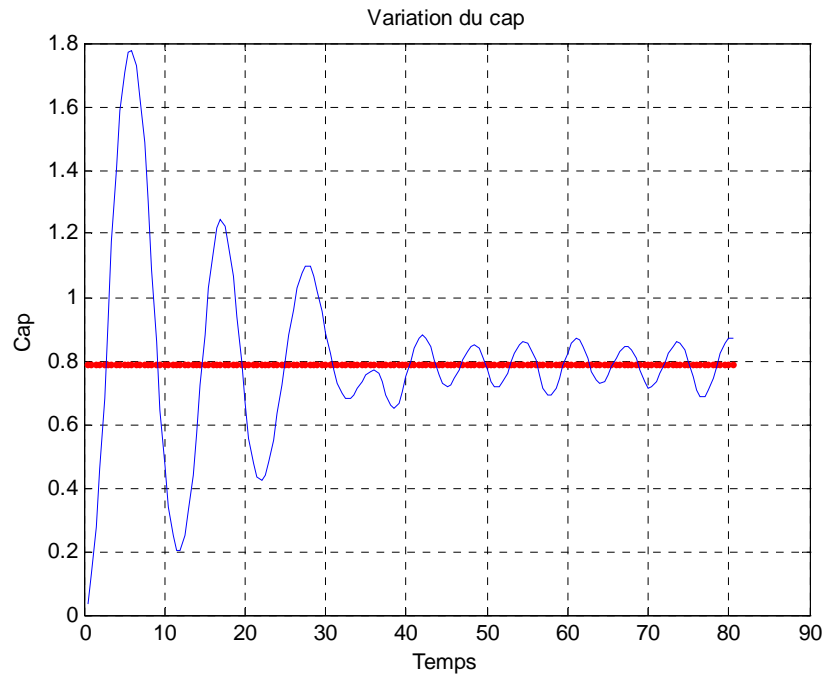


Figure 5.11 : Variation du cap en fonction du temps

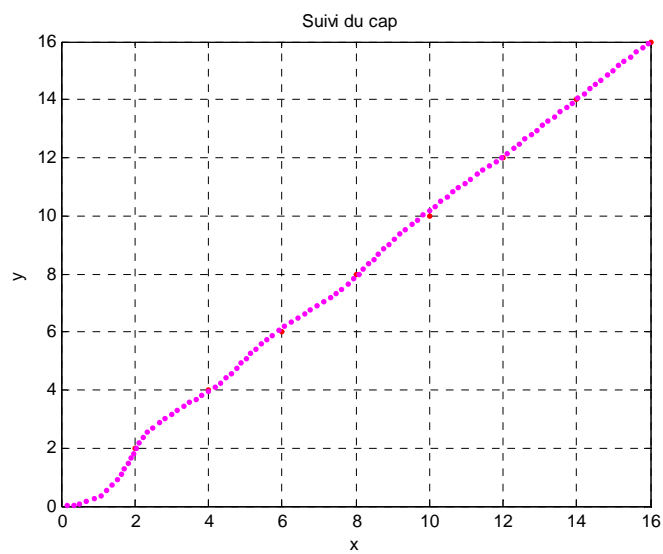


Figure 5.12 : Le suivi de la trajectoire

Le résultat obtenu de l'application du contrôleur figure 5.11 et 5.12 montre que le changement de cap vers l'orientation désirée s'est effectué après 35 secondes environ et après avoir parcouru une distance de 4.5m ;on constate aussi que le dirigeable avance avec des petites oscillations d'ordre 10%.

d- Navigation du dirigeable :

Selon la méthodologie décrite en 5.3, On simule un circuit fermé de forme rectangulaire pour voir l'évolution du dirigeable lors de la navigation.

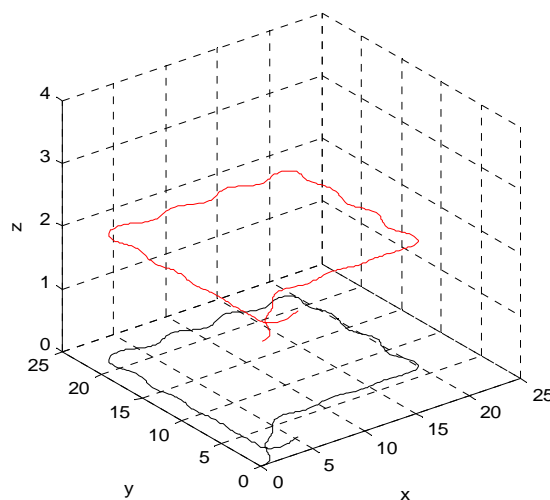


Figure 5.13 : Navigation du dirigeable le long d'une courbe fermée

-La courbe en noire représente l'ombre de la trajectoire

-La courbe en rouge représente la trajectoire en 3D.

Interprétation :

Le suivi de la trajectoire est grosso modo réalisé, mais il y a toujours des ondulations qui peuvent influencer sur le temps du parcours et la consommation d'énergie. L'amélioration de cette trajectoire dépend du choix des éléments constituant la structure du contrôleur flou. Le choix optimal des fonctions d'appartenance et des règles floues consiste à trouver les positions relatives de chaque terme linguistique au sein d'une variable linguistique et à retenir les règles qui s'avèrent pertinentes. Cette façon de faire requiert du temps, de l'expérience et de l'habileté de la part du chercheur. Toutefois, un réglage des paramètres du contrôleur flou par un algorithme génétique peut être envisagé afin d'en améliorer ces performances. Cette stratégie fait l'objet du paragraphe suivant.

5.5. Synthèse d'un contrôleur flou par Algorithme Génétique

Les méthodes permettant d'optimiser les différents paramètres d'un contrôleur flou ont été décrites dans la littérature [19] [24]. Elles sont pour la plupart basées sur des techniques d'apprentissages qui permettent de définir de façon itérative le meilleur jeu de paramètres du contrôleur. Les chercheurs se sont concentrés principalement sur les approches suivantes :

- Optimisation des fonctions d'appartenance
- Optimisation des règles floues
- Optimisation simultanée des fonctions d'appartenance et les règles floues

Pour optimiser les fonctions d'appartenance on a utilisé un algorithme génétique standard où le nombre des termes linguistiques est fixé à priori, les fonctions de type triangulaire radial et gaussien sont codés dans un chromosome [47]. Pour éliminer les fonctions d'appartenance ou des règles inutiles on ajoute un code binaire qui constitue les bits de contrôle.

L'approche d'optimisation des règles floues est décrite [50], les algorithmes génétiques utilisent le codage binaire pour chaque règle. La méthode d'optimisation qui prend en compte simultanément les fonctions d'appartenance et les règles floues est proposée dans [48] où les différents paramètres sont codés dans un chromosome unique sous forme binaire.

Tandis que d'autres auteurs optimisent un contrôleur flou comportant deux entrées et une seule sortie où le chromosome utilisé comporte des chaînes exprimées en base sept ce qui leur permet de représenter au mieux sept fonctions d'appartenance et d'autres utilisent un chromosome codé en base cinq pour optimiser un contrôleur flou comportant deux entrées et une sortie et 25 règles floues [49]

5.5.1. Méthode de codage des fonctions d'appartenance [24]

Il est possible de représenter les fonctions d'appartenance sur un chromosome

5.5.1.1. Codage des fonctions d'appartenance triangulaires

Soit la fonction d'appartenance triangulaire dans un univers de discours fini $[a, b]$; figure 5.14

Le codage sera de la forme suivante

$C(x_i)$	$C(x_i+x_1)$	$C(x_i-x_2)$
----------	--------------	--------------

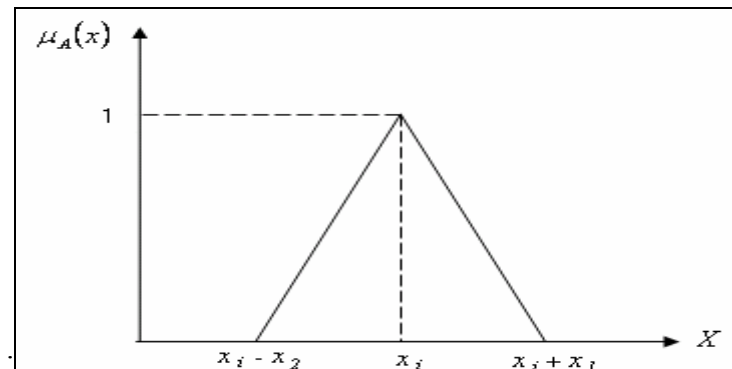


Figure 5.14 : Forme de la fonction d'appartenance triangulaire

C représente le type de codage (binaire ; base n ; nombre réel)

Si le triangle est symétrique le chromosome contiendra deux gènes

$C(x_i)$	$C(x)$
----------	--------

Avec x_i l'abscisse du sommet et x l'ouverture de la base

5.5.1.2. Codage des fonctions d'appartenance trapézoïdales

Une fonction d'appartenance trapézoïdale dans un univers de discours fini $[a, b]$ peut être défini selon la figure 5.15.

Le codage sera de la forme suivante

$C(x_1)$	$C(x_2)$	$C(x_3)$	$C(x_4)$
----------	----------	----------	----------

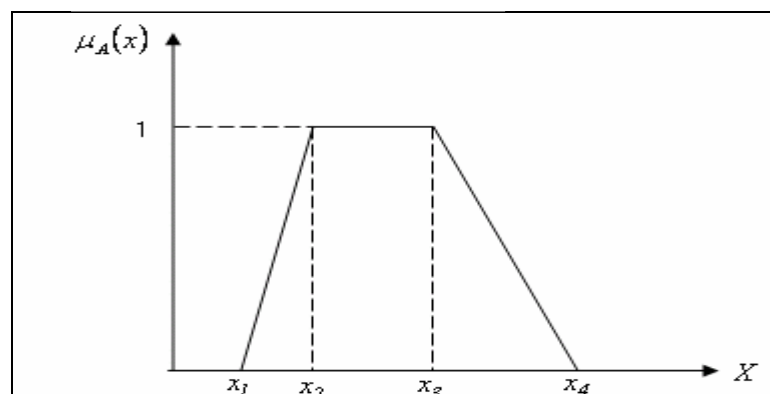


Figure 5.15 : Forme de la fonction d'appartenance triangulaire

5.1.1.3. Codage des fonctions d'appartenance gaussiennes

Une fonction d'appartenance gaussienne dans un univers de discours fini $[a, b]$ peut être exprimée sous la forme :

$$\mu_A(x) = e^{-y} \quad \text{Avec} \quad y = (x - x_1)^2 / \sigma^2 \quad (5.14)$$

Cette fonction est bien définie par deux paramètres x_1 et σ . Avec $x_1 \in [a, b]$ et σ prend des valeurs dans un intervalle plus restreint qui est fonction de l'application.

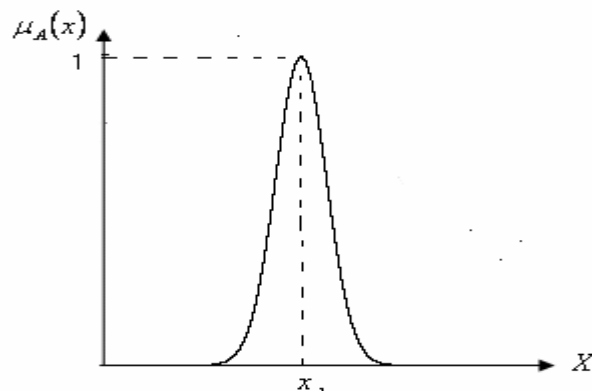


Figure 5.16 : Forme de la fonction d'appartenance gaussienne

La représentation d'une fonction d'appartenance gaussienne est alors donnée par le codage suivant

$C(x_1)$	$C(\sigma)$
----------	-------------

5.5.2. Contraintes sur les variables linguistiques

La variable linguistique utilisée d'un contrôleur flou doit satisfaire les contraintes suivantes :

- limiter autant que possible la zone de recouvrement entre termes linguistiques
- au voisinage du max de chaque fonction d'appartenance, il faut que les autres fonctions d'appartenance prennent des valeurs aussi faibles que possible

5.5.3. Codage des règles floues :

Si le contrôleur flou comporte deux entrées qualifiées chacune par n_1 et n_2 termes linguistiques et sa sortie est qualifiée par n_3 termes, on peut définir au maximum $n_1 \times n_2 \times n_3$ règles floues dont chaque règle possède deux prémices et une conclusion de la forme

Si X1 est A1 et X2 et A2 alors Y est b

Pour représenter la totalité des règles on dresse un tableau des conclusions correspondantes à toutes les configurations possibles des prémisses et la conclusion d'une règle est repérée par un nombre selon le tableau suivant :

	A_{21}	A_{22}	...	A_{2i_2}	...	$A_{2N_{te}}$
A_{11}	R_{11}	R_{12}	...			$R_{1N_{te}}$
A_{12}	R_{21}	R_{22}	...			$R_{2N_{te}}$
⋮				⋮		
A_{1i_1}		...		$R_{i_1i_2}$...	
⋮				⋮		
$A_{1N_{te}}$	$R_{N_{te}1}$...			$R_{N_{te}N_{te}}$

Tableau 5.2 : Tableau des conclusions

Pour constituer la représentation de l'ensemble des règles floues sous forme d'un chromosome, on réorganise le tableau en séquence de valeurs en le balayant ligne par ligne. Selon la configuration suivante $n = n1 \times n2$

$CR11$	$CR22$	$CRn1n2$
--------	--------	-------	------	----------

Où les R_{ij} peuvent prendre n^3 configurations possibles. Généralement le codage se fait en binaire ou en base n.

Exemple d'un codage d'une base de règle en base 5

	NM	NP	EZ	PP	PM
NM	1	4	3	2	1
NP	5	2	4	3	2
EZ	1	2	4	5	1
PP	4	3	1	2	2
PM	1	1	3	4	5

Tableau 5.3 : Base des règles codée en base 5

Avec NM=1 ; NP=2 ; EZ=3 ; PP=4 ; PM=5

Le chromosome aura la forme suivante

1	4	3	2	1	5	2	4	3	2	1	2	4	5	1	1	3	1	2	2	1	1	3	4	5
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Figure 5.17 : Exemple d'un chromosome codé en base cinq

Méthode de codage adoptée

Dans notre étude on a utilisé trois sous-ensemble flou pour la répartition d'entrée (e) et sept pour le deuxième espace d'entrée (ce), la base des règles est constituée par 21 règles qui représentent toute les combinaisons possibles des prémisses .

Les fonctions d'appartenance choisies sont des triangles symétriques ; donc deux paramètres suffisent pour leurs descriptions à savoir la position du sommet et l'ouverture de la base.

les sommets des fonctions d'appartenance associées aux centre et aux extrémités de l'univers de discours sont fixées à priori , par conséquent trois allèles sont affectés à la première variable d'entrée(e),onze allèles pour chaque variable d'entrée (ce) et de sortie.[39]

Ainsi le codage nécessite 46 allèles, les 21 premiers sont réservés à la base des règles, les autres représentent les paramètres des fonctions d'appartenance d'entrée et de sortie

On utilise un codage entier pour coder les informations chromosomiques des individus de la population de l'algorithme génétique où chaque individu représente un contrôleur flou.

Les premiers allèles représentant la base des règles sont des éléments de l'ensemble $\{1,2,\dots,7\}$; chaque nombre représente l'ensemble flou de sortie à utiliser pour une règle indiquée (où : NG =1, NM =2, NP =3, EZ =4, PP =5, PM =6, PG =7).

Les derniers allèles représentant la base de données sont aussi des éléments de l'ensemble $\{1,2,\dots,7\}$, mais cette fois-ci, chaque nombre représente une position de pique ou une ouverture de base pour une fonction d'appartenance indiquée. Alors tous les allèles d'un chromosome prennent leurs valeurs dans l'ensemble $\{1,2,\dots,7\}$

-Représentation du chromosome

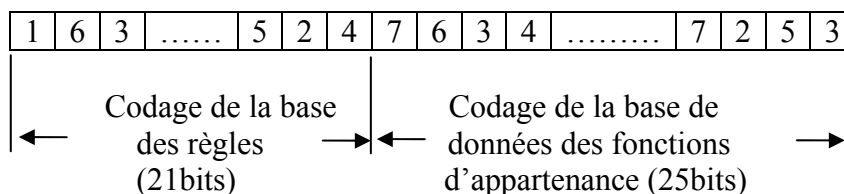


Figure 5.18 : Structure du chromosome codé en base sept

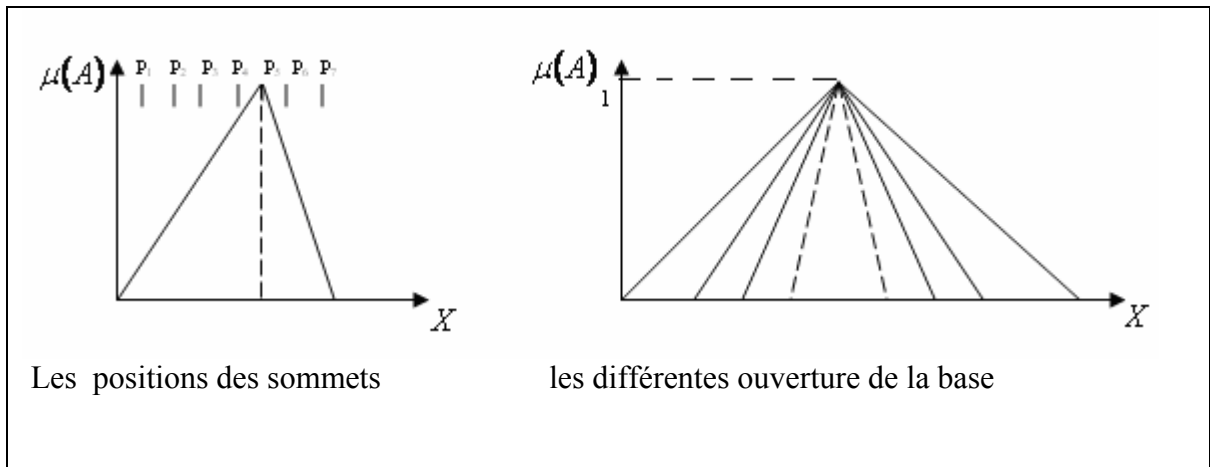


Figure 5.19: Représentation des pics et des ouvertures des fonctions d'appartenance

5.5.4. Fonction d'adaptation

La fonction d'adaptation ou fitness de l'algorithme génétique est construite de telle sorte qu'elle présente un minimum global lorsque les sorties calculées sont comparées aux valeurs désirées. On utilise la somme des différences au carré entre les deux séries de données que l'on appelle la somme des erreurs quadratiques

$$\text{Fitness} = \sum er^2 \quad (5.15)$$

5.5.5. Opérateurs génétiques utilisés

5.5.5.1. Sélection :

Le rôle de la sélection est de choisir parmi tous les individus d'une population des parents qui assureront la reproduction. Ce choix est réalisé par tirage au sort parmi les individus, en tenant compte d'une probabilité de sélection affectée à chacun d'eux. Un individu a d'autant plus de chances d'être sélectionné que sa fonction d'adaptation est meilleure. La méthode utilisée pour sélectionner les individus assurant la reproduction est la Stochastic remainder without replacement sélection décrite dans (4.3)

5.5.5.2. Croisement

L'étape de sélection a permis de choisir parmi une population les individus les plus aptes à se reproduire. Ensuite, ces individus sont regroupés par paires qui constitueront les parents de la génération suivante. Pour chaque paire de parents, un tirage au sort de probabilité $p_c=0.7$ détermine le point de croisement situé à la position p comprise entre 1 et $l=46$, où l est la longueur du chromosome. Le croisement s'effectue à la position p selon la procédure décrite en (4.4)

5.5.5.3. Mutation

L'opérateur de mutation s'effectue avec une probabilité $P_m=0.04$. La valeur de remplacement est prise aléatoirement de l'ensemble $\{1,2, \dots, 7\}$

5.5.6. Evolution de la population

Les phases de sélection, reproduction et de mutation ont permis de créer une nouvelle population en partant de la population initiale, la nouvelle population est globalement plus performante que la population initiale, Le processus d'évolution se poursuit ,en réitérant les opérations précédentes de façon à créer une succession de populations. Dans notre application le processus des algorithmes génétique suit l'organigramme de la figure (5.20)

5.5.7. Caractéristique de l'A.G

- Taille de la population : 30 individus
- Nombre de gènes codants la base des règles : 21 bits
- Nombre de gènes codants la base de données : 25 bits
- Probabilité de croisement : 0.85
- Probabilité de mutation : 0.02
- Type de codage : base 7

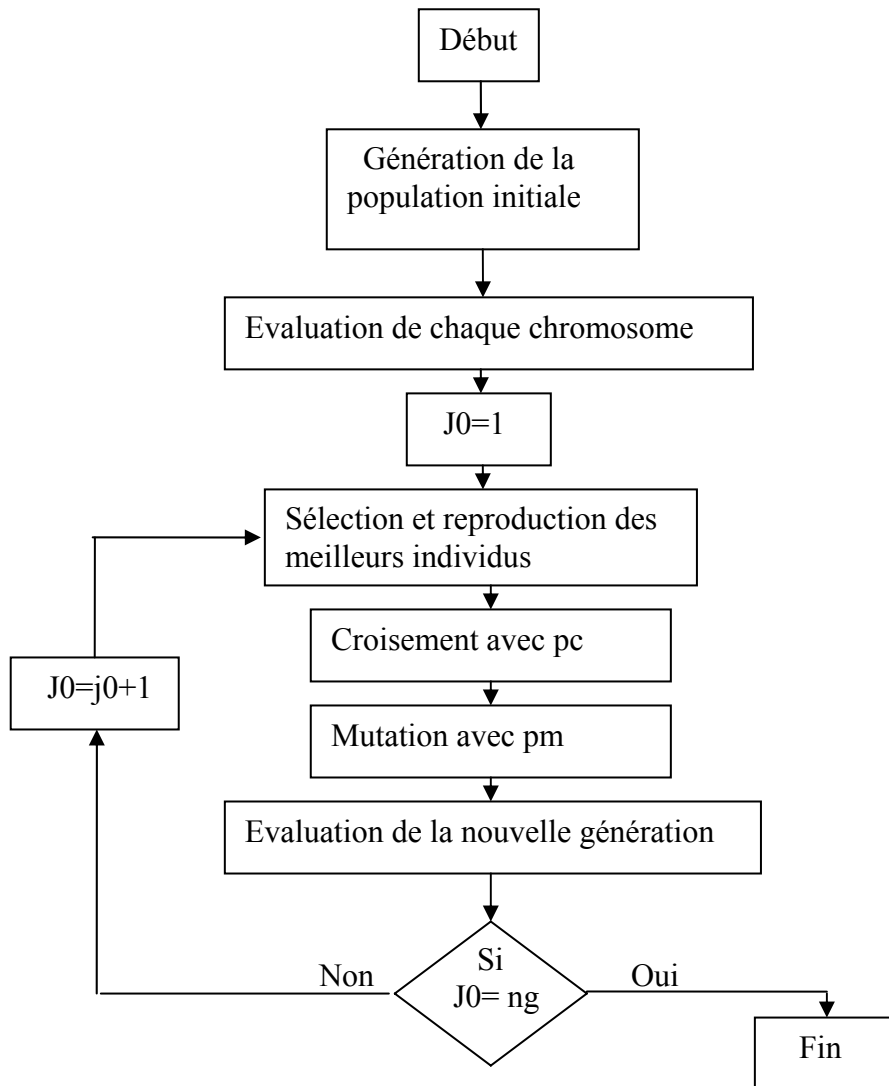


Figure 5.20 : Structure de Algorithme Génétique

5.5.8. Résultats de simulation

a- Les fonctions d'appartenance des différents termes linguistiques représentent les entrées du contrôleur et sa sortie:

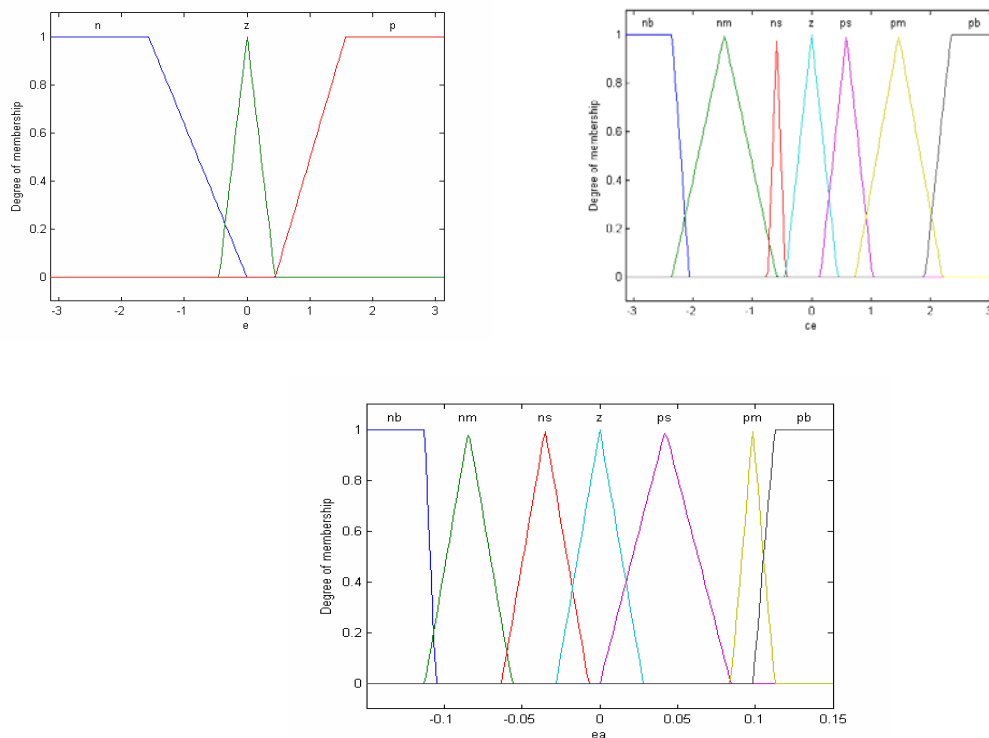


Figure 5.21 : Les fonctions d'appartenance.

b-Le tableau ci-dessous représente les conclusions des règles floues définies par le processus d'optimisation :

e\ce	NG	NM	NP	EZ	PP	PM	PG
N	NB	PM	NS	NB	NB	PS	PM
EZ	PB	PM	NS	PS	NB	NB	PB
P	PB	NB	NB	PB	NM	PS	PM

Tableau 5.4 : Tableau de la base des règles

c - Courbe représentant la variation du cap du dirigeable obtenue par la régulation dans le plan image et le suivi de la trajectoire :

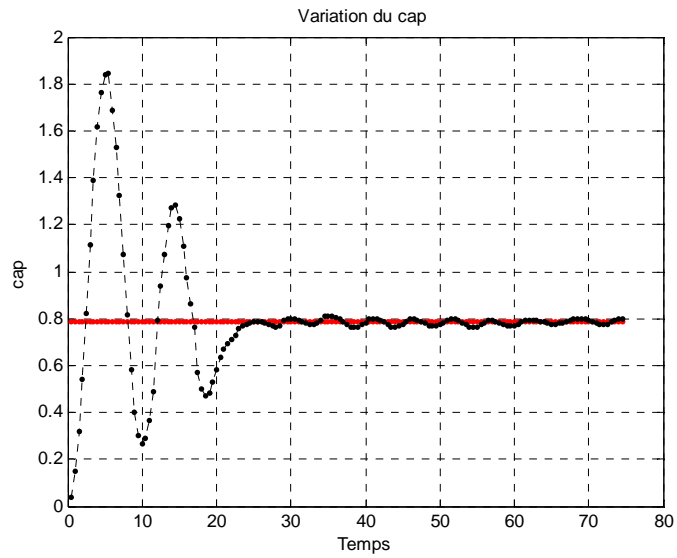


Figure 5.22: Variation du cap

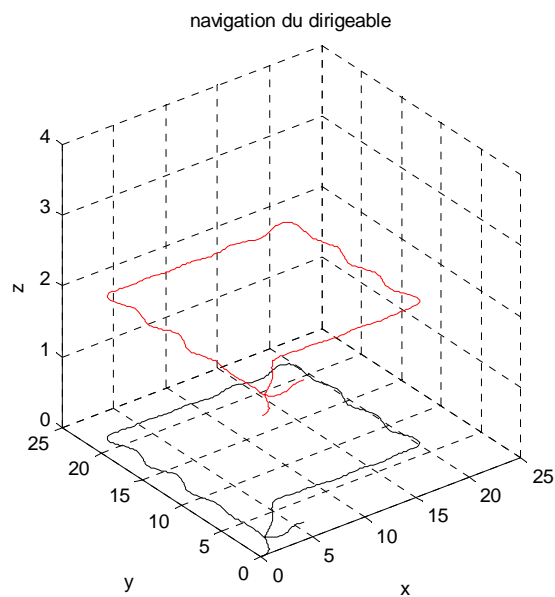


Figure 5.23 : Navigation du dirigeable

-La courbe en noire représente l'ombre de la trajectoire

-La courbe en rouge représente la trajectoire en 3D

d- Évolution de l'AG

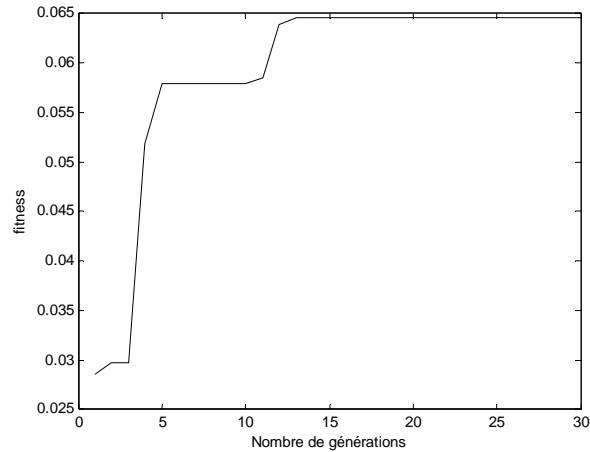


Figure 5.24 : Variation de la fonction d'adaptation en fonction du nombre de générations

Interprétation :

Le changement de cap s'est effectué après 20 secondes, après avoir parcouru une distance de 4.5m ; il y a diminution en temps de réponse ; on constate aussi que le dirigeable avance avec des oscillations faibles par rapport au résultat précédent.

La synthèse d'un contrôleur flou par des algorithmes génétiques a donné des performances supérieures à celle du régulateur conçu d'une manière intuitive, avec un bon contrôle de position et de petites oscillations du cap. Ces résultats montrent les performances de notre contrôleur étudié comparativement à la simulation réalisée par les auteurs de l'article [2] où le dirigeable suit une trajectoire sous forme spirale cubique avec une commande basée sur la position ; le ballon avance et tourne sur lui-même [APPENDICE B].

Le processus d'optimisation n'a pas tenu compte du nombre des fonctions d'appartenance et des règles floues car ce nombre a été fixé a priori. Cependant on a souvent constaté que certains termes linguistiques et des règles floues deviennent inutiles dans la pratique.

Leur élimination totale simplifierait la structure du contrôleur flou sans incidence majeure sur les performances. La méthode qui permet d'éliminer les termes linguistiques et les règles inutiles utilise les algorithmes génétiques hiérarchiques. Leur application fera l'objet du paragraphe suivant.

5.6. Synthèse d'un contrôleur flou par les algorithmes génétiques hiérarchiques.

Introduction

Dans la synthèse précédente, le nombre des fonctions d'appartenance est choisi à priori et le nombre de règles est toujours égale à $n_1 \times n_2$ avec n_1 et n_2 constituent le nombre des termes linguistiques des entrées du contrôleur ; l'optimisation était basée sur la position et l'ouverture des fonctions d'appartenance et la qualité de la base des règles.

On peut même concevoir un contrôleur avec le minimum des termes linguistiques et l'élimination des règles inutiles ; pour le faire on fait appel aux algorithmes génétiques hiérarchiques. Le processus d'optimisation par les HGA consiste à trouver les positions relatives de chaque terme des fonctions d'appartenance au sein d'une variable linguistique et d'éliminer celle qui sera inutile alors que l'optimisation des règles floues consiste à sélectionner parmi toutes les règles possibles celles qui s'avèrent pertinentes.

5.6.1. Description d'un algorithme génétique hiérarchique

Dans ces algorithmes le chromosome est constitué de deux types de gènes, à savoir les gènes de contrôle et les gènes de structure ou gène de paramètres.

La relation entre ces deux gènes dans le même chromosome est de nature de contrôle ; le choix des gènes de paramètres est tributaire de l'état du gène de contrôle ; ces derniers sont codés en binaire où le 0 équivaut à la désactivation du paramètre et le 1 à sa validation ; comme est illustré dans cette figure :(Le nombre 5 est activé, Le nombre 7 est désactivé

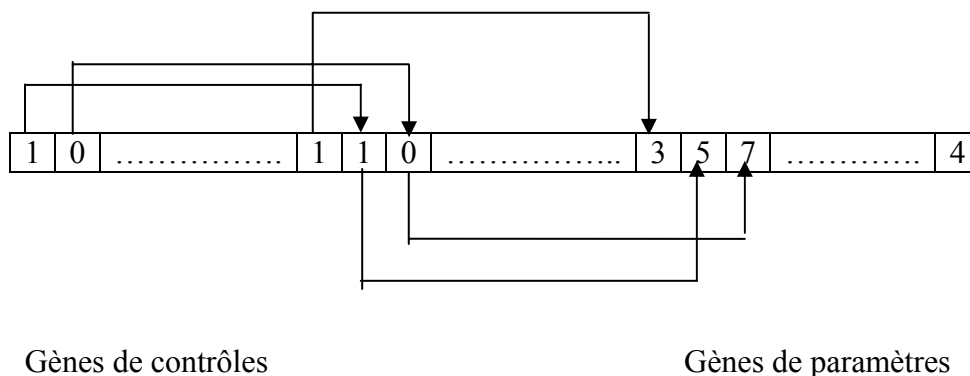


Figure 5.25 : Chromosome de algorithmes génétiques hiérarchiques

Les algorithmes génétiques hiérarchiques permettent de déterminer le nombre et la position optimale des fonctions d'appartenance et le choix des règles floues, sans définir à

priori la structure des sous-ensemble flous .Le processus des AGH est similaire à celui des AG classiques, mais les opération de croisement et mutation s'applique séparément sur les gènes de contrôle et les gènes de structure, comme l'illustre la figure suivante :

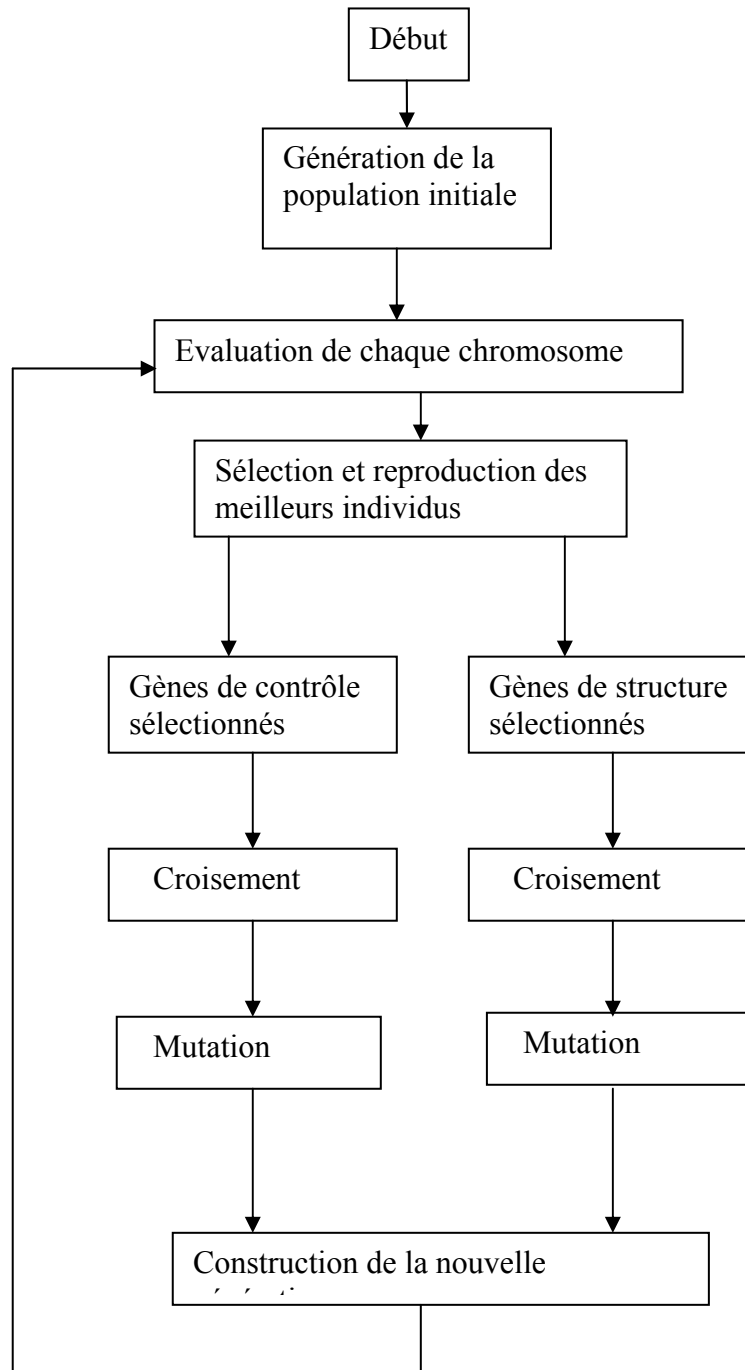


Figure 5.26 : Structure de l'algorithme génétique hiérarchique

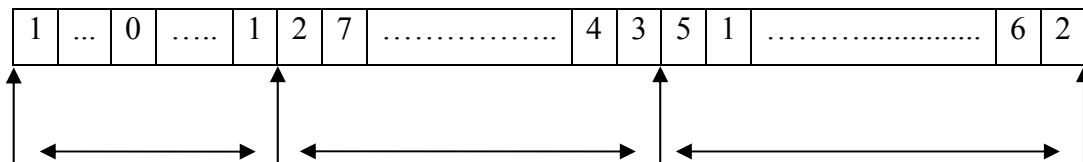
5.6.2. Structure de l'algorithme génétique hiérarchique adopté

Dans notre application nous avons choisi le nombre maximum des fonctions d'appartenance égale à sept, le nombre max des règles floues à optimiser est égale à 7×7 . La structure du chromosome adoptée utilise deux gènes

-gènes de contrôle sous forme de bits binaires

-gènes de structure ou de paramètres dont les bits représentent les fonctions d'appartenance et la base des règles codées en entier de 1 à 7

Le chromosome a la structure suivante



Gènes de contrôle Gènes de la base des règles Gènes des fonctions d'appartenance

(21bits)

(49 bits)

(33bits)

Figure 5.27: Structure du chromosome adopté

Dans le processus de cet algorithme on rencontre des cas où les fonctions d'appartenance invalides laissent un espace vide dans l'univers de discours de la forme suivante :

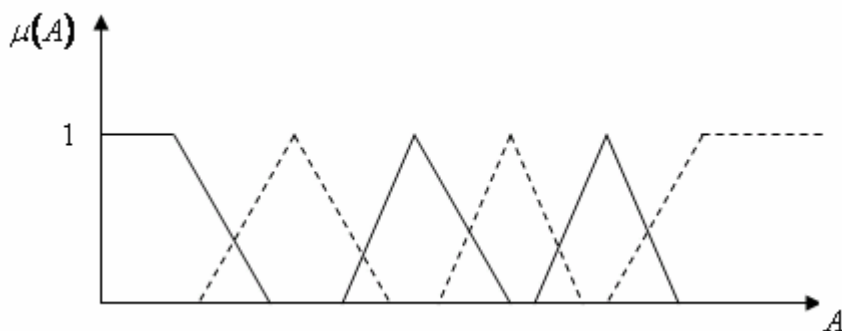


Figure 5.28a : Univers de discours avec des termes linguistique invalides

Pour pallier à cet inconvénient on procède à une technique de récupération des fonctions d'appartenance invalides selon le principe de recouvrement suivant:

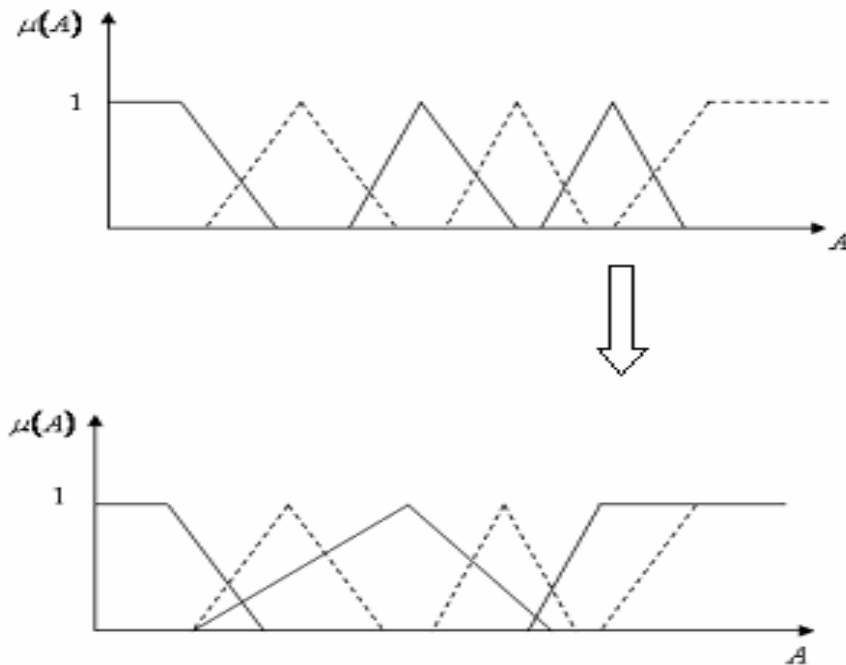


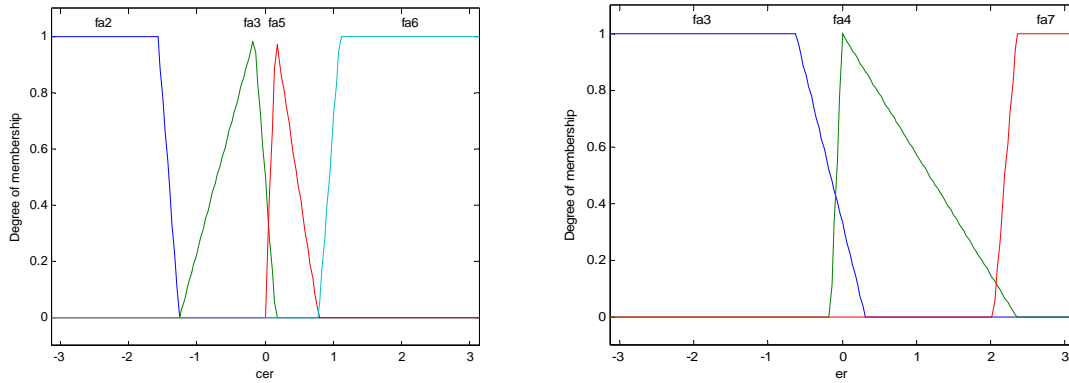
Figure 5.28b : Procédure de recouvrement des fonctions d'appartenance

5.6.3. Caractéristique de l'A.G

- Taille de la population : 30 individus
- longueur du chromosome : 103 bits
- Nombre de gènes codants les gènes de contrôle : 21 bits
- Nombre de gènes codants la base des règles : 49 bits
- Nombre de gènes codants la base de données : 33 bits
- Probabilité de croisement : 0.85
- Probabilité de mutation : 0.02
- Type de codage : base 7

5.6.3. Résultats de simulation :

-Les fonctions d'appartenance des différents termes linguistiques qualifiant les entrées



Les fonctions d'appartenance des différents termes linguistiques qualifiant la sortie

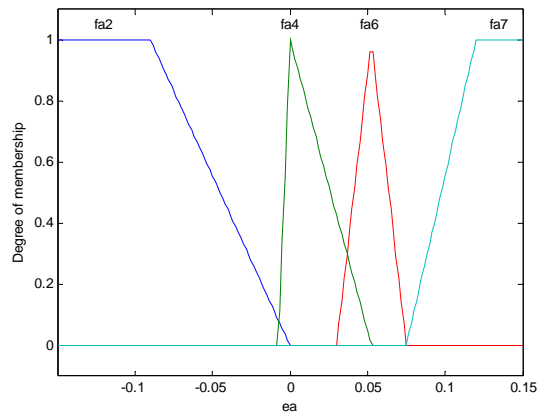


Figure 5.29 : Fonctions d'appartenance du contrôleur optimisé par les HGA

Base de règles définies par le processus d'optimisation.

1. If (er is fa4) and (cer is fa2) then (ea is fa7)
2. If (er is fa3) and (cer is fa3) then (ea is fa2)
3. If (er is fa4) and (cer is fa3) then (ea is fa7)
4. If (er is fa7) and (cer is fa3) then (ea is fa4)
5. If (er is fa3) and (cer is fa5) then (ea is fa6)
6. If (er is fa4) and (cer is fa5) then (ea is fa4)
9. If (er is fa7) and (cer is fa5) then (ea is fa4)
7. If (er is fa4) and (cer is fa6) then (ea is fa4)
8. If (er is fa7) and (cer is fa6) then (ea is fa4)

-Courbe représentant la variation du cap du dirigeable par la régulation dans le plan image et le suivi de la trajectoire, obtenues par un contrôleur flou optimisé par les HGA.

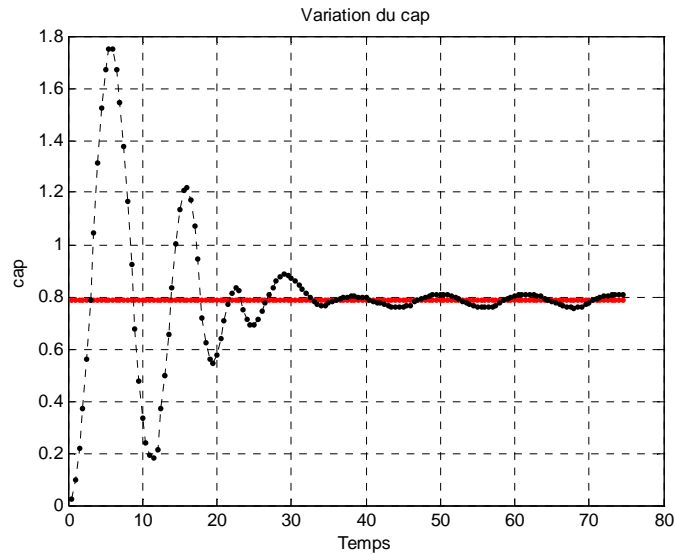


Figure 5.30 : Variation du cap obtenue par les HGA

Navigation:

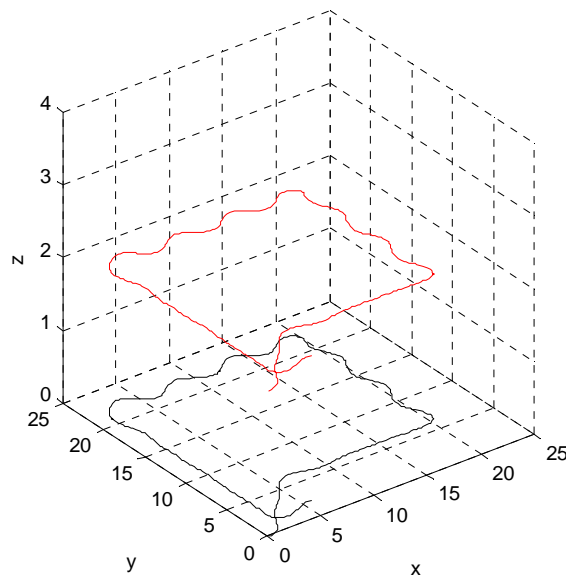


Figure 5.31 : Navigation du CF optimisé par les HGA

Le résultat montre que l'élimination des termes linguistique et la réduction des règles floues ont simplifié la structure du contrôleur flou sans incidence majeure sur les performances

Le nombre des variables linguistiques des fonctions d'appartenance d'entrée et de sortie est optimisé ainsi que la base des règles ; le nombre est de 8 au lieu de 21

La structure du contrôleur aura des répercussions sur le temps de traitement et l'espace mémoire ce qui présentera un avantage du point de vue implémentation.

Les séquences images montrant la commande par la vision.

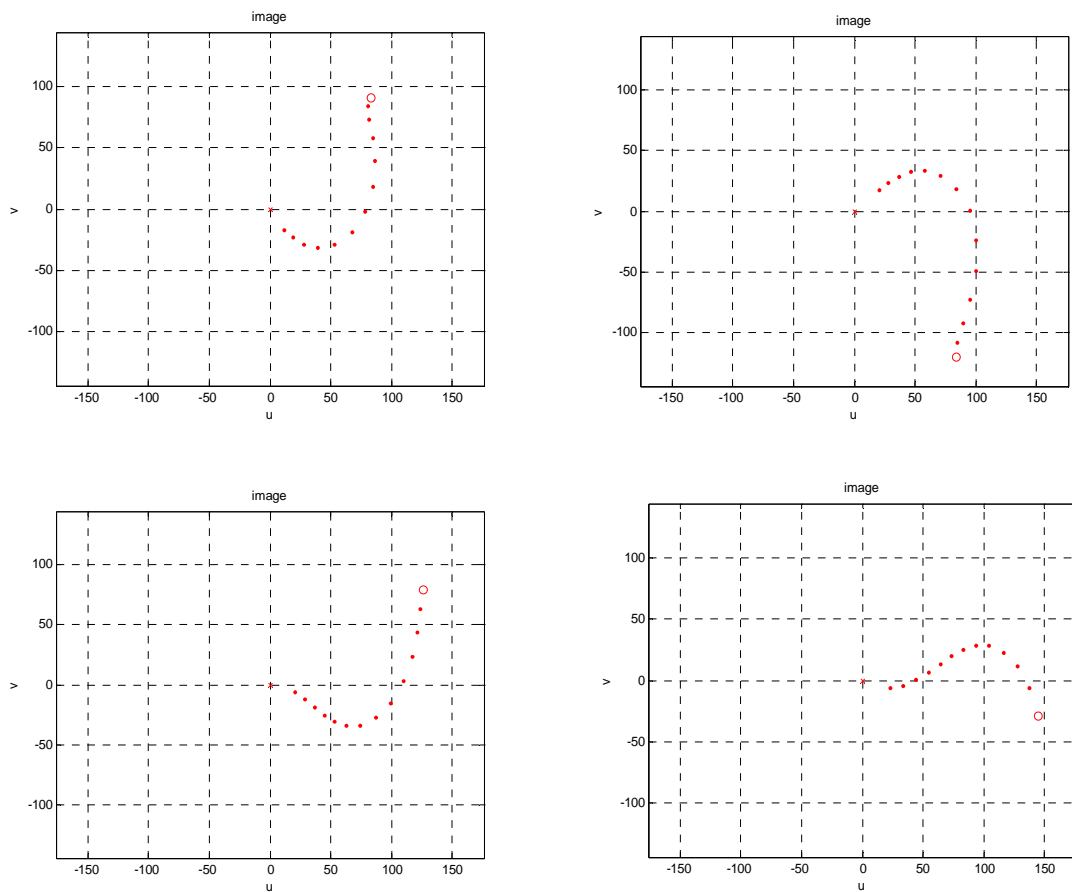


Figure 5.32a : Séquences d'images obtenues

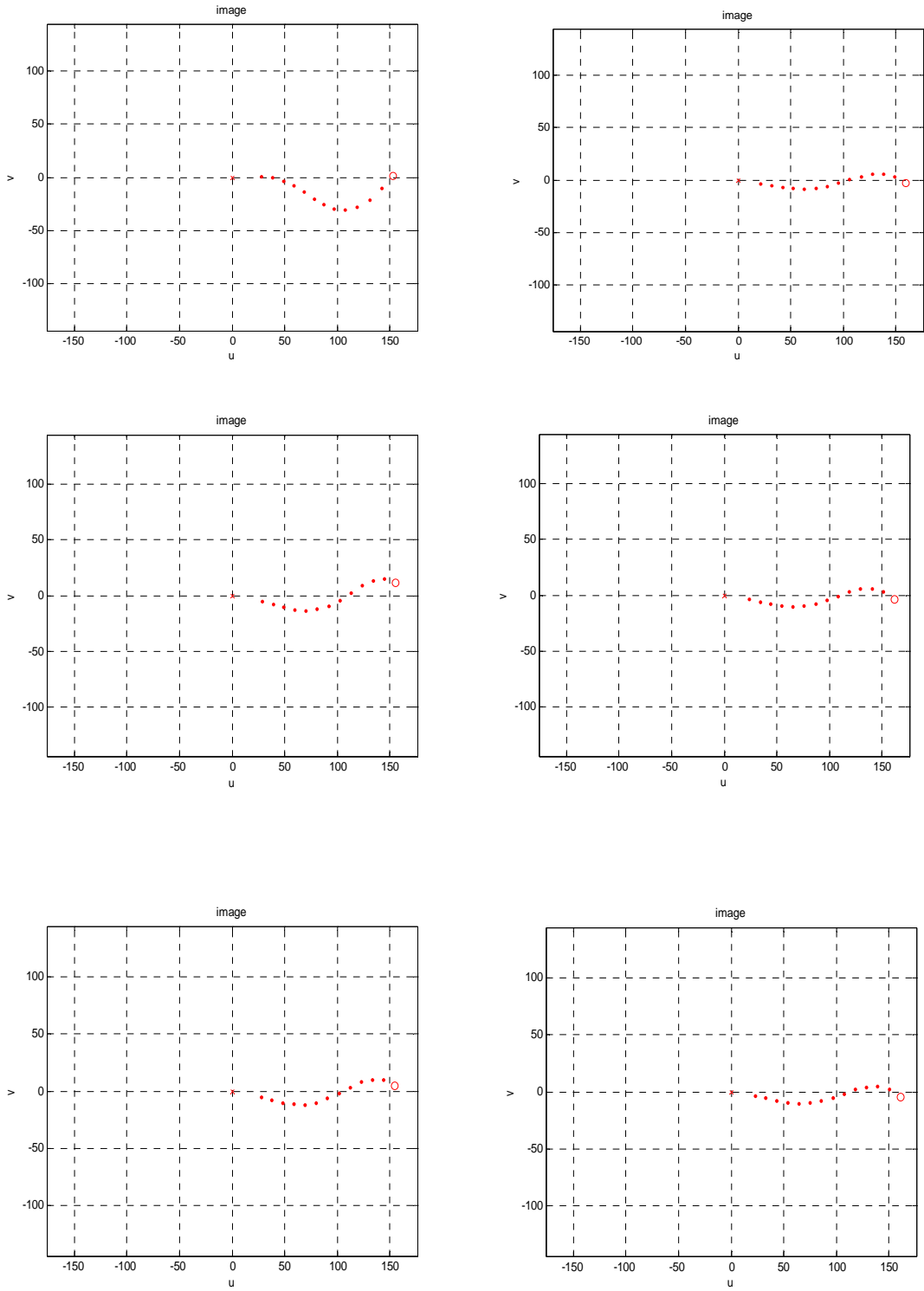


Figure 5.32b : Suite des séquences d'images obtenues

-Superposition des images

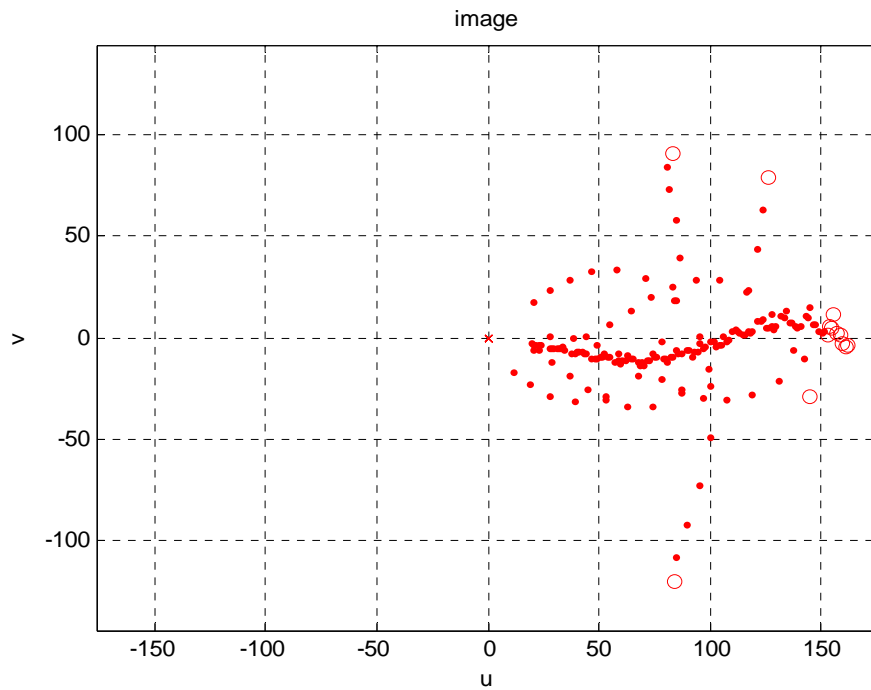


Figure 5.33 : Superposition des images

La commande par la vision est mise en évidence par cette série de séquences d'images et leur superposition sur le même plan. On constate bien que notre contrôleur tente de positionner le point (petit cercle rouge) ; représentant le repère observé par la caméra ; le long de l'axe v du plan d'image et il essaye de le centrer au milieu de l'image.

L'alignement le long de l'axe v signifie que la commande par la vision essaye d'asservir le cap vers l'angle désiré tandis que le centrage du point signifie que la commande par la vision fait avancer le ballon vers le but à atteindre.

5.7. Synthèse d'un contrôleur neuro-flou par l'algorithme génétique

Le Contrôleur neuro-flou est une structure qui comprend les éléments de base d'un système flou et d'un réseau de neurones à multicouche, chaque couche est caractérisée par un ensemble de neurones simulant un sous systèmes du contrôleur flou : la fuzzification, les règles floues et la défuzzification

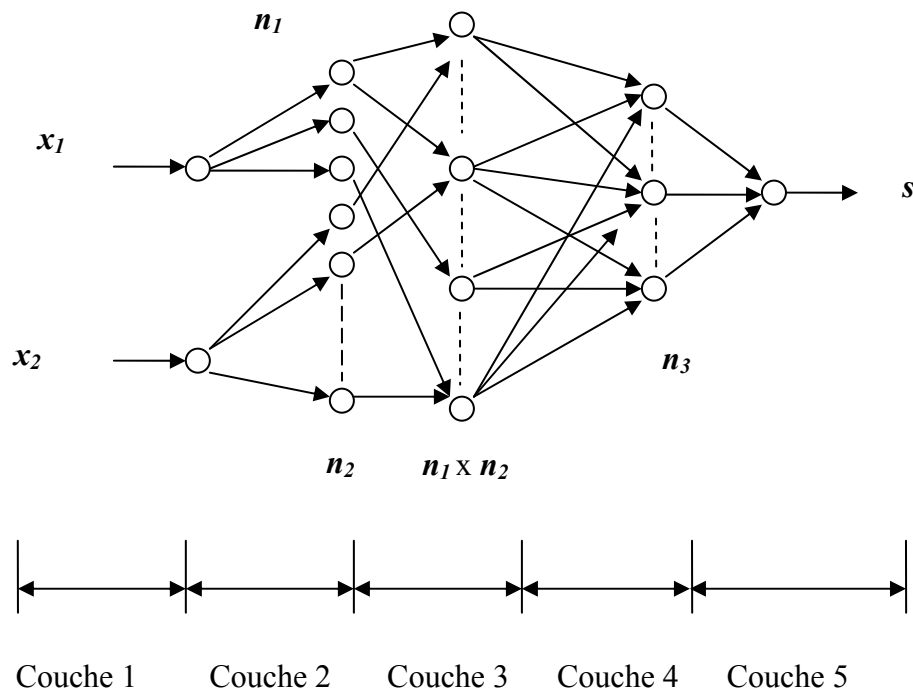


Figure5.34 : Structure d'un réseau neuro-flou à cinq couches

5.7.1. Description du réseau neuro-flou adopté

$$x_1 = e; \quad x_2 = \nabla e \quad n_1 = 3; \quad n_2 = 5; \quad n_3 = 5$$

Notation :

x_i^L : la valeur d'entrée du i ème neurone de la couche L.

y_i^L la valeur de sortie du i ème neurone de la couche L .

m_i^L, σ_i^L le centre et l'écart type de la fonction gaussienne du neurone i de la couche L

w_{ij} : poids de connexion de sortie de j ème neurone de la couche 3 vers le i ème neurone de la couche 4

Couche 1 :

L'entrée de cette couche est transmise directement vers la couche cachée suivante

Ses nœuds s'appellent les neurones de distribution

$$\begin{aligned} y_1^1 &= x_1^1 \\ y_2^1 &= x_2^1 \end{aligned} \quad (5.17)$$

Couche 2 :

Les neurones de cette couche fuzzifient les entrées par l'application des fonctions d'appartenance de forme gaussienne ; il y a autant de neurones qu'il y a des sous ensembles flous dans le système d'inférence ; chaque neurone calcule le degré de vérité d'une fonction d'appartenance par sa fonction de transfert .

$$\begin{aligned} x_i^2 &= y_1^1 \quad \text{pour } i=1,2,\dots,n_1 \\ &= y_2^1 \quad \text{pour } i=n_1+1,\dots,n_1+n_2 \end{aligned} \quad (5.18)$$

$$y_i^2 = \exp\left(-\frac{x_i^2 - m_i^2}{\sigma_i^2}\right) \quad \text{pour } i=1,2,\dots, n_1+n_2 \quad (5.19)$$

Couche 3 :

Les neurones de cette couche représentent chacun la prémisse d'une règle, ils reçoivent en entrée le degré de vérité des différents sous ensembles flous composant cette prémisse.

Cette couche réalise la conjonction des prémisses des règles floues.

Chaque nœud a deux entrées issues de la couche précédente.

$$\begin{aligned} y_i^3 &= \min(y_j^2, y_k^2) & i=1,2,\dots,n_1 \times n_2 \\ & & j=1,2,\dots,n_1 \\ & & k= n_1+1,\dots,n_1+n_2 \end{aligned} \quad (5.20)$$

-Les poids des connexions sont égaux à 1

Couche 4 :

Les poids de connexions de cette couche expriment les conséquences de la j ème règle avec la i ème variable linguistique de sortie ; les poids prennent des valeurs entre 0 et 1 ou tout simplement 0 ou 1

Les entrées sont regroupées par l'opérateur OU ($x_i^4 = \max(y_i^3)$) pour que chaque règle aura une conséquence appropriée

la sortie sera de la forme

$$y_i^4 = \sum_{j=1}^{n1 \times n2} w_{ij} x_j^4 \quad \text{pour } j=1,2,\dots,n3 \quad (5.21)$$

Couche 5 :

C'est la couche de défuzzification qui simule le centre de gravité

$$y_1^5 = \frac{\sum_{j=1}^{n3} m_j \sigma_j^4 y_j^4}{\sum_{j=1}^{n3} \sigma_j^4 y_j^4} \quad (5.22)$$

Où les m_j^4 et σ_j^4 représentent les paramètres des fonctions d'appartenance gaussiennes des variables linguistiques de la sortie du contrôleur

5.7.2 Caractéristiques du contrôleur adopté :

- deux entrées et une seule sortie
- la première entrée comprend trois variables linguistiques (N, EZ, P) dont les fonctions d'appartenance ont la forme gaussiennes
- cinq variables linguistiques (NG, NM, EZ, PM, PG) dont les fonctions d'appartenance de forme gaussiennes affectées à la deuxième entrée et à la sortie
- codage de chaque fonction d'appartenance

C(m _i)	C(σ _i)
--------------------	--------------------

-Type de codage : base 5

-Forme du chromosome

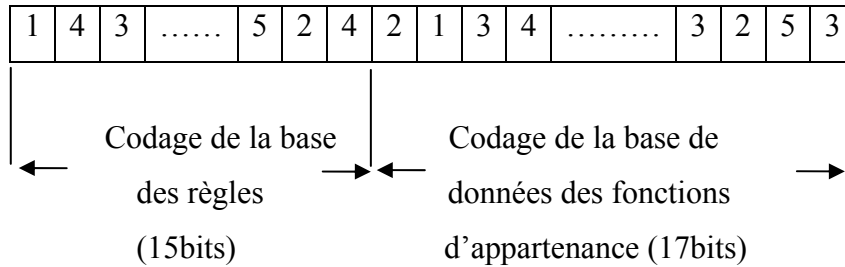


Figure 5.35 : Forme du chromosome du contrôleur neuro-flou

5.7.3. Caractéristique de l'A.G

-Taille de la population : 30 individus

-Nombre de gènes codants la base des règles 32 bits

-Nombre de gènes codants la base de données : 15 bits

-Probabilité de croisement : 0.85

-Probabilité de mutation : 0.02

-Type de codage : base 5

5.7.4. Résultats de simulation

a- Les fonctions d'appartenance des différents termes linguistiques qualifiant les entrées et la sortie :

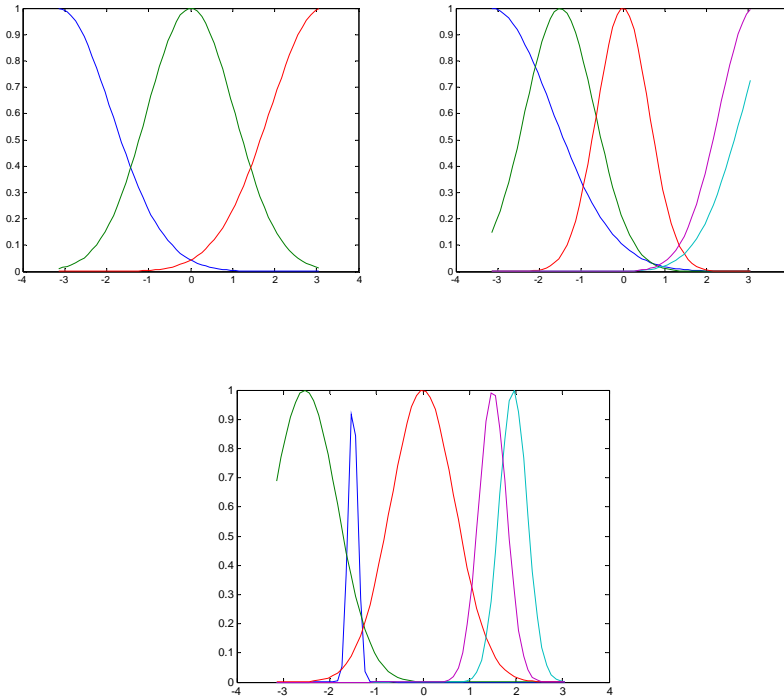


Figure 5.36 : Les fonctions d'appartenance des entrées et de sortie du réseau neuro- flou

b-La réponse de la commande du cap du ballon dirigeable et le suivi de la trajectoire.

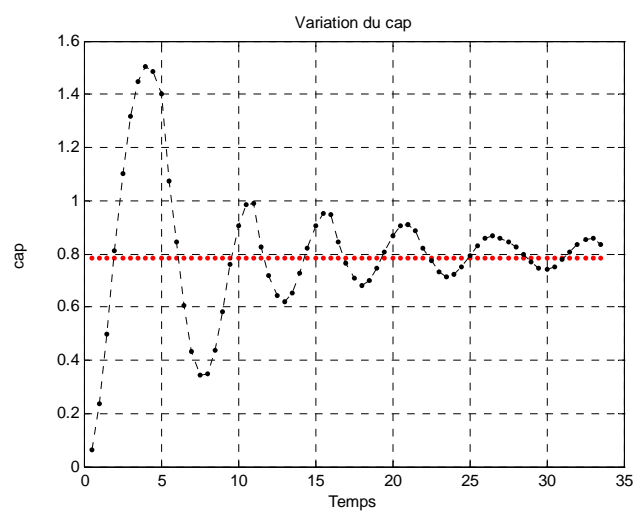


Figure 5.37 : Variation du cap en fonction du temps obtenue par réseau neuro- flou

c-Navigation :

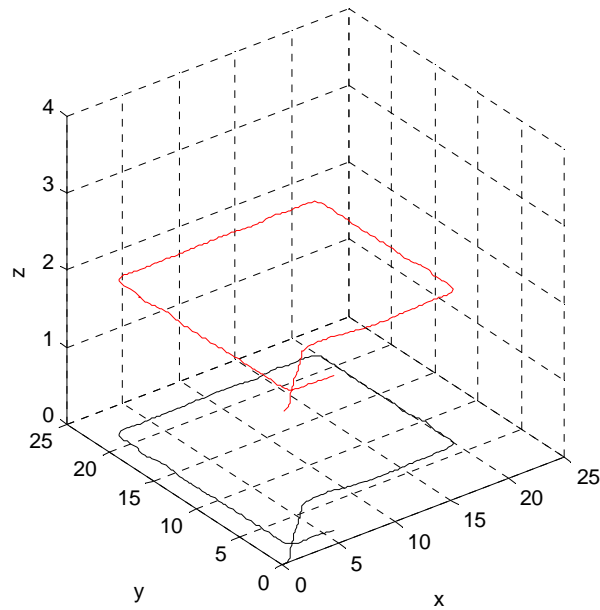


Figure 5.38 : Navigation obtenue par réseau neuro-flou

Interprétation :

Les résultats sont similaires au contrôleur flou mais avec des performances supérieures en temps de réponse qui est égale à 15s environ, mais pas d'amélioration du taux d'oscillation.

Cette application a montré la capacité d'adaptation des réseaux de neurones avec le système d'inférence flou et l'utilisation des Algorithmes génétique comme outil d'apprentissage.

CONCLUSION

Le travail que nous avons présenté dans ce mémoire traite la navigation du ballon dirigeable lorsque les mesures sont fournies par un capteur de vision; il apporte une contribution aux applications des techniques d'asservissement visuel des robots volants (UAV).

L'étude des commandes des dirigeables dans la littérature nous a permis de bien définir les techniques de contrôle de l'interaction entre le robot volant et son environnement physique, par un schéma de commande en boucle fermée sur les informations issues des capteurs de vision. Cette formulation ne nécessite pas une connaissance explicite des paramètres intrinsèques et extrinsèques de la caméra.

Toutefois l'intégration vision-robotique a posé plusieurs problèmes tant au niveau conceptuel qu'au niveau réalisation. On peut citer parmi eux, la phase de modélisation du système dirigeable plus caméra embarquée.

Dans ce contexte nous avons étudié et développé deux techniques de commande ; la commande flou basé sur le modèle de Mamdani et la commande hybride flou -réseau de neurones que l'on appelle neuro-flou. Leurs synthèses sont réalisées par les algorithmes génétiques qui optimisent simultanément les fonctions d'appartenance et les règles floues dans le premier, tandis que dans le deuxième contrôleur, ils améliorent les poids de connexion du réseau de neurones pour son apprentissage.

Pour les besoins de la simulation, le premier chapitre traite la modélisation cinématique et dynamique du dirigeable ; on peut constater dans l'étude de la littérature relative au ballon dirigeable l'utilisation des techniques de vol des avions et des drones et les techniques des sous marin au milieu d'un fluide; ce qui explique l'utilisation du formalisme de Fossen dans les références de la littérature traitant le domaine d'étude de la dynamique des dirigeables .Grâce à ce formalisme la notion de masses et inerties ajoutée à été introduite où la masse et l'inertie du fluide induit en mouvement sont ajoutés à la masse et à l'inertie du corps du solide , en ignorant l'existence du fluide.

En second chapitre on s'intéresse à l'asservissement visuel où des conclusions ont été déduites ; la reconstruction de la localisation du ballon dirigeable par des mesures fournies par la caméra nécessitent une phase d'interprétation géométrique de la scène représentant l'environnement d'évolution du dirigeable, alors que l'intégration des

informations visuelles dans la boucle de commande permet de s'affranchir de l'étape de reconstruction et d'estimation de la position grâce à la matrice Jacobienne qui exprime l'interaction entre l'environnement et le mouvement de la caméra embarquée.

Les notions de base de la commande flou et neuro- flou ont été reprises dans le chapitre 3 alors que le chapitre 4 est consacré à la structure de l'algorithme génétique inspiré de la génétique et le processus d'évolution naturelle et il est utilisé comme outil d'optimisation.

Dans le dernier chapitre, nous avons, tout d'abord présenté la stratégie de navigation de vol du ballon dirigeable en utilisant la vision où des points représentant les primitives visuelles ont été introduites dans la boucle de commande, ensuite la synthèse du contrôleur flou à été abordée où les paramètres du contrôleur ont été déduite par plusieurs simulation ; des résultats acceptables ont été obtenues.

En deuxième partie la synthèse d'un contrôleur flou par les algorithmes génétiques classiques est présentée ; le principe de codage est décrit par une chaîne de bit représentant le chromosome regroupant toutes les informations du contrôleur flou et le mode de codage à base n est utilisé. Ensuite nous avons présenté les opérateurs génétiques permettant à l'algorithme génétique d'explorer l'espace de solution et de converger vers l'optimum. L'utilisation des algorithmes génétiques a donné des performances supérieures à celle régulateur précédent où la navigation par la vision est testé par une simulation du suivi d'une trajectoire fermée de forme rectangulaire ;on a un bon contrôle de position avec de petites oscillations du cap. Ces résultats montrent les performances de notre contrôleur étudié comparativement à la simulation réalisée par les auteurs de l'article[2] où le dirigeable suit une trajectoire sous forme spirale cubique ave une commande basée sur la position ; le ballon avance et tourne sur lui-même [APPENDICE B].

Les résultats de simulation démontrent l'influence des fonctions d'appartenance et les règles floues sur le bon fonctionnement du contrôleur. Cependant l'implémentation des contrôleurs flous nécessite un temps de traitement et un espace de mémoire qui dépendent du nombre de règles et le nombre de fonctions d'appartenance.

A cet effet une méthode d'optimisation par les algorithmes génétiques hiérarchique a été abordée en troisième partie pour la synthèse d'un contrôleur flou avec l'optimum du nombre de fonction d'appartenance et de règles floues. L'application des ces algorithmes à donnés des résultats satisfaisants.

En dernière partie de ce chapitre la synthèse d'un contrôleur neuro-flou est présentée où les étapes d'un système d'inférence flou ont été introduites dans les

différentes couches d'un réseau de neurones artificiels. L'apprentissage supervisé utilise les algorithmes génétiques. Le résultat de simulation a donné des performances supérieures à celle du régulateur obtenu auparavant.

On ne peut pas considérer que le contrôleur hybride comme méthode de contrôle concurrent de la commande classique mais ces approches constituent un outil supplémentaire pour développer des méthodes de contrôle robuste. Ces commandes peuvent être utilisées dans les systèmes à multi contrôleur.

A l'issue de ces travaux, cette thèse peut ouvrir de nouvelles perspectives de recherche, parmi lesquelles on cite :

- l'introduction des approches d'amélioration de la convergence des algorithmes génétiques tel que recuit simulé, mise à l'échelle.
- l'asservissement visuel par d'autres primitives tel que les droites les segments les cercles.
- introduction à la commande par la vision en utilisant la commande optimale, adaptative et robuste
- comparaison des méthodes d'apprentissage du réseau de neurones du contrôleur neuro-flou.
- la commande du vol du ballon dirigeable par l'introduction des perturbations.

APPENDICE A
LISTE DES SYMBOLES ET DES ABREVIATIONS

$\boldsymbol{\eta}$:	Vecteur de configuration
$\boldsymbol{\eta}_1$:	Vecteur des positions
$\boldsymbol{\eta}_2$:	Vecteur d'orientations
\boldsymbol{v} :	Vecteur vitesses
\boldsymbol{v}_1 :	Vecteur de vitesses linéaires
\boldsymbol{v}_2 :	Vecteur de vitesses angulaires
ϕ :	Angle de roulis
θ :	Angle de tangage
ψ :	Angle de lacet
$\boldsymbol{\omega}$:	Vecteur de vitesses angulaires
ω_x :	Vitesse angulaire autour de l'axe x
ω_y :	Vitesse angulaire autour de l'axe y
ω_z :	Vitesse angulaire autour de l'axe z
\mathbf{R} :	Matrice de rotation
\mathbf{J}_1 :	Matrice de transformation des vitesses linéaires
\mathbf{J}_2 :	Matrice de transformation des vitesses angulaires
\mathbf{J} :	Matrice de transformation
\boldsymbol{r}_G :	Distance du centre de gravité .
\boldsymbol{v}_o :	Vecteur vitesses linéaires du centre du repère
\boldsymbol{v}_G :	Vecteur vitesses linéaires du centre de gravité
m :	Masse
\boldsymbol{a}_o :	Vecteur accélérations du centre du repère
\boldsymbol{a}_G :	Vecteur accélérations du centre de gravité
H :	Moment cinétique
\boldsymbol{f}_0 :	Forces externes
\boldsymbol{M}_o :	Moments externes
\boldsymbol{I}_o :	Matrice d'inertie
\boldsymbol{M}_{RB} :	Matrice d'inertie du système
\boldsymbol{C}_{RB} :	Matrice des forces de Coriolis et des forces centrifuges
\boldsymbol{u} :	Vitesse locale du fluide
\boldsymbol{n} :	Vecteur normal à la surface du dirigeable orienté vers le fluide
ϕ :	Potentiel des vitesses
ρ_0 :	La densité du fluide
\boldsymbol{M}_a :	Matrice de masses ajoutées.
T_f :	Energie cinétique du fluide
T_a :	Energie cinétique des efforts de la masse ajoutée
F_a :	Force
n_a :	Couple

- Γ_a : Couple induit par l'inertie et la masse ajoutées.
 C_a : Matrice de force de Coriolis et de forces de centrifuges hydrodynamique
 D : Matrice des forces et les couples appliquées au corps par l'environnement
 D_f : Force d'amortissement due à la forme du dirigeable
 τ_D : Force d'amortissement
 C_D : Coefficient de traînée
 A : Surface du corps en incidence
 V_f : Volume du fluide déplacé
 F_B : Poussée d'Archimède
 F_G : Force de pesanteur
 W : Force de pesanteur
 B : Poussée d'Archimède
 M : Matrice 6x6 contenant les termes de masse et d'inertie
 C : Matrice 6x6 des termes de forces de Coriolis et de centrifuges
 D : Matrice contenant des termes de forces d'amortissements
 g : Matrice contenant les termes des forces hydrostatiques
 τ : Forces et couples des propulseurs
 P^* : Vecteur d'attitude désirée
 \hat{P} : Vecteur d'attitude estimée
 U : Commande du robot
 T : Torseur cinématique de la caméra
 f : Informations visuelles
 L : Matrice d'interaction
 f^* : Informations visuelles désirées
 α : Angle d'orientation de la camera autour de l'axe x
 β : Angle d'orientation de la camera autour de l'axe y
 γ : Angle d'orientation de la camera autour de l'axe z
 ${}^b_c T$: Matrice de passage du repère ballon au repère camera
 ${}^b_c R$: Matrice de rotation du repère ballon au repère camera
 ${}^b_c t$: Vecteur de translation du repère ballon au repère camera
 G : Matrice homogène du changement de repère.
 ${}^c R_w$: Matrice de rotation
 ${}^c t_w$: Vecteur de translation
 k_u, k_v : les facteurs d'échelles horizontal et vertical des pixels
 u_0, v_0 : les coordonnées du point principal défini comme étant l'intersection entre l'axe optique et le plan image
 θ' : Angle entre les deux axes du repère image
 f_d : Distance focale
 K : Matrice des paramètres intrinsèques de la camera
 $\alpha_u, \alpha_v, \alpha_{uv}$: Paramètres intrinsèques de la camera
 L_S^T : Matrice d'interaction
 μ_A : Fonction caractéristique
NG : Négatif Grand

NM :	Négatif Moyen
NP :	Négatif Petit
EZ :	Environ de Zéro
PP :	Positif Petit
PM :	Positif Moyen
PG :	Positif Grand
w_i :	Poids synaptiques
$e(r_i)$:	la partie entière
r_i :	Rapport du fitness sur la moyenne des fitness
F_1 :	Force du propulseur latéral
F_2 :	Force du propulseur latéral
F_3 :	Force du propulseur de déviation
ψ :	Angle du cap
UAV:	Unmanned Arial Vehicule
PID :	Proportional Integral Derivative
AG :	Algorithme Génétique
HGA :	Algorithme Génétique Hiérarchique

APPENDICE B

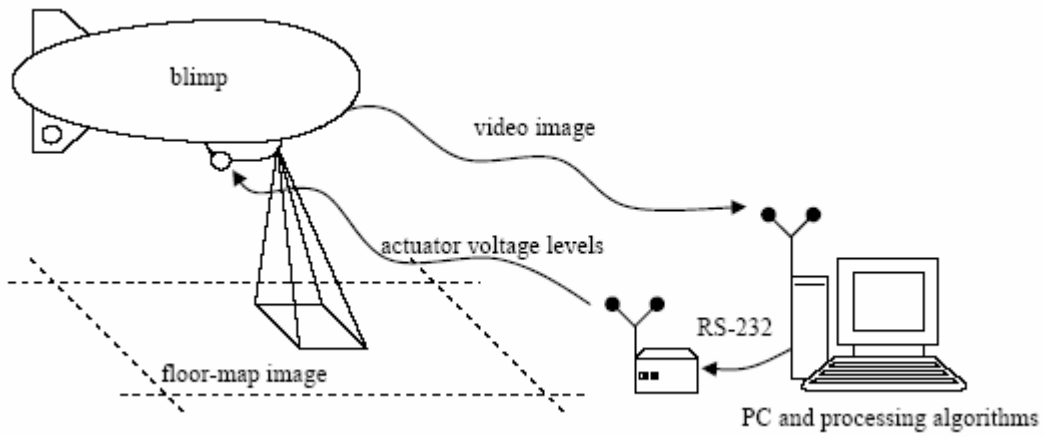
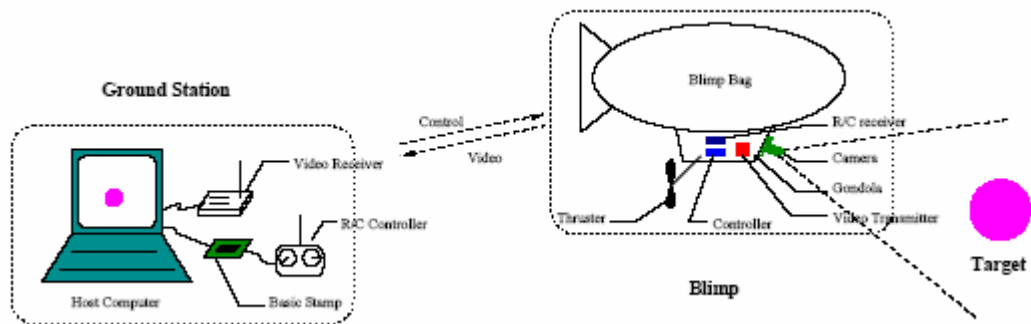
A- Les différents ballons dirigeables utilisés en laboratoires Les ballons d'intérieur



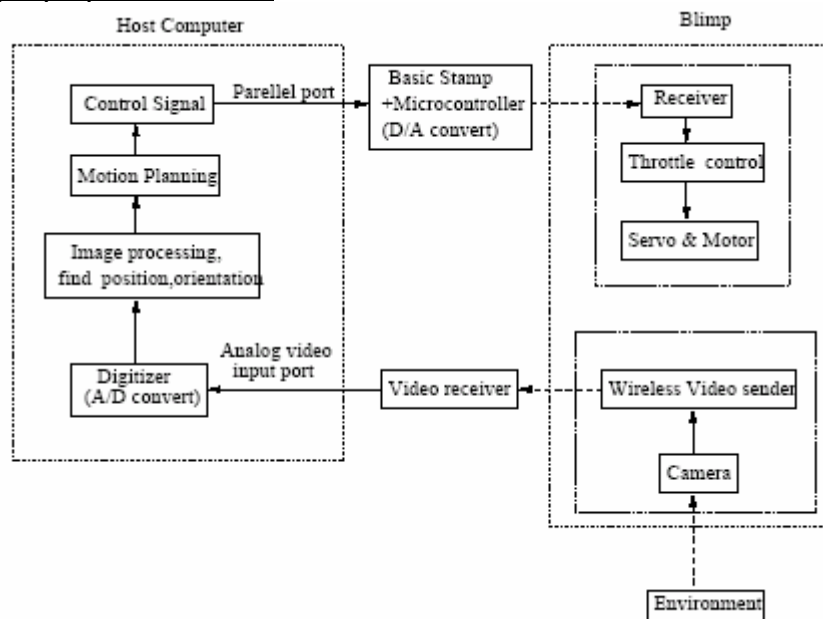
Les ballons d'extérieur

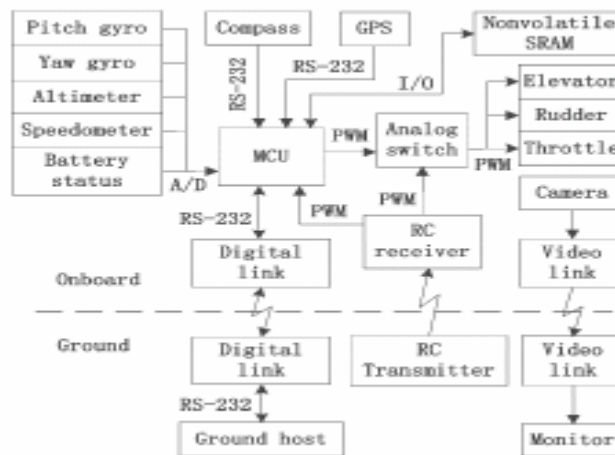
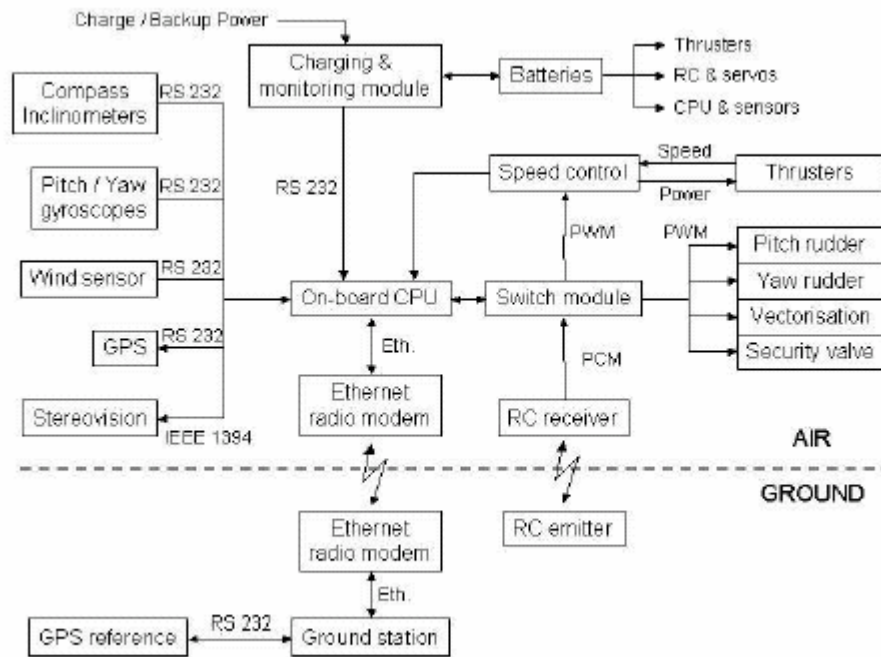


B- Postes de commande

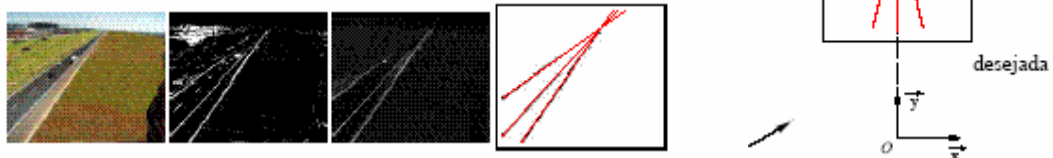


C-Le synoptique de contrôle

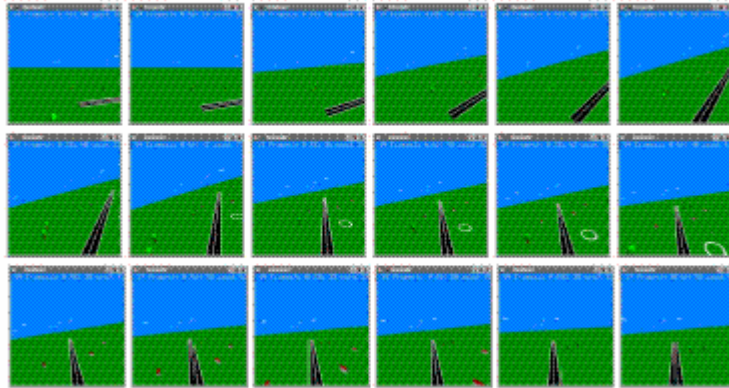




D- Le principe d'asservissement par les primitives de droite



E- Séquences d'images montrant l'asservissement le long d'une route []



F- résultat de simulation de l'article [2]

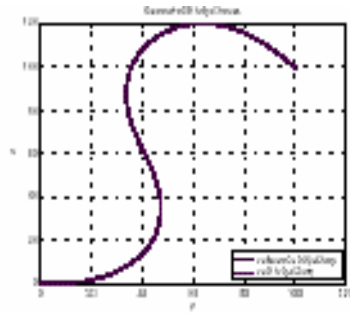
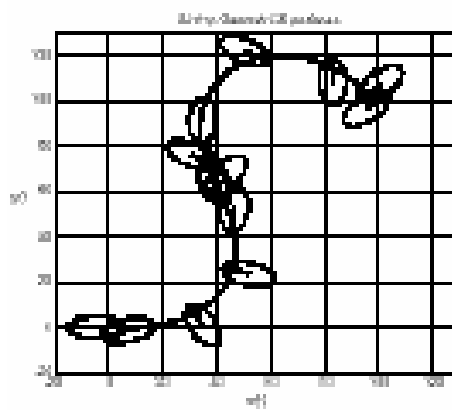


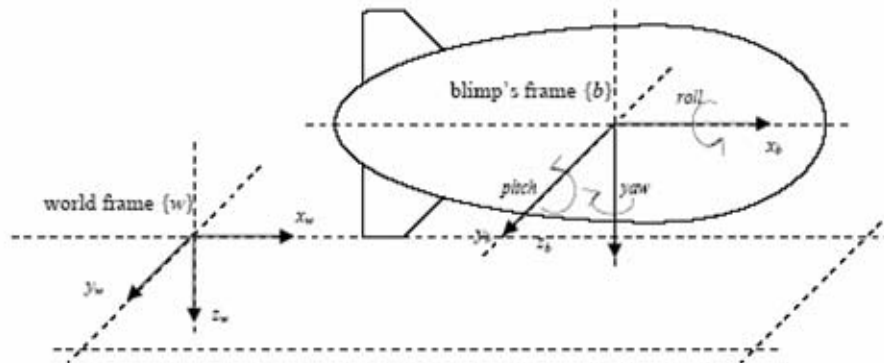
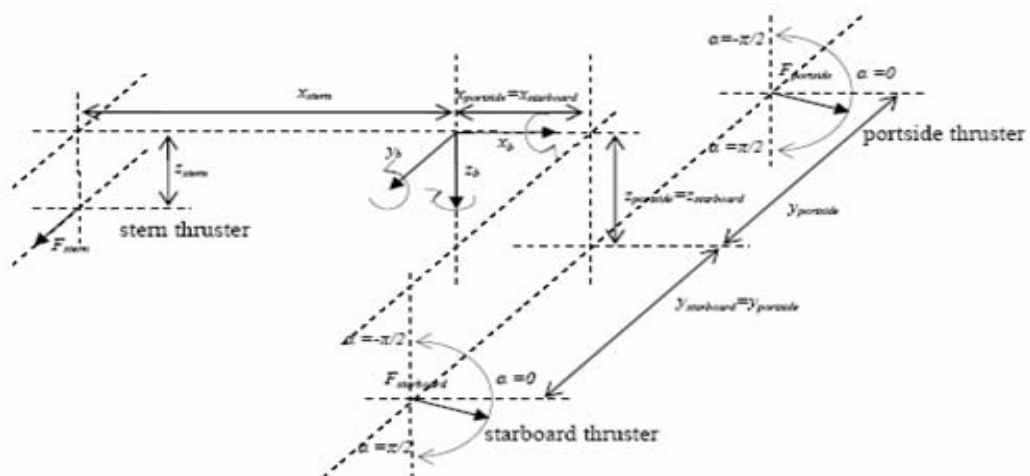
Fig 1



APPENDICE C

A- Paramètres du ballon utilisé dans notre simulation

- a- Placement des références
 b-

b- Position des propulseurs

x_{stern}	-0.77
y_{stern}	0
z_{stern}	0.37
x_{star}	0.275
y_{star}	0.21
z_{star}	0.59
x_{port}	0.275
y_{port}	-0.21
z_{port}	0.59

B- Paramètre dynamique du dirigeable

a_{11}	0.29846 kg
a_{22}	0.70180 kg
a_{33}	0.70180 kg
a_{44}	0.00000 kg
a_{55}	0.04767 kg
a_{66}	0.04767 kg
I_{xx}	0.1710 kg
I_{yy}	0.3341 kg
I_{zz}	0.3341 kg

component	weight [g]
gondola with motors and batteries attached	329.09g±0.01g
camera and support rig	35.33g±0.01g
Rechargeable 9v battery (Phillips)	38.46g±0.01g
non-rechargeable 9v battery (Duracell)	46.81g±0.01g
tail fin	31.63g±0.01g
tail fin with stern engine	53.08g±0.01g
helium envelope (empty)	523g±1g
total	1.073kg

D_{vx}	0.1904	D_{vxx}	0.1763
D_{vy}	0.78	D_{vyy}	0.8011
D_{vz}	0.78	D_{vzz}	0.8011
D_{vx}	0.1	D_{vzx}	0.01
D_{vy}	0.3	D_{vzy}	0.1
D_{vz}	0.09	D_{vzz}	0.0954

C- Paramètres de la caméra

192×144 pixels	values and uncertainties
focal length [f_x, f_y]	[181.29822 180.66439]±[1.39125 1.24790]
principal point [c_x, c_y]	[85.05022 57.01526]±[1.72825 2.08181]
skew	0
lens distortion	[-0.45963 0.20020 0.00615 -0.00104]±[0.02482 0.09336 0.00245 0.00209]

REFERENCE

1. Yasmina BESTAOUI, Salim HIMA «Some insights in path planning of small autonomous blimps» Laboratoire des Systèmes Complexes, CEMIF, Université d'Evry Val d'Essonne, France.2002
2. Y. Bestaoui, S. Hima «Trajectory tracking of a dirigible in high constant altitude» IFAC Nonlinear control systems (NOLCOS), Saint Petersburg, Russia, July 2001
- 3.. Salim HIMA, Yasmina BESTAOUI «Time optimal paths for lateral navigation of autonomous underactuated airship»Laboratoire des Systèmes Complexes» CNRS FRE 2492 Université d'Evry Val d'Essonne, 38 rue du pelvoux, 91020 Evry, France.2002
4. Salim HIMA, Yasmina BESTAOUI « Motion generation on trim trajectories for an autonomous underactuated airship » Laboratoire des Systèmes Complexes, CNRS FRE 2492 Université d'Evry Val d'Essonne, 38 rue du pelvoux, 91020 Evry, France.2001
5. E. C. de Paiva, S. S. Bueno, S. B. Gomes, J. J. Ramos, M. Bergerman «A control system development environment for AURORA's semi-autonomous robotic airship». IEEE,International Conference on Robotics and Automation, Detroit, MI, USA, May 1999
6. S. B. V. Gomes and J. J. G. Ramos«Airship dynamic modeling for autonomous operation» in Proc. of the IEEE International Conf. on Robotics and Automation, Leuven, Belgium, May 1998.
7. Elfes, A, Siqueira Bueno, S, Bergerman, M.,Guimaraes Ramos, J. « A semi – autonomous robotic airship for environmental monitoring mission» IEEE international Conference on Robotics and Automation, Detroit,MI, pp. 3449 – 3455.1999.
8. Emmanuel Hygouneq, Il-Kyun Jung, Philippe Soueres, Simon Lacroix «The Autonomous Blimp Project of LAAS-CNRS:Achievements in Flight Control and Terrain mapping». LAAS/CNRS.France;2001.
9. Jongwoo Kim James P. Ostrowski «Motion Planning of Aerial Robot using Rapidly-exploring Random Trees with Dynamic Constraints General Robotics, Automation, Sensing, and Perception » (GRASP) Laboratory University of Pennsylvania, 3401 Walnut Street, Philadelphia, PA 19104-6228. 2000
10. H. Zhang and J. P. Ostrowski, «Visual servoing with dynamics Control of an unmanned blimp»in Proc. of the IEEE International Conf. on Robotics and Automation; 1999
- 11.B. Espiau, F. Chaumette, and P. Rives, «A new approach to visual servoing in robotics» IEEE Trans. on Rob. And Autom., vol. 8, no. 3, pp. 313-326, 1992.

12. François Chaumette « la relation vision-commande :théorie et application à des s robotiques » Thèse de doctorat de l'université de Rennes1 France . 1990.
- 13.S. Hutchinson, G. D. Hager, and P. I. Corke«A tutorial on visual servo control». IEEE Trans. on Rob. and Autom., vol. 12, no. 5, pp. 651,October 1996.
14. Paul lee ,Tim Dean, Andrew Yap«On-board vision using visual-servoing for robocup F-180 League mobile robots» Department of computer science and software engineering the University of Melbourne Australia. 2002
- 15.G. F. Silveira et al. «Optimal visual servoed guidance of outdoor autonomous robotic airships» in Proc. of the American Control Conf., USA, 2002.
16. Silveira, G. F. et al. «Line following visual servoing for aerial robots combined with complementary sensors» In Proc. IEEE International Conference on Advanced Robots, Coimbra, Portugal.2003
17. Silveira, G. F. et al.. « A fast vision-based road following strategy applied to the control of aerial robots» In IEEE XIV Braz. Symp. on Comp. Graphics and Image Processing, pages 226–231, Brasil.2001
- 18- Jinjun Rao, Zhenbang Gong, Jun Luo and Shaorong Xie« Unmanned Airships for Emergency Management» Proceedings of the 2005 IEEE International Workshop on Safety, Security and Rescue Robotics Kobe, Japan, June 2005.
19. Marion G.Perhinschi «Parameter optimiszation via genetic algorithm of fuzzy controller for autonomous air vehicle flight Dynamics and control» laboratory National Aerospace Research Institue Buchaerest,Romania .AIAA99-4084. 1999
20. Masahito Yamamoto and Azuma Ohuchi«Development of autonomous blimp robot with intelligent control»Graduate School of Engineering, Hokkaido University.IWEC2003.
21. George Kantora David Wettergreena, James P. Ostrowskib, Sanjiv Singha«collection of environmental data From an airship platform» a The Robotics Institute, Carnegie Mellon University; Department of Mechanical Engineering and Applied Mechanics, The Univeristy of Pennsylvania.
- 22.Takanori Fukao,Kazushi Fujitani,Takeo Kanade «An Autonomous Blimp for Surveillance system». Proceedings of the 2003 IEEE/RSJ Int.Conference on Intelligent Robots and systems Las Vegas,Nevada. Octobre 2003.
23. CANG YE and DANWEI WANG«A Novel Navigation Method for Autonomous Mobile Vehicles» Journal of Intelligent and Robotic Systems 32: 361–388, 2001.
24. Wei WU «Synthèse d'un contrôleur flou par Algorithme Génétique :Application au réglage dynamique des paramètres d'un système»thèse doctorat de l'Université de Lille 1998

25. Jonwoo.Kim Jim Keller Vijay Kumar. «Design and verification of controllers for airships» Proceedings of the 2003 IEEE/RJS.Int.Conference on Intelligent Robots and systems.
26. J. R. Azinheira et al., «Visual servo control for the hovering of an outdoor robotic airship» in Proc. IEEE InternationalConf. on Robotics and Autom., Washington, USA, ,pp. 2787-2792. 2002
27. Filipe Manuel da Silva Metelo, Luís Ricardo Garcia Campos« Vision based control an autonomous blimp (vidéoblimp) ».Thèse ; Université Technique de Lisbonne. 2003
28. Jean- François Lots «Application of visual servoing to the dynamique positioning of underwater vehicle»thesis for the degree of de doctor Heriot-watt university. 2001
29. Hong Zhang« motion control for Dynamique mobile robots». Thesis of doctorate; the Faculties of the University of Pennsylvania USA. 2002.
30. Tomas SALGADO JIMENEZ«Contribution a la commande d'un robot sous-marin autonome de type torpille».Thèse de doctorat , Université de Montpellier II France .2004
31. Sjoerd van der Zwaan «Vision based station keeping and docking for floating robots» Master Université Technique de Lisbonne .2001
32. Fossen, Thor I., «Guidance and Control Of Ocean Vehicles, John Wiley & Sons, 1994;
33. Jean-Michel Renders , «Algorithmes génétiques et réseaux de neurones applications à la commande de processus» Edition Hermes Paris 1995.
- 34.Louis Gacogne . « Eléments de logique floue»Edition Hermes ,1997.
- 35 .David e.Goldberg, «Algorithmes Génétiques,exploration,optimisation et apprentissage automatique» Edition Addisson-Wesly.france 1994
36. M.Hadj Sadok« Méthodologie de contrôle basée sur la logique flou» Thèse de magister Université de Blida Algérie 1995
37. Amrouche Badia«Commande adaptative à base de réseaux de neurones artificiels pour bras manipulateur» Thèse de magister Université de Blida Algérie 2003
38. Benhellal Belkheir«Application de deux commandes robustes : la commande à structure variable et la commande floue optimisée par un algorithme génétique à un robot manipulateur» Thèse de magister Université de Blida Algérie 2001
- 39.A.kerfa ,T. Dahmane « Conception d'un contrôleur flou optimisé par algorithme génétique pour l'articulation d'un robot» mémoire PFE Institut de mécanique , Université de Blida Algérie 2002.
- 40.T.loutani ,H .Bencherif «optimisation rendement d'un contrôleur flou par un HGA application à l'articulation d'un robot »Mémoire PFE Institut d'électronique, Université de Blida Algérie 2003.

41. S.Tahar ,A.Samai «Conception d'un filtre numérique par algorithme génétique hiérarchique». Mémoire PFE, Institut d'électronique, Université de Blida .Algérie 2004.
42. Technique de l'ingénieur A1661« Mécanique générale, cinématique générale »par Jean-Pierre BROSSARD ;Professeur de Mécanique à l'Institut National des Sciences Appliquées (INSA) de Lyon.
43. Vratislav Jerabek, Gerard Lachiver « Algorithmes micro-génétiques dans l'optimisation des contrôleurs neuro-flous»Université de Sherbrooke:, Faculte des sciences appliquées. Département de génie électrique et de génie informatiqueSherbrooke (Quebec) CANADA . IEEE 1995
44. Ashok Kumar Goel Suresh Chandra Saxena Surekha Bhanot« A Genetic Based Neuro-Fuzzy Controller for Thermal Processes »JCS&T April 2005
45. Manish Kumar Devendra P. «Intelligent learning of fuzzy logic controllers via neural network and génetic algorithm»Proceedings of 2004 JUSFA2004 Japan – USA Symposium on Flexible AutomationDenver, Colorado, July 2004
- 46 Catherine Fargeon et Jean-Philippe Quin«Robot mobile » Edition Teknea Toulouse 1993
- 47-C.L. Karr et E.J. Gentry. « Fuzzy control of pH using genetic algorithms». IEEE Trans. on Fuzzy Systems, 1(1) :46–53, 2 1993.
- 48-M.A. Lee et H. Takagi. « Dynamic control of genetic algorithms using fuzzy logic Techniques»”. Dans Proceedings of the 5th Int'l Conf. on Genetic Algorithms,1993
- 49-K. Chwee Ng et Y. Li. « Design of sophisticated fuzzy logic controllers using genetic algorithms». Dans Proc. of ICCI94/Fuzzy Systems. 1994.
- 50-P. Thrift. Fuzzy logic synthesis with genetic algorithms. Dans Proc. of the Fourth Int. Conf. Genetic Algorithms, pages 509–513, 1991.
- 51-Naomi Ehrich Leonard «Stability of bottom-heavy underwater vehicule» Department of Mechanical and Aerospace Engineering Princeton University,Princeton,NJ08544 1996
- 52-Arnold M Kuethe Chuen-Yen Chow«Foundations of Aerodynamic Deign»fifth edition Edition ,John Willey & Sonc,Inc 1998
- 53-E.H. Mamdani et S. Assilian. « An experiment in linguistic synthesis with a fuzzy logic controller». Int. J. Man Mach. Studies 1975.