

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET
POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université de Batna 2
Faculté de Technologie
Département d'Électronique



Mémoire

Présenté pour l'obtention du diplôme de
MAGISTER en Électronique

OPTION

Contrôle Industriel

Par

Salah Eddine Ghamri

Thème:

**Commande Neuronale d'un Robot Mobile Non-
Holonome avec glissement des roues**

Soutenu devant le jury composé de :

Pr. Saidi Lamir	Prof.	Université de Batna	Président
Pr. Foudil Abdessemed	Prof.	Université de Batna	Rapporteur
Pr. Benmahammed Khier	Prof.	Université de Sétif	Examineur
Dr. Slimane Noureddine	M.C.A.	Université de Batna	Examineur
Dr. Melaab Djamel	M.C.A.	Université de Batna	Examineur
Dr. Ameddah Djamel	M.C.A.	Université de Batna	Invité

For my Father, and those who fought, and still fighting against tyranny and aggression.

REMERCIEMENTS:

Je tiens tout d'abord à remercier **Pr. Foudil Abdessemed** professeur à l'université de Batna pour avoir accepté de m'encadrer et diriger mon travail, je le remercie pour son aide et ses conseils, j'ai été honoré de travailler avec lui.

Je remercie également :

Pr. Saidi Lamir ; Professeur à l'université de Batna pour l'honneur qu'il me fait en acceptant de présider ce travail. **Pr. Benmahammed Khier** ; Professeur à l'université de Sétif, **Dr. Slimane Noureddine** ; MCA à l'université de batna et **Dr. Melaab Djamel** ; MCA à l'université de Batna pour avoir accepté d'examiner et de juger ce modeste travail.

En fin, je remercie tous ceux qui ont contribué à la réalisation de ce travail (Ma mère – mes sœurs – mes frères).

TABLE DES MATIERES

Remerciements :	2
Introduction :	8
Etat de l'art:	11
Chapitre I : La modélisation mathématique d'un robot mobile	12
I-1. Le Pioneer 3DX :	13
I-2. La conduite différentielle :	14
I-3. La holonomie et non-holonomie :	14
I-3-1. Les robots mobiles holonomes :	14
I-3-2. Les robots mobiles non-holonomes :	15
I-4. Modélisation géométrique :	15
I-5. Modélisation cinématique :	17
I-5-1. Les contraintes cinématiques d'un robot à conduite différentielle :	17
I-5-2. La cinématique du robot et du glissement :	20
I-6. Modélisation dynamique :	23
I-6-1. Modèle dynamique de Newton-Euler :	24
I-7. Modèle générale du robot :	27
Chapitre II : Interaction roue-chaussée	31
II-1. Le taux de glissement et l'angle de dérive :	32
II-1-1. Le taux de glissement :	32
II-1-2. l'angle de dérive :	33
II-2. Les forces de tractions :	33
II-3. Modèle d'interaction roue-chaussée	34
II-3-1. La formule magique :	35
Chapitre III : La commande	41
III-1. Introduction :	42
III-1-1. Les taches de mouvement de base :	42
III-1-1-1. Suivi de chemin :	42
III-1-1-2. Poursuite de trajectoire :	43

III-1-1-3. Stabilisation de configuration fixe :	44
III-2. Commande de suivi non-linaire par backstepping:	45
III-2-1. Contrôleur de poursuite cinématique non-linaire :	47
III-2-2. Contrôleur de poursuit dynamique:	49
III-3. Commande adaptative neuronale :	51
III-3-1. les réseaux de neurones artificiels :	51
Chapitre IV : simulation	66
IV-1. Suivi de trajectoire par backstepping :	67
IV-1-1. trajectoire Ligne droite :	67
IV-1-2. Trajectoire cercle :	74
IV-1-3. virage a 90° :	80
IV-2. Suivi de trajectoire par commande neuronale :	85
IV-2-1. trajectoire Ligne droite :	85
IV-2-2. Trajectoire cercle :	89
IV-2-3. virage a 90° :	95
Conclusion :	100
Bibliographie :	102

Table des figures

Figure I.1 Le Pioneer 3DX.....	13
Figure I.2 Structure et éléments du Pioneer 3DX	14
Figure I.3 La conduite différentielle	14
Figure I.4. A) Une roue à galets tangentiels (holonomique) B) Uranus le robot holonome	15
Figure I.5 Repérage du robot.....	16
Figure I.6 Roulement sans glissement.....	18
Figure I.7 Relation entre les coordonnées du point P et les point de contacts des roues.	18
Figure I.8 Représentation des vitesses de glissement.....	21
Figure I.9 Localisation du point G	21
Figure I.10 Diagramme du corps libre de la plateforme du robot sans les roues	24
Figure I.11 Diagramme du corps libre des roues	24
Figure II.1 Contact des aspérités avec charge et sans charge.....	34
Figure II.2 Quelques paramètres indiquer sur la courbe de force longitudinale	36
Figure II.3 Quelques exemples de courbes des forces de traction longitudinales.....	38
Figure II.4 Quelques exemples de courbes des forces de traction latérales.	39
Figure III.1 Suivi de Chemin.....	42
Figure III.2 Poursuite de Trajectoire	43
Figure III.3 Stabilisation autour d'une posture	44
Figure III.4 Bloc-diagramme du robot	45
Figure III.5 Bloc-diagramme de la commande	45
Figure III.6 Bloc-diagramme de la commande backstepping	50
Figure III.7 Le neurone biologique	52
Figure III.8 Diagramme opérationnel d'un neurone biologique	53
Figure III.9 Le neurone artificiel.....	54
Figure III.10 Fonction sigmoïde logistique.....	55
Figure III.11 Fonction hyperbolique tangentielle.....	55
Figure III.12 Structure d'un réseau de neurones MLP.....	56
Figure III.13 Commande neuronale par apprentissage supervisé	60
Figure III.14 Apprentissage du modèle inverse	60
Figure III.15 Commande neuronale spécialisée par un modèle inverse.....	61
Figure III.16 le neuro-contrôle indirect.....	61
Figure III.17 Bloc-diagramme de la commande neuronal étudiée.	64
Figure IV.1 Suivi de la trajectoire ligne (roulement pur).....	68
Figure IV.2 Les erreurs du suivi (ligne-roulement pur)	68
Figure IV.3 Les vitesses du robot (ligne-roulement pur)	68
Figure IV.4 Les couples des roues droite et gauche (ligne-roulement pur)	69
Figure IV.5 Suivi de la trajectoire ligne cas de surface (1).....	69
Figure IV.6 Les erreurs de suivi pour la surface (1)	70
Figure IV.7 Les vitesses du robot surface (1)	70

Figure IV.8 Les couples moteurs droite et gauche surface (1).....	70
Figure IV.9 Les taux de glissement et les angles de dérive gauche et droite surface (1).....	71
Figure IV.10 Les forces de traction longitudinales et latérales droite et gauche surface (1)	71
Figure IV.11 Suivi de trajectoire ligne cas de surface (2).....	71
Figure IV.12 Les erreurs du suivi surface (2)	72
Figure IV.13 Les vitesses du robot surface (2)	72
Figure IV.14 Les couples moteurs des roues droite et gauche surface (2).....	72
Figure IV.15 Les taux de glissement et les angles de dérive droites et gauches surface (2).....	73
Figure IV.16 Les forces de traction longitudinales et latérale droite et gauche surface(2).....	73
Figure IV.17 Suivi d'une trajectoire cercle sans glissement	74
Figure IV.18 Les erreurs du suivi sans glissement.....	74
Figure IV.19 Les vitesses linéaire et angulaire du robot sans glissement	75
Figure IV.20 Les couples moteurs des roues droite et gauche sans glissement	75
Figure IV.21 Suivi d'une trajectoire cercle surface (1).....	76
Figure IV.22 Les erreurs du suivi surface (1)	76
Figure IV.23 Les vitesses linéaire et angulaire du robot surface (1).....	76
Figure IV.24 Les couples moteurs des roues droite et gauche surface (1).....	77
Figure IV.25 Les taux de glissement et les angles de dérive droite et gauche surface (1).....	77
Figure IV.26 Les forces de traction longitudinales et latérales droite et gauche surface (1)	77
Figure IV.27 Suivi d'une trajectoire cercle surface (2).....	78
Figure IV.28 Les erreurs du suivi surface (2)	78
Figure IV.29 Les vitesses linéaire et angulaire du robot surface (2).....	78
Figure IV.30 Les couple moteurs des roues droites et gauche surface (2).....	79
Figure IV.31 Les taux de glissement et les angles de dérive droite et gauche surface (2).....	79
Figure IV.32 Les forces de traction longitudinales et latérale gauche et droite surface (2)	79
Figure IV.33 Suivi d'une trajectoire L (sans glissement)	80
Figure IV.34 Les erreurs du suivi (sans glissement)	80
Figure IV.35 Les vitesses linéaire et angulaire du robot (sans glissement)	81
Figure IV.36 Les couples moteurs droite et gauche (sans glissement)	81
Figure IV.37 Suivi d'une trajectoire L (surface (1))	82
Figure IV.38 Les erreurs du suivi (surface (1)).....	82
Figure IV.39 Les vitesses linéaire et angulaire du robot (surface (1))	82
Figure IV.40 Les couples moteurs droite et gauche (surface (1))	83
Figure IV.41 Suivi d'une trajectoire L (surface (2))	83
Figure IV.42 Les erreurs du suivi (surface (2)).....	84
Figure IV.43 Les vitesses linéaire et angulaire du robot (surface (2))	84
Figure IV.44 Suivi d'une trajectoire ligne droite (surface (1)).....	85
Figure IV.45 Les erreurs du suivi (surface (1)).....	85
Figure IV.46 Les vitesses linéaire et angulaire du robot (surface (1))	86
Figure IV.47 Suivi d'une trajectoire ligne droite (surface (2))	86
Figure IV.48 Les erreurs du suivi (surface (2)).....	87
Figure IV.49 Les vitesses linéaire et angulaire du robot (surface (2))	87

Figure IV.50 Les couples moteurs droite et gauche du robot (surface (2)).....	87
Figure IV.51 Les taux de glissement et les angles de dérive droite et gauche (surface (2))	88
Figure IV.52 Les forces de traction longitudinale et latérales droite et gauche (surface (2))	88
Figure IV.53 La fonction F approximée (surface (2)).....	88
Figure IV.54 Quelques poids du réseau de neurones de la couche caché et de sortie (surface (2))..	88
Figure IV.55 Suivi d'une trajectoire cercle (surface (1)).....	89
Figure IV.56 Les erreurs du suivi (surface (1)).....	89
Figure IV.57 Les vitesses linéaire et angulaire du robot (surface (1))	90
Figure IV.58 Les couples des roues droite et gauche (surface (1))	90
Figure IV.59 La fonction F approximée ((surface 1)).....	90
Figure IV.60 Quelques poids du réseau de neurones (surface (1)).....	91
Figure IV.61 Les taux de glissement et les angles de dérive droite et gauche surface (1).....	91
Figure IV.62 Les forces de traction longitudinales et latérale surface (1)	91
Figure IV.63 Suivi d'une trajectoire cercle (surface (2), $k_x=8, k_y=5, k_\varphi=3$).....	92
Figure IV.64 Les erreurs du suivi (surface (2), $k_x=8, k_y=5, k_\varphi=3$).....	92
Figure IV.65 Les vitesses linéaire et angulaire du robot (surface (2))	93
Figure IV.66 Suivi d'une trajectoire cercle (surface (2), $k_x=1.5, k_y=1.5, k_\varphi=2$).....	93
Figure IV.67 Les erreurs du suivi (surface (2), $k_x=1.5, k_y=1.5, k_\varphi=2$).....	94
Figure IV.68 Les vitesses linéaire et angulaire du robot (surface (2), $k_x=1.5, k_y=1.5, k_\varphi=2$).	94
Figure IV.69 Comparaison de suivi d'une trajectoire cercle avec une posture de départ loin	94
Figure IV.70 Suivi d'une trajectoire L (surface (1)).....	95
Figure IV.71 Les erreurs du suivi (surface (1)).....	95
Figure IV.72 Les vitesses linéaire et angulaire du robot (surface 1))	96
Figure IV.73 Les taux de glissement et les angles de dérive droite et gauche (surface (1))	96
Figure IV.74 Les forces de traction longitudinales et latérales (surface (1))	96
Figure IV.75 Fonction F approximée (surface (1))	97
Figure IV.76 Quelques poids du réseau de neurones	97
Figure IV.77 Suivi d'une trajectoire L surface (2)	98
Figure IV.78 Les erreurs du suivi (surface (2)).....	98
Figure IV.79 Les vitesses linéaire et angulaire du robot (surface (2))	98
Figure IV.80 Les taux de glissement et les angles de dérive droite et gauche (surface (2))	99
Figure IV.81 Les forces de traction longitudinales et latérales droite et gauche (surface (2)).....	99

INTRODUCTION :

Ces dernières années, une grande attention a été consacrée au problème de la commande des systèmes non-holonomes, malgré que ces systèmes aient été étudiés dans la mécanique classique depuis plus de 150 ans. Beaucoup de systèmes mécaniques (tel que les robots mobiles à roues, les systèmes à remorque, les véhicules sous-marins, etc.) sont soumis à des contraintes de vitesse non-holonomes non-intégrables, i.e. les contraintes ne peuvent pas être écrites sous la forme des dérivées de temps de quelque fonction des coordonnées généralisées. Tel contraintes peuvent être généralement exprimées en termes des relations non intégrables linéaires de vitesse [1].

Les robots mobiles ont reçus une attention considérable ces dernières années en raison de leur utilité et la nature non-holonomique qui caractérise un certain nombre d'entre eux. Cette nature a rendu la stabilité de ces robots un défi de taille dans le domaine de contrôle.

La difficulté de commande des robots mobiles non-holonomes découle du théorème de Brockett [2] ; il est bien connu que ces systèmes à mobilité réduite ne peuvent pas être stabilisés à une configuration (position) désirée via un retour d'état différentiable ou encore continue. En conséquence la commande de cette classe des systèmes non linéaires ne se prête pas aux méthodes de la théorie de contrôle linéaire, et ne peut pas être transformée en un problème de contrôle linéaire d'une façon sérieuse [3].

La plupart des travaux de recherche sur la conception des commandes pour les robots mobiles non-holonomes a été portés sur le problème de la stabilisation cinématique. C'est-à-dire les robots sont présentés mathématiquement par leurs modèles cinématiques et les vitesses sont considérées comme des entrées de commande. Cependant, en pratique il est plus réaliste de formuler le problème de commande des systèmes non-holonomes au niveau de la dynamique, où les couples et les forces sont pris comme des entrées de commande.

En générale, tous les systèmes en réalité sont des systèmes non-linéaires, et la connaissance de leur dynamique est non disponible ou incertaine. C'est le cas pour les robots mobiles essentiellement à roues, où les incertitudes et les dynamiques inconnues dans des scénarios réels influent sévèrement sur leur comportement. Dans ce cas la stabilisation de tels systèmes nécessite l'utilisation des commandes basées sur l'approximation de ces non-linéarités dynamiques.

Trois taches peuvent être distinguées dans la commande des robots mobiles non-holonomes :

- La stabilisation de configurations fixes.
- La stabilisation de trajectoires.

- Le suivi de chemin.

L'objectif de « la stabilisation de configurations fixes » est de stabiliser asymptotiquement le robot dans une position d'équilibre donnée. Alors que le but de « poursuite de trajectoire » est d'avoir le robot suivre une courbe de référence avec une contrainte horaire dite trajectoire. Le facteur du temps est négligé pour « le suivi de chemin » mais la tâche reste la même que la poursuite de trajectoire.

Récemment l'intérêt dans la poursuite de trajectoire des robots mobiles a augmenté avec diverses contributions théoriques et pratiques. La stabilisation de poursuite peut être principalement divisée en 6 approches [4] :

- linéarisation par retour d'état.
- Commande par mode glissant.
- Commande par « Backstepping ».
- Commande par calcul des couples calculé.
- Commande adaptative.
- Commande intelligente.

En termes de consommation des ressources hardware et de calculs, seule la commande par « Backstepping » est considérée comme raisonnable pour la tâche de suivi de trajectoire. Mais l'inconvénient majeur de cette méthode est le besoin de la modélisation complète et la connaissance exacte de la dynamique du robot, ce qui ne peut pas être possible tout le temps. Une alternative pour surmonter cet inconvénient est d'utiliser les techniques d'intelligence computationnelle pour aider cette commande à stabiliser le suivi.

Passionné par leur capacité d'approximation et d'apprentissage, nous essayons dans ce mémoire d'examiner l'utilisation des réseaux de neurones dans la commande des robots mobiles à roues en présence des dynamiques inconnus. Il s'agit en l'occurrence du phénomène de glissement des roues.

Ce mémoire sera structuré en quatre chapitres :

Le premier chapitre concerne la modélisation mathématique du robot mobile à roues de configuration différentielle « Pioneer 3DX », le chapitre commence par une modélisation cinématique en présentant les contraintes non-holonomes et le glissement qui affecte les vitesses du robot. Ensuite Une étude explicite sur la dynamique du robot est présentée, en utilisant l'approche de Newton-Euler pour déterminer les composants de la dynamique, les forces et les couples qui agissent sur le robot.

Le second chapitre présente l'interaction Roue-chassée qui est très importante pour cette étude et la modélisation du glissement des roues. Ce chapitre commence par une définition

des taux de glissement et l'angle de dérive. Ensuite la modélisation de cette interaction suivant « La Formule Magique » de Hans B. Pacejka est présentée.

Le troisième chapitre est divisé en deux. En premier lieu la conception d'une commande backstepping capable de stabiliser le suivi de trajectoire est présentée. Ensuite la conception d'une commande neuronale est élaborée, en passant par une étude préliminaire sur les réseaux de neurones.

Le quatrième chapitre concerne le travail de simulation qui a été réalisée pour mettre en évidence le phénomène de glissement et le rendement des réseaux de neurones. Ce chapitre présente des nombreux tests de simulation sur le suivi de plusieurs trajectoires.

Ce travail se termine par une conclusion.

ETAT DE L'ART:

Depuis que les robots mobiles à roues sont de plus en plus utilisés pour des applications de haute vitesse et des environnements non-structurés, le problème de glissement des roues dans la robotique mobile est devenu de taille. Le glissement peut apparaître durant l'accélération ou la décélération du robot, même dans un virage à grande vitesse. Si le glissement est négligé dans la stratégie de commande, la stabilité du système peut être affectée, et le robot mobile peut ne pas accomplir ses tâches.

Il existe peu de travaux sur la modélisation de glissement des roues dans la littérature. Dans la communauté de la robotique mobile, [5] est l'un des premiers travaux qui a considéré le glissement dans la dynamique d'un robot mobile, les auteurs ont utilisé des forces de traction qui dépend linéairement sur des taux de glissements définis, ils ont utilisé une commande de retour de sortie en basant sur la méthode des variétés lentes. Dans [6] le frottement de la chaussée a été considéré comme des dynamiques inconnues, un facteur anti-glissement a été présenté qui indique le pourcentage des vitesses angulaires des roues par rapport aux vitesses réelles du robot et les forces motrices. Une technique adaptative basée sur les réseaux de neurones a été utilisée pour réaliser un suivi de vitesse optimal. Dans [7] les états de glissement sont introduits dans un modèle cinématique généralisé. Dans [8] le glissement a été considéré comme des petites perturbations limitées et mesurables, un contrôleur cinématique qui surmonte la perturbation a été développé. Dans [9] les efforts de traction longitudinale sont introduits dans un modèle de robot omnidirectionnel par la mesure extérieure des glissements. Par souci de simplicité un modèle qui ne prend pas en considération les glissements a été utilisé pour la conception de la commande. Dans [10] des efforts de tractions latérales qui dépendent linéairement de glissement latéral sont introduits, une commande de guidage (steering) a été appliquée. Dans [11] les forces de traction longitudinales et latérales sont approximées de façon qu'ils dépendent linéairement sur les glissements longitudinaux et latéraux, un modèle uni-cycle réduit d'un robot à quatre roues est utilisé. Pour la commande, la génération des couples motrices nécessaires dépend de glissements. Dans [12] un nouveau modèle d'un robot mobile non-holonome qui prend en considération les glissements et les forces de tractions longitudinales et latérales a été présenté. Le suivi de chemin a été adopté comme stratégie de commande. Dans [12] et [13] le suivi a été mené à l'aide d'une technique basée sur la linéarisation de feedback et un limiteur de vitesse dynamique. Dans [14] et [15] deux techniques sont utilisées pour asservir le robot, une basée sur le mode glissant et l'autre basée sur la linéarisation entrée-sortie. Le même modèle est utilisé dans ce travail.

CHAPITRE I : LA MODELISATION MATHEMATIQUE D'UN ROBOT MOBILE

I-1. Le Pioneer 3DX

I-2. La conduite différentielle

I-3. La holonomie et la non-holonomie

I-4. La modélisation géométrique

I-5. La modélisation cinématique

I-6. La modélisation dynamique

I-7. Le modèle générale du robot

Afin d'étudier le comportement d'un robot mobile, on doit en premier lieu décrire son mouvement par des équations algébriques, donc un modèle mathématique doit être établi.

Nous nous sommes intéressés dans ce travail aux robots mobiles à roues de type non-holonome à conduite différentielles, tout en tenant compte du phénomène de glissement des roues et des forces de traction.

Dans ce chapitre nous allons présenter d'abord le robot Pioneer 3DX, son type et sa configuration, puis nous allons construire son modèle mathématique en passant par les deux niveaux de modélisation à savoir : la modélisation cinématique et la modélisation dynamique.

I-1. LE PIONEER 3DX :

Le Pioneer 3DX (Fig. I.1) est un robot mobile destiné pour la recherche. Il est construit par **Adept Mobile Robots** [16]. Grâce à sa fiabilité et sa durabilité, le Pioneer est une plate-forme de référence pour la recherche en robotique, entièrement programmable et équipé de capteurs nécessaires pour une bonne navigation.



Figure I.1 Le Pioneer 3DX

Le robot mobile Pioneer 3DX est un robot non holonome à conduite différentielles (**Differential Drive WMR**). Il possède 2 roues motrices indépendantes non orientables sur le même axe et une roue décentrée orientable libre « roue folle » (Fig. I.2).

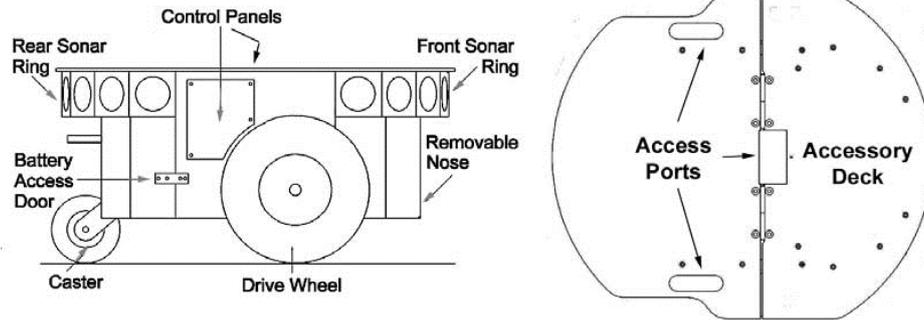


Figure I.2 Structure et éléments du Pioneer 3DX

I-2. LA CONDUITE DIFFÉRENTIELLE :

Cette conduite se compose de deux roues motrices fixes montées sur le côté gauche et à droite de la plate-forme du robot, les deux roues sont entraînées indépendamment. Un ou deux roues pivotantes passives sont utilisées pour l'équilibre et la stabilité. La conduite différentiel c'est l'entraînement mécanique le plus simple, car il n'a pas besoin de rotation d'un axe entraîné comme le montre la figure I.3. Si les roues tournent à la même vitesse, le robot se déplace en ligne droite vers l'avant ou vers l'arrière. Si une roue tourne plus vite que l'autre, le robot suit une trajectoire incurvée le long de l'arc d'un cercle instantanée. Si les deux roues sont en rotation à la même vitesse dans des directions opposées, le robot tourne autour du point médian de deux roues motrices.

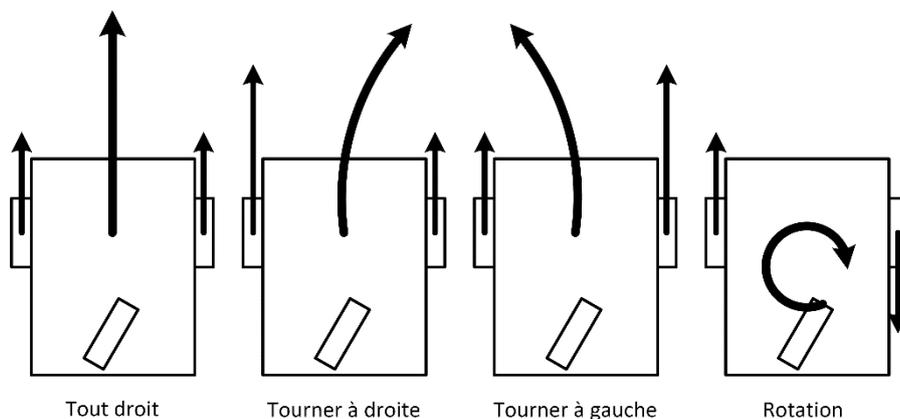


Figure I.3 La conduite différentielle

I-3. LA HOLONOMIE ET NON-HOLONOMIE :

Dans le monde des robots mobiles à roues on peut distinguer deux grandes catégories des robots : les robots holonomes et non-holonomes.

I-3-1. LES ROBOTS MOBILES HOLONOMES :

Un robot mobile à roues capable de se déplacer dans n'importe quelle direction quelque soit son orientation (sans manœuvre) est surnommé Holonome ou omnidirectionnel.

Grâce à des roues dites holonomes à galets tangentiels ou des roues mecanum, il possède trois degrés de liberté, Fig I.4 (a),(b) :

- Une translation selon X
- Une translation selon Y
- Une rotation selon Z



Figure I.4. A) Une roue à galets tangentiels (holonomique) B) Uranus le robot holonome

I-3-2. LES ROBOTS MOBILES NON-HOLONOMES :

Les robots mobiles non holonomes ont une structure et un mouvement très similaire aux véhicules de notre vie courante : voiture particulière, les bus, les camions...

En générale leur structure mécanique est simple. Ils ont des roues motrices, des roues directrices et des roues porteuses, même on peut trouver des roues avec les deux ou trois fonctions combinées.

Un robot non-holonome ne dispose que de deux degrés de libertés sur un plan :

- Une translation : avance ou recule.
- Une rotation : tourne vers la droite ou vers la gauche.

Il est difficile de faire revenir un robot non-holonome après son démarrage à sa position initiale sans manœuvres.

I-4. MODELISATION GEOMETRIQUE :

La modélisation géométrique permet de situer les différentes parties mobiles d'un robot les unes par rapport aux autres.

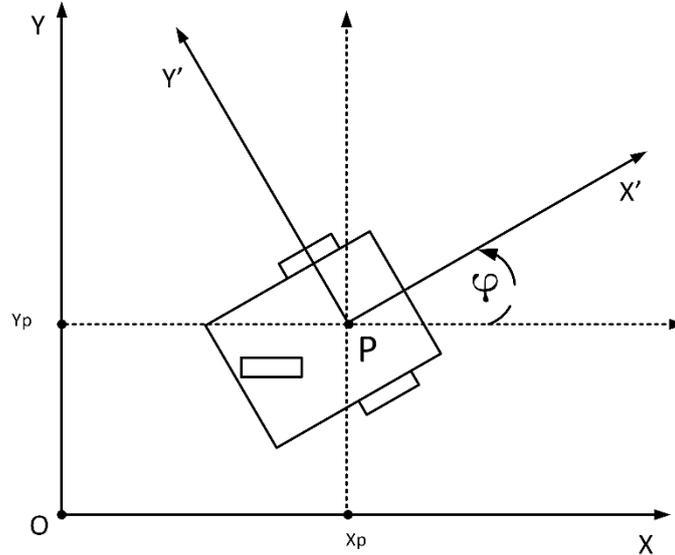


Figure I.5 Repérage du robot

Dans le cas des robots mobiles, tout l'ensemble du robot est considéré comme un seul corps solide qui se déplace sur un plan horizontal. Ce qui rend la géométrie plus simple et nous permet de trouver une relation entre le repère de référence globale, dit repère monde $F(O, \vec{x}, \vec{y}, \vec{z})$ et le repère de référence local du robot $F'(O', \vec{x}', \vec{y}', \vec{z}')$, comme le montre clairement la Figure I.5.

En cherchant à définir la position du robot dans le plan, un point P sur son châssis est choisi. C'est le point de référence de sa position et l'origine de son repère local $F'(P, \vec{x}', \vec{y}', \vec{z}')$.

La position du point P en repère monde est spécifiée à l'aide des coordonnées x, y et l'angle φ définit entre le repère globale et local (l'orientation). On peut décrire la pose du robot par un vecteur de ces trois éléments [18].

$$q = \begin{Bmatrix} x_p \\ y_p \\ \varphi \end{Bmatrix} \quad (1.1)$$

A ce point, c'est évident que la relation entre le repère monde et le repère du robot est une rotation suivant l'angle φ , elle est exprimée par la relation :

$$q = R(\varphi) * q' \quad (1.2)$$

Avec $R(\varphi)$ est la matrice de rotation orthogonale :

$$R(\varphi) = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.3)$$

Et q' est la pose du robot dans le repère local :

$$q' = \begin{pmatrix} x'_p \\ y'_p \\ \varphi' \end{pmatrix} \quad (1.4)$$

Cette transformation exprime aussi le déplacement dans les deux repères. \dot{q} et \dot{q}' décrivent les vitesses dans les deux repères :

$$\dot{q} = R(\varphi) * \dot{q}' \quad (1.5)$$

I-5. MODELISATION CINEMATIQUE :

La modélisation cinématique est l'étude du mouvement d'un system mécanique sans considération des forces qui influent sur son mouvement. À ce stade on s'intéresse qu'aux vecteurs de vitesse.

I-5-1. LES CONTRAINTES CINEMATQUES D'UN ROBOT A CONDUITE DIFFERENTIELLE :

Le mouvement d'un robot à conduite différentielle est caractérisé par deux contraintes non holonomes, qui sont obtenus par deux hypothèses principales [18] :

Hypothèse I : Aucun glissement latéral. Cela signifie simplement que le robot ne peut pas se déplacer latéralement dans le repère local, ce qui se traduit mathématiquement par l'équation :

$$\dot{y}'_p = 0 \quad (1.6)$$

Autrement la vitesse du point centre P est nulle le long de l'axe latéral du repère local. En utilisant la matrice de rotation $R(\theta)$, les vitesses dans le repère monde sont obtenues comme suit :

$$\begin{bmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} \dot{x}'_p \\ \dot{y}'_p \\ \dot{\varphi}' \end{bmatrix} \quad (1.7)$$

$$\dot{x}_p = \dot{x}'_p \cos(\varphi) - \dot{y}'_p \sin(\varphi) \quad (1.8a)$$

$$\dot{y}_p = \dot{x}'_p \sin(\varphi) + \dot{y}'_p \cos(\varphi) \quad (1.8b)$$

$$\dot{\varphi} = \dot{\varphi}' \quad (1.8c)$$

De Eq. (1.6) et Eq. (1.8) on trouve la première contrainte non-holonomique :

$$-\dot{x}_p \sin(\varphi) + \dot{y}_p \cos(\varphi) = 0 \quad (1.9)$$

Hypothèse II : le roulement est pur. Cette contrainte représente le fait que chaque roue du robot maintient un seul point de contact C avec le sol comme indiqué sur la figure Fig. I.6.

En outre il n'y a pas de glissement de la roue dans son axe longitudinal X' et aucun dérapage dans son axe orthogonale Y' .

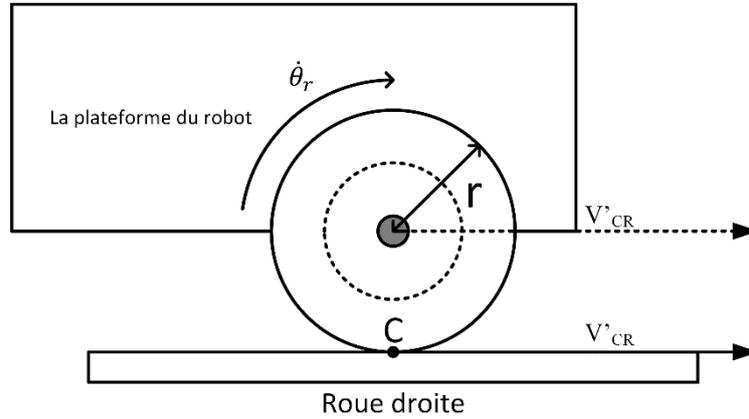


Figure I.6 Roulement sans glissement

Les vitesses des points de contact dans le repère robot local sont liées aux vitesses des roues par :

$$V'_{CR} = r \dot{\theta}_r \tag{1.10a}$$

$$V'_{CL} = r \dot{\theta}_l \tag{1.10b}$$

Avec V'_{CR} et V'_{CL} sont les vitesses des point de contact de la roue droite et gauche dans le repère locale respectivement. $\dot{\theta}_r$ est la vitesse de rotation de la roue droite et $\dot{\theta}_l$ est la vitesse de rotation de la roue gauche et r est le rayon de chaque roue.

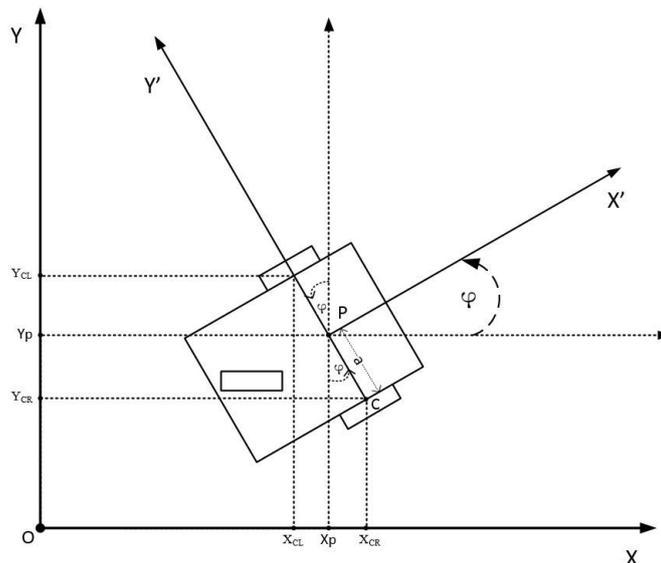


Figure I.7 Relation entre les coordonnées du point P et les point de contacts des roues.

Dans le repère monde, ces vitesses peuvent être calculées en fonction des vitesses du point centre P. Suivant la figure Fig. I.7 on a:

$$x_{CR} = x_p + a \sin(\varphi) \quad (1.11a)$$

$$y_{CR} = y_p - a \cos(\varphi) \quad (1.11b)$$

$$x_{CL} = x_p - a \sin(\varphi) \quad (1.11c)$$

$$y_{CL} = y_p + a \cos(\varphi) \quad (1.11d)$$

« a » étant la distance entre le point P et le point C. En dérivant, on trouve les vitesses des points de contact des roues droite et gauche :

$$\dot{x}_{CR} = \dot{x}_p + a \dot{\varphi} \cos(\varphi) \quad (1.12a)$$

$$\dot{y}_{CR} = \dot{y}_p + a \dot{\varphi} \sin(\varphi) \quad (1.12b)$$

$$\dot{x}_{CL} = \dot{x}_p - a \dot{\varphi} \cos(\varphi) \quad (1.12c)$$

$$\dot{y}_{CL} = \dot{y}_p - a \dot{\varphi} \sin(\varphi) \quad (1.12d)$$

En utilisant la matrice de rotation $R(\varphi)$, on peut trouver les vitesses V'_{CR} et V'_{CL} dans le repère monde :

$$\begin{bmatrix} \dot{x}_{CR} \\ \dot{y}_{CR} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} \dot{x}'_{CR} \\ \dot{y}'_{CR} \\ \dot{\varphi}' \end{bmatrix} \quad (1.13)$$

Où $\dot{y}'_{CR} = 0$ de l'hypothèse I, on trouve:

$$\dot{x}_{CR} = \dot{x}'_{CR} \cos(\varphi) \quad (1.14a)$$

$$\dot{y}_{CR} = \dot{x}'_{CR} \sin(\varphi) \quad (1.14b)$$

$$V'_{CR} = \dot{x}'_{CR} \quad (1.14c)$$

De Eq. (1.10a) et Eq. (1.12) et Eq. (1.14), et sachant que $\sin(\varphi)^2 + \cos(\varphi)^2 = 1$ on trouve la contrainte de roulement pour la roue droite :

$$\dot{x}_p \cos(\varphi) + \dot{y}_p \sin(\varphi) + a\dot{\varphi} = r\dot{\theta}_r \quad (1.15a)$$

En suivant les mêmes étapes pour la roue gauche on trouve la contrainte de roulement pour la roue gauche :

$$\dot{x}_p \cos(\varphi) + \dot{y}_p \sin(\varphi) - a\dot{\varphi} = r\dot{\theta}_l \quad (1.15b)$$

Dans le cas d'un roulement pur sans glissement on peut écrire les trois équations des contraintes sous la forme matricielle :

$$\Lambda(q) * \dot{q} = 0 \quad (1.16)$$

Telle que $\Lambda(q)$ est la matrice des contraintes non-holonomes définie comme :

$$\Lambda(q) = \begin{bmatrix} -\sin(\varphi) & \cos(\varphi) & 0 & 0 & 0 \\ \cos(\varphi) & \sin(\varphi) & a & -r & 0 \\ \cos(\varphi) & \sin(\varphi) & -a & 0 & -r \end{bmatrix} \quad (1.17)$$

Dont \dot{q} représente la dérivée du vecteur des coordonnées généralisées q :

$$\dot{q} = [\dot{x}_p \quad \dot{y}_p \quad \dot{\varphi} \quad \dot{\theta}_r \quad \dot{\theta}_l]^T \quad (1.18)$$

Les vitesses linéaires des roues sont exprimées comme :

$$V_r = r\dot{\theta}_r \quad (1.19a)$$

$$V_l = r\dot{\theta}_l \quad (1.19b)$$

I-5-2. LA CINÉMATIQUE DU ROBOT ET DU GLISSEMENT :

Pour modéliser la cinématique d'un robot à roues qui subit des glissements longitudinaux et latéraux, on introduit deux variables $\dot{\zeta}_r$ et $\dot{\zeta}_l$ pour les glissements longitudinaux de la roue droite et gauche respectivement, et deux autres variables $\dot{\eta}_r$ et $\dot{\eta}_l$ pour les glissements latéraux [12][19].

En cas de glissement longitudinal, la forme précédente des vitesses V_r et V_l Eq. (1.19) n'a aucun sens. Dans ce cas les nouvelles variables $\dot{\rho}_r$ et $\dot{\rho}_l$ représentent les nouvelles vitesses linéaires des roues droite et gauche respectivement:

$$\dot{\rho}_r = (r\dot{\theta}_r - \dot{\zeta}_r)\cos(\delta_r) \quad (1.20a)$$

$$\dot{\rho}_l = (r\dot{\theta}_l - \dot{\zeta}_l)\cos(\delta_l) \quad (1.20b)$$

$\dot{\zeta}_r$ et $\dot{\zeta}_l$ sont les vitesses de glissement longitudinal des deux roues et δ_r, δ_l sont les angles de braquage de la roue droite et gauche. Dans notre cas les roues ont un axe commun et ne sont pas guidées, dans ce cas $\delta_r = \delta_l = 0$.

De même pour le glissement latéral, l'hypothèse I de la section I.5.1 n'est plus valide ce qui veut dire qu'on a un déplacement sur l'axe latéral du repère local du robot $\dot{y}'_p \neq 0$. Ce déplacement est présenté par les vitesses des glissements latéraux :

$$\dot{y}'_r = \dot{\eta}_r \cos(\delta_r) \quad (1.21a)$$

$$\dot{y}'_l = \dot{\eta}_l \cos(\delta_l) \quad (1.21b)$$

Avec $\dot{\eta}_r$ et $\dot{\eta}_l$ qui sont les vitesses des glissements latéraux des roues et $\delta_r = \delta_l = 0$ car les roues ne sont pas guidées Fig. I.8.

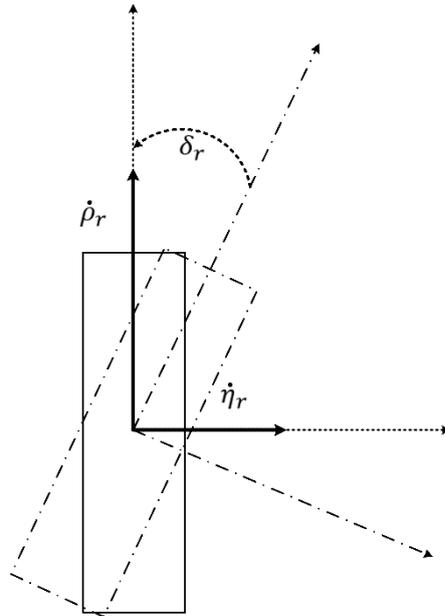


Figure I.8 Représentation des vitesses de glissement

On peut constater que les glissements sont des petites violations des contraintes non-holonomes qui deviennent :

$$-\dot{x}_p \sin(\varphi) + \dot{y}_p \cos(\varphi) = \dot{\eta}_r \quad (1.22a)$$

$$-\dot{x}_p \sin(\varphi) + \dot{y}_p \cos(\varphi) = \dot{\eta}_l \quad (1.22b)$$

$$\dot{x}_p \cos(\varphi) + \dot{y}_p \sin(\varphi) + a\dot{\varphi} = \dot{\rho}_r \quad (1.22c)$$

$$\dot{x}_p \cos(\varphi) + \dot{y}_p \sin(\varphi) - a\dot{\varphi} = \dot{\rho}_l \quad (1.22d)$$

En considérant la rotation du robot autour le centre de gravité G (Fig. I.9), les relations entre les vitesses de point d'axe P et le centre de gravité du robot G sont :

$$\dot{x}_G = \dot{x}_p + d \dot{\varphi} \sin(\varphi) \quad (1.23a)$$

$$\dot{y}_G = \dot{y}_p - d \dot{\varphi} \cos(\varphi) \quad (1.23b)$$

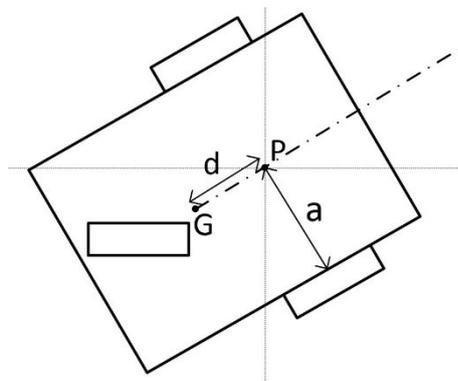


Figure I.9 Localisation du point G

Donc les équations des contraintes non-holonomes deviennent :

$$-\dot{x}_G \sin(\varphi) + \dot{y}_G \cos(\varphi) + d\dot{\varphi} = \dot{\eta}_r \quad (1.24a)$$

$$-\dot{x}_G \sin(\varphi) + \dot{y}_G \cos(\varphi) + d\dot{\varphi} = \dot{\eta}_l \quad (1.24b)$$

$$\dot{x}_G \cos(\varphi) + \dot{y}_G \sin(\varphi) + a\dot{\varphi} = \dot{\rho}_r \quad (1.24c)$$

$$\dot{x}_G \cos(\varphi) + \dot{y}_G \sin(\varphi) - a\dot{\varphi} = \dot{\rho}_l \quad (1.24d)$$

En écrivant ces équations sous la forme matricielle comme l'équation Eq. (1.16), on obtient la matrice des contraintes suivante :

$$\Lambda(q) = \begin{bmatrix} -\sin(\varphi) & \cos(\varphi) & d & -1 & 0 & 0 & 0 & 0 & 0 \\ -\sin(\varphi) & \cos(\varphi) & d & 0 & -1 & 0 & 0 & 0 & 0 \\ \cos(\varphi) & \sin(\varphi) & a & 0 & 0 & -1 & 0 & 0 & 0 \\ \cos(\varphi) & \sin(\varphi) & -a & 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \quad (1.25)$$

Avec \dot{q} le nouveau vecteur des vitesses généralisées

$$\dot{q} = [\dot{x}_G \quad \dot{y}_G \quad \dot{\varphi} \quad \dot{\eta}_r \quad \dot{\eta}_l \quad \dot{\rho}_r \quad \dot{\rho}_l \quad \dot{\theta}_r \quad \dot{\theta}_l]^T \quad (1.26)$$

Par l'utilisation des Eq. (1.24) on peut trouver la cinématique du robot sous la forme :

$$\dot{q}(t) = S(q) * v(t) \quad (1.27)$$

Où $v(t)$ est un vecteur des vitesses comprenant les vitesses de glissement et les vitesses de rotation des roues

$$v(t) = [\dot{\eta}_l \quad \dot{\rho}_r \quad \dot{\rho}_l \quad \dot{\theta}_r \quad \dot{\theta}_l]^T \quad (1.28)$$

On note que les deux roues sont fixées au même axe et ne sont pas guidées ce qui nous donne $\dot{\eta}_l = \dot{\eta}_r$.

$S(q)$ est une matrice de rang complet son produit avec la matrice des contraintes non-holonomes est nul :

$$S(q)^T * \Lambda(q)^T = 0 \quad (1.29)$$

La manipulation des équations Eq. (1.24) nous donne :

$$2\dot{\eta}_l = -2\dot{x}_G \sin(\varphi) + 2\dot{y}_G \cos(\varphi) + 2d\dot{\varphi} \quad (1.30a)$$

$$\dot{\rho}_l + \dot{\rho}_r = 2\dot{x}_G \sin(\varphi) + 2\dot{y}_G \cos(\varphi) \quad (1.30b)$$

$$\dot{\rho}_r - \dot{\rho}_l = 2a\dot{\varphi} \quad (1.30c)$$

De Eq. (1.30) on obtient

$$\dot{x}_G = -\dot{\eta}_l \sin(\varphi) + \dot{\rho}_r \left(\frac{a \cos(\varphi) + d \sin(\varphi)}{2a} \right) + \dot{\rho}_l \left(\frac{a \cos(\varphi) - d \sin(\varphi)}{2a} \right) \quad (1.31a)$$

$$\dot{y}_G = \dot{\eta}_l \cos(\varphi) + \dot{\rho}_r \left(\frac{a \sin(\varphi) - d \cos(\varphi)}{2a} \right) + \dot{\rho}_l \left(\frac{a \sin(\varphi) + d \cos(\varphi)}{2a} \right) \quad (1.31b)$$

$$\dot{\varphi} = \frac{1}{2a} \dot{\rho}_r - \frac{1}{2a} \dot{\rho}_l \quad (1.31c)$$

Si on écrit les équations (1.31) sous la forme Eq. (1.27) on obtient la cinématique du robot mobile [12][13], où la matrice $S(q)$ est :

$$S(q) = \begin{bmatrix} -\sin(\varphi) & \frac{a \cos(\varphi) + d \sin(\varphi)}{2a} & \frac{a \cos(\varphi) - d \sin(\varphi)}{2a} & 0 & 0 \\ \cos(\varphi) & \frac{a \sin(\varphi) - d \cos(\varphi)}{2a} & \frac{a \sin(\varphi) + d \cos(\varphi)}{2a} & 0 & 0 \\ 0 & \frac{1}{2a} & -\frac{1}{2a} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.32)$$

I-6. MODELISATION DYNAMIQUE :

La dynamique est l'étude du mouvement d'un système mécanique en tenant compte les différentes forces qui affectent son mouvement. Le modèle dynamique est essentiel pour la simulation et l'analyse du mouvement du robot, et pour la conception des variétés des algorithmes de commande.

Un robot non-holonome avec un vecteur de coordonnées généralisées q , et soumis à des contraintes cinématiques peut être décrit par l'équation de mouvement suivante [18][19][12]:

$$M(q)\ddot{q} + C(q, \dot{q}) = B(q)\tau + F(\dot{q}) + \Lambda^T(q)\lambda \quad (1.33)$$

Où $M(q)$ est une matrice d'inertie symétrique définie positive. $C(q, \dot{q})$ est la matrice des forces centrifuge et de Coriolis. $B(q)$ est une matrice constante non singulière. $F(\dot{q})$ la matrice de frottement de la surface. $\Lambda^T(q)$ est la matrice des contraintes non-holonomes. λ le vecteur des multiplicateurs de Lagrange.

Il y a deux méthodologies pour obtenir la dynamique d'un système :

- La méthode de Newton-Euler.

- La méthode de Lagrange-Euler.

Dans ce travail on va voir la méthode de Newton-Euler.

I-6-1. MODELE DYNAMIQUE DE NEWTON-EULER :

Ce modèle est obtenu par l'application directe des équations de Newton-Euler en translation et en rotation, après l'isolement des solides du robot (les roues – le corps sans les roues) [12].
 Suivant ces deux diagrammes du corps libre :

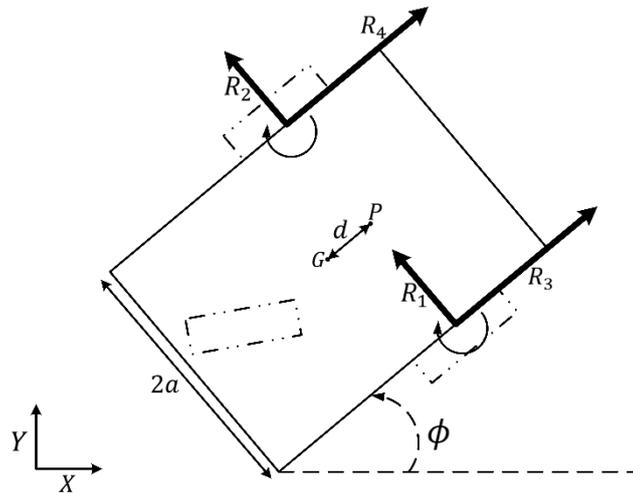


Figure I.10 Diagramme du corps libre de la plateforme du robot sans les roues

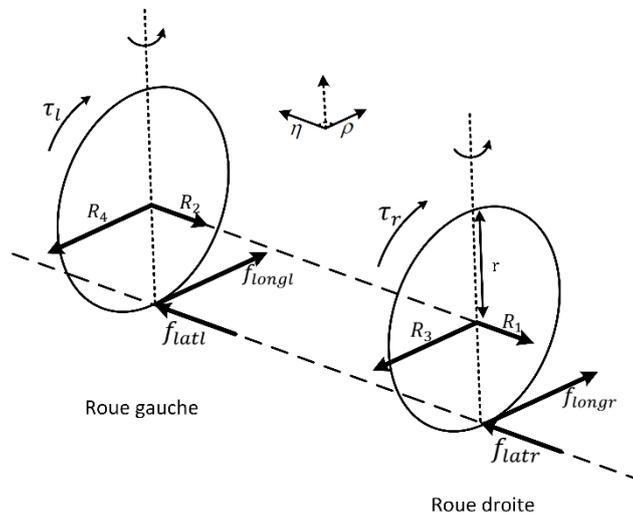


Figure I.11 Diagramme du corps libre des roues

Sachant que les équations de Newton-Euler sont [20]:

Pour le mouvement de translation

$$m * \dot{v} = F \tag{1.34a}$$

Où m est la masse du corps et \dot{v} est l'accélération de ce corps et F la somme des forces exercées sur le centre de gravité du corps.

Pour le mouvement de rotation

$$I * \dot{\omega} = N \quad (1.34b)$$

Où I est le moment d'inertie du corps et $\dot{\omega}$ est l'accélération angulaire de ce corps et N est la somme des couples qui produisent cette rotation autour le centre de gravité.

Du diagramme du corps libre de la plateforme (robot sans les roues) Fig. I.10, on dérive les trois premières équations dynamiques : les deux premières équations concernant la translation du corps du robot et une équation concernant sa rotation. Du deuxième diagramme du corps libre des roues Fig. I.11 on a quatre autres équations dynamiques concernant la translation des roues en tenant compte des forces centripètes, et deux autres équations concernant la rotation. On note que $\ddot{\eta}_r$ et $\ddot{\eta}_l$ sont les accélérations de glissement latérale pour les deux roues droite et gauche respectivement, $\ddot{\rho}_r$ et $\ddot{\rho}_l$ sont les accélérations de glissement longitudinal pour les roues droite et gauche.

X-Y : Les coordonnées de repère monde.

$\rho - \eta$: Les coordonnées du repère fixé sur chaque roue.

m_r : La masse de la plateforme du robot sans les roues, les rotors et les moteurs.

m_w : La masse de chaque roue plus le rotor du moteur.

I_{rz} : Le moment d'inertie de la plateforme du robot sans les roues, les rotors et les moteurs, autour l'axe vertical qui passe par le point G centre de gravité.

I_{wz} : Le moment d'inertie de chaque roue avec le rotor du moteur, autour le diamètre de la roue.

I_{wy} : Le moment d'inertie de chaque roue avec le rotor du moteur, autour l'axe des roues.

R_i : Les forces de réaction entre le corps du robot et les deux roues motrices.

$f_{long(r,l)}$: La force de traction longitudinale pour chaque roue motrice.

$f_{lat(r,l)}$: La force de traction latérale pour chaque roue motrice.

$\tau_{(r,l)}$: Le couple moteur agissant sur l'axe de chaque roue généré par son moteur.

On écrit

$$\left| \begin{array}{l} m_r \ddot{x}_G = -(R_1 + R_2) \sin(\varphi) + (R_3 + R_4) \cos(\varphi) \\ m_r \ddot{y}_G = (R_1 + R_2) \cos(\varphi) + (R_3 + R_4) \sin(\varphi) \end{array} \right. \quad (1.35a)$$

$$\left| \begin{array}{l} m_r \ddot{x}_G = -(R_1 + R_2) \sin(\varphi) + (R_3 + R_4) \cos(\varphi) \\ m_r \ddot{y}_G = (R_1 + R_2) \cos(\varphi) + (R_3 + R_4) \sin(\varphi) \end{array} \right. \quad (1.35b)$$

$$(I_{rz} + 2 I_{wz}) \ddot{\phi} = d (R_1 + R_2) + a (R_3 - R_4) \quad (1.35c)$$

$$m_w \ddot{\eta}_r + m_w \dot{\phi} \dot{\rho}_r = f_{latr} - R_1 \quad (1.35d)$$

$$m_w \ddot{\eta}_l + m_w \dot{\phi} \dot{\rho}_l = f_{latl} - R_2 \quad (1.35e)$$

$$m_w \ddot{\rho}_r - m_w \dot{\phi} \dot{\eta}_r = f_{longr} - R_3 \quad (1.35f)$$

$$m_w \ddot{\rho}_l - m_w \dot{\phi} \dot{\eta}_l = f_{longl} - R_4 \quad (1.35g)$$

$$I_{wy} \ddot{\theta}_r = \tau_r - f_{longr} r \quad (1.35h)$$

$$I_{wy} \ddot{\theta}_l = \tau_l - f_{longl} r \quad (1.35i)$$

Ces équations peuvent être écrites sous forme compacte comme l'expression donnée par Eq. (1.33), telle que :

$$M(q) = \begin{bmatrix} m_r & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & m_r & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I_{rz} + 2I_{wz} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_w & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & m_w & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & m_w & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & m_w & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_{wy} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_{wy} \end{bmatrix} \quad (1.36)$$

Et le vecteur $C(q, \dot{q})$ comprend les forces centrifuges appliquées sur les deux roues telle que:

$$C(q, \dot{q}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ m_w \dot{\phi} \dot{\rho}_r \\ m_w \dot{\phi} \dot{\rho}_l \\ -m_w \dot{\phi} \dot{\eta}_r \\ -m_w \dot{\phi} \dot{\eta}_l \\ 0 \\ 0 \end{bmatrix} \quad (1.37)$$

Le vecteur λ qui contient les forces des réactions telle que

$$\lambda = [R_1 \ R_2 \ R_3 \ R_4]^T \quad (1.38)$$

La matrice $B(q)$ est

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (1.39)$$

Et le vecteur des forces de traction est

$$F(\dot{q}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ f_{latr} \\ f_{latl} \\ f_{longr} \\ f_{longl} \\ -rf_{longr} \\ -rf_{longl} \end{bmatrix} \quad (1.40)$$

Avec

$$q = [x_G \quad y_G \quad \varphi \quad \eta_r \quad \eta_l \quad \rho_r \quad \rho_l \quad \theta_r \quad \theta_l]^T \quad (1.41a)$$

$$\dot{q} = [\dot{x}_G \quad \dot{y}_G \quad \dot{\varphi} \quad \dot{\eta}_r \quad \dot{\eta}_l \quad \dot{\rho}_r \quad \dot{\rho}_l \quad \dot{\theta}_r \quad \dot{\theta}_l]^T \quad (1.41b)$$

$$\ddot{q} = [\ddot{x}_G \quad \ddot{y}_G \quad \ddot{\varphi} \quad \ddot{\eta}_r \quad \ddot{\eta}_l \quad \ddot{\rho}_r \quad \ddot{\rho}_l \quad \ddot{\theta}_r \quad \ddot{\theta}_l]^T \quad (1.41c)$$

I-7. MODELE GENERALE DU ROBOT :

Le modèle générale du robot mobile à roues est une combinaison de deux modèles, cinématique et dynamique [12]. On continue à développer le modèle en dérivant Eq. (1.27) on obtient:

$$\ddot{q}(t) = S(q) * \dot{v}(t) + \dot{S}(q) * v(t) \quad (1.52)$$

Où

$$v(t) = [\dot{\eta}_l \quad \dot{\rho}_r \quad \dot{\rho}_l \quad \dot{\theta}_r \quad \dot{\theta}_l]^T$$

$$\dot{v}(t) = [\ddot{\eta}_l \quad \ddot{\rho}_r \quad \ddot{\rho}_l \quad \ddot{\theta}_r \quad \ddot{\theta}_l]^T$$

Et la matrice dérivée de la matrice $S(q)$ Eq. (1.32) est

$$\dot{S}(q) = \begin{bmatrix} -\dot{\varphi} \cos(\varphi) & \frac{-a\dot{\varphi} \sin(\varphi) + d\dot{\varphi} \cos(\varphi)}{2a} & \frac{-a\dot{\varphi} \sin(\varphi) - d\dot{\varphi} \cos(\varphi)}{2a} & 0 & 0 \\ -\dot{\varphi} \sin(\varphi) & \frac{a\dot{\varphi} \cos(\varphi) + d\dot{\varphi} \sin(\varphi)}{2a} & \frac{a\dot{\varphi} \cos(\varphi) - d\dot{\varphi} \sin(\varphi)}{2a} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.53)$$

On substitue Eq. (1.27) et Eq. (1.52) dans le modèle dynamique du robot Eq. (1.33) comme suit :

$$M(q)(S(q) * \dot{v}(t) + \dot{S}(q) * v(t)) + C(q, \dot{q}) = B(q)\tau + F(\dot{q}) + \Lambda^T(q)\lambda \quad (1.54)$$

Si on multiplie l'équation Eq. (1.54) par $S(q)^T$ comme suit

$$(S(q)^T M(q) S(q)) \dot{v}(t) + (S(q)^T M(q) \dot{S}(q)) v(t) + S(q)^T C(q, \dot{q}) = S(q)^T B(q) \tau + S(q)^T F(\dot{q}) + S(q)^T \Lambda^T(q) \lambda \quad (1.55)$$

Et sachant qu'on a la relation Eq. (1.29) on peut simplifier l'équation Eq. (1.55) et ça nous donne :

$$(S^T M S) \dot{v}(t) + (S^T M \dot{S}) v(t) + S^T C = S^T B \tau + S^T F \quad (1.56)$$

En écrivant l'équation Eq. (1.56) en une autre forme, on peut séparer les équations des couples comme suit :

$$\overline{(S^T M S)} \dot{v}(t) + \overline{(S^T M \dot{S})} \bar{v}(t) + \overline{S^T C} = \overline{S^T F} \quad (1.57a)$$

$$I_{wy} \ddot{\theta}_r = \tau_r - f_{longr} r \quad (1.57b)$$

$$I_{wy} \ddot{\theta}_l = \tau_l - f_{longl} r \quad (1.57c)$$

On définit :

$$\bar{M} = \overline{S^T M S}$$

$$\overline{C}(q, \dot{q}) = \overline{S^T M \dot{S}} + \overline{S^T H S} \quad \text{Où} \quad \overline{S^T C} = \overline{S^T H S} * \bar{v}(t)$$

$$\bar{F} = \overline{S^T F}$$

Les équations complètes qui représentent le modèle mathématique d'un robot mobile non holonomique à conduite différentielle qui subit des glissements sont :

$$\left| \begin{array}{l} \dot{\bar{q}} = \bar{S} \bar{v} \end{array} \right. \quad (1.58a)$$

$$\left| \begin{array}{l} \bar{M} \dot{\bar{v}} + \overline{C}(q, \dot{q}) \bar{v} = \bar{F} \end{array} \right. \quad (1.58b)$$

$$\left| \begin{array}{l} \bar{I} \dot{\Theta} = \bar{\tau} - r \bar{f} \end{array} \right. \quad (1.58c)$$

Où \bar{M} est la matrice d'inertie :

$$\bar{M} = \begin{bmatrix} m_r + 2m_w & \frac{-dm_r}{2a} & \frac{dm_r}{2a} \\ \frac{-dm_w}{2a} & \frac{(I_{rz} + 2I_{wz} + a^2 m_r + 4a^2 m_w + d^2 m_r)}{4a^2} & \frac{-(-m_r a^2 + m_r d^2 + I_{rz} + 2I_{wz})}{4a^2} \\ \frac{dm_r}{2a} & \frac{-(-m_r a^2 + m_r d^2 + I_{rz} + 2I_{wz})}{4a^2} & \frac{(I_{rz} + 2I_{wz} + a^2 m_r + 4a^2 m_w + d^2 m_r)}{4a^2} \end{bmatrix}$$

Et $\overline{C}(q, \dot{q})$ est la matrice qui contient les forces centrifuges et Coriolis :

$$\overline{C}(\bar{q}, \dot{\bar{q}}) = \begin{bmatrix} 0 & \frac{\dot{\varphi}(m_r + 2m_w)}{2} & \frac{\dot{\varphi}(m_r + 2m_w)}{2} \\ \frac{-\dot{\varphi}(m_r + 2m_w)}{2} & 0 & \frac{-d\dot{\varphi}m_r}{2a} \\ \frac{-\dot{\varphi}(m_r + 2m_w)}{2} & \frac{d\dot{\varphi}m_r}{2a} & 0 \end{bmatrix}$$

\bar{F} est le vecteur des forces de traction :

$$\bar{F} = \begin{bmatrix} f_{latr} + f_{latl} \\ f_{longr} \\ f_{longl} \end{bmatrix}$$

Et \bar{I} contient les moments d'inertie autour l'axe de rotation des deux roues :

$$\bar{I} = \begin{bmatrix} I_{wy} & 0 \\ 0 & I_{wy} \end{bmatrix}$$

Et $\dot{\Theta}$ contient les accélérations angulaires des roues :

$$\dot{\Theta} = \begin{bmatrix} \ddot{\theta}_r \\ \ddot{\theta}_l \end{bmatrix}$$

$\bar{\tau}$ contient les couples moteurs des roues droite et gauche :

$$\bar{\tau} = \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix}$$

Et \bar{f} contient les forces de tractions longitudinales des deux roues

$$\bar{f} = \begin{bmatrix} f_{longr} \\ f_{longl} \end{bmatrix}$$

Et l'équation (1.58a) est la cinématique du robot après la séparation des couples

Où

$$\dot{\bar{q}} = [\dot{x}_G \quad \dot{y}_G \quad \dot{\varphi} \quad \dot{\eta}_r \quad \dot{\eta}_l \quad \dot{\rho}_r \quad \dot{\rho}_l]^T$$

Et

$$\bar{v}(t) = [\dot{\eta}_l \quad \dot{\rho}_r \quad \dot{\rho}_l]^T$$

Et

$$\bar{S}(q) = \begin{bmatrix} -\sin(\varphi) & \frac{a \cos(\varphi) + d \sin(\varphi)}{2a} & \frac{a \cos(\varphi) - d \sin(\varphi)}{2a} \\ \cos(\varphi) & \frac{a \sin(\varphi) - d \cos(\varphi)}{2a} & \frac{a \sin(\varphi) + d \cos(\varphi)}{2a} \\ 0 & \frac{1}{2a} & -\frac{1}{2a} \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Dans ce chapitre, nous avons modélisé mathématiquement un robot mobile à roues non holonome à conduite différentielle, tout en tenant compte de glissement et les forces de traction qui influent sévèrement sur son mouvement. Les forces de traction sont déterminées à l'aide d'un modèle d'interaction roues-chaussée.

| CHAPITRE II : INTERACTION ROUE-CHAUSSEE

II-1. Le taux de glissement et l'angle de dérive

II-2. Les forces de traction

II-3. Modèle d'interaction roue-chaussée

Dans le chapitre précédent, nous avons vu comment modéliser un robot mobile à roues lorsque l'hypothèse du roulement pur n'est plus vraie, chose qui nous a conduits à ajouter les forces de traction afin de modéliser l'effet du glissement sur le robot.

Dans ce chapitre et tandis qu'on essaye de donner plus d'informations sur le phénomène de glissement, on définit le taux de glissement et l'angle de dérive qui nous donnent une vue sur les glissements longitudinaux et latéraux. Puis on va essayer d'expliquer comment la traction est générée et à la fin on donnera un modèle mathématique d'interaction pneu-chaussée.

II-1. LE TAUX DE GLISSEMENT ET L'ANGLE DE DERIVE :

Comme il a été mentionné dans le chapitre précédent, le glissement peut être considéré comme une violation des contraintes non-holonomes. Si on veut le traiter on doit quantifier cette violation, cela se fait à travers deux proportions : le taux de glissement pour le glissement longitudinal et l'angle de dérive pour le glissement latéral.

II-1-1. LE TAUX DE GLISSEMENT :

Soit une roue en mouvement caractérisée par :

- V sa vitesse linéaire décomposé en $(\dot{\rho}, \dot{\eta})$ dans le plan comme illustré à la figure Fig. 1.8.
- $\dot{\theta}$ sa vitesse de rotation.
- r son rayon.

On appelle le taux de glissement de cette roue, s défini par [23] :

$$\left| \begin{array}{l} s = \frac{r\dot{\theta} - \dot{\rho}}{r\dot{\theta}} \text{ si } \dot{\rho} \text{ et } \dot{\theta} \text{ sont de même signe, et } |\dot{\theta}| \geq |\dot{\rho}| \text{ (phase d'accélération)} \\ s = \frac{r\dot{\theta} - \dot{\rho}}{\dot{\rho}} \text{ si } \dot{\rho} \text{ et } \dot{\theta} \text{ sont de même signe, et } |\dot{\theta}| < |\dot{\rho}| \text{ (phase de freinage)} \end{array} \right.$$

Plus les deux conditions limites :

$$\left| \begin{array}{l} s = 1 \quad \text{dans le cas limite : } \dot{\rho} \text{ négative et } \dot{\theta} \text{ positive} \\ s = -1 \quad \text{dans le cas limite : } \dot{\theta} \text{ négative et } \dot{\rho} \text{ positive} \end{array} \right.$$

On peut écrire s de façon plus compacte :

$$s = \frac{r\dot{\theta} - \dot{\rho}}{\max(|r\dot{\theta}|, |\dot{\rho}|)} \quad (2.1)$$

Si on pose que $r\dot{\theta}$ et $\dot{\rho}$ sont dans le même sens, l'interprétation du taux de glissement et comme suit :

- Si $r\dot{\theta} \neq 0$ et $\dot{\rho} = 0$ alors $s = 1$ nous sommes dans une situation de patinage totale.
- Si $r\dot{\theta} > \dot{\rho} > 0$ alors $s \in]0; 1[$ nous sommes dans une phase d'accélération.
- Si $r\dot{\theta} = \dot{\rho} > 0$ alors $s = 0$ nous sommes dans une phase de roulement sans glissement.
- Si $\dot{\rho} > r\dot{\theta} > 0$ alors $s \in]-1; 0[$ nous sommes dans une phase de freinage.
- Si $r\dot{\theta} = 0$ et $\dot{\rho} \neq 0$ alors $s = -1$ nous sommes dans une situation de roue bloqué.

Il est évident que le taux de glissement s indique bien la présence d'un glissement longitudinal.

II-1-2. L'ANGLE DE DERIVE :

Soit une roue en mouvement caractérisée par :

- V sa vitesse linéaire décomposé en $(\dot{\rho}, \dot{\eta})$ dans le plan comme illustré à la figure Fig. 1.8.
- $\dot{\theta}$ sa vitesse de rotation.
- r son rayon.

On appelle dérive de cette roue, le scalaire noté δ , défini par [23] :

$$\delta = \arctan\left(\frac{\dot{\eta}}{|\dot{\rho}|}\right) \quad (2.2)$$

Dans un mouvement pur $\dot{\eta} = 0$ alors $\delta = 0$.

Il est clair que l'angle de dérive δ indique bien la présence d'un glissement latéral.

II-2. LES FORCES DE TRACTIONS :

Il semble que les pneus se déforment quand ils sont mis en contact avec le sol en raison de leurs caractéristiques pneumatiques et les forces exercées par le robot. Ce qui rend l'hypothèse du roulement pur (I.5.1) invalide car il n'y a plus un seul point de contact entre la roue et la chaussée mais une aire.

Au niveau moléculaire et dans cette aire de contact, les aspérités de la dernière couche du pneu vont pénétrer les aspérités de la surface de chaussée Fig. II.1 grâce aux forces exercées sur la roue (le poids de la plateforme – le couple moteur ou de freinage – les forces centrifuges, etc.). Lorsque cela se produit, il donne une force résistive ou une force de traction, à côté de cette déformation et grâce à la nature collante du caoutchouc, les pneus ont une adhérence qui joue un rôle important à la génération de ces forces.



Figure II.1 Contact des aspérités avec charge et sans charge

La déformation, l'usure et l'adhérence des pneus sont parmi les causes de la traction. Ils sont affectés par plusieurs facteurs comme la pression de gonflage, l'état de la surface, la température, le poids et la quantité de glissement, ce qui augmente la difficulté de décrire analytiquement les forces de traction [21]. Pour des raisons de simplification on s'intéresse dans ce travail, qu'à forces générées par la déformation tangentielle du pneu, qui peuvent être décomposées en deux : la force longitudinale et la force latérale.

II-3. MODELE D'INTERACTION ROUE-CHAUSSEE

Dans la littérature plusieurs chercheurs ont essayé de donner des modèles mathématiques d'interaction roue-chaussée, lesquels on peut classifier en deux branches majeures : les modèles empiriques et les modèles analytiques [24].

Les modèles empiriques sont basés sur les techniques d'ajustement de courbe, ils peuvent capter avec justesse les caractéristiques non-linéaires des forces de traction. Mais la plupart de ces modèles manquent l'interprétation physique. En outre ils ne peuvent pas refléter directement l'effet de certains facteurs dynamiques tel que l'hystérésis, la pression du pneu et l'humidité de la surface. Par contre les modèles analytiques qui sont composés d'équations différentielles peuvent modéliser ces facteurs dynamiques. Mais en manquant la justesse empirique. Le tableau II.1 présente la plupart des modèles de traction trouvés dans la littérature [45] :

Tableau II.1 Les modèles d'interaction Roue-Chaussé dans la littérature.

Nom de modèle	Type	Commentaire
Piecewise linéaire	Empirique	Un modèle très simple. L'approximation se fait à travers les fonctions Piecewise, mais il ne peut pas capturer les non-linéarités de l'interaction pneu-chaussée [25][26].
Modèle Burkhardt	Semi-empirique	Il est fréquemment mentionné et peut décrire l'interaction mais à un cout très élevé [27].
Modèle Rill	Semi-empirique	Un modèle simple basé sur les caractéristique du régime permanent couple/force [28].
Modèle Delft La Formule Magique	Semi-empirique	le modèle le plus utilisé et le plus important. Il peut décrire l'interaction avec justesse mais Il est considéré sur-paramétré [29][30][31].
Modèle Dahl	Analytique	Un ancien modèle basé sur la généralisation de frottement de Coulomb [32][33].
Bliman-Sorine	Analytique	Il peut capter plusieurs phénomènes de frottement, amélioration sur le modèle de Dahl [34][35].
Modèle LuGre	Analytique	Amélioration sur Bliman-Sorine. Il combine les mérites de Dahl et les caractéristiques de régime permanant [36][37].
Proportionnel linéaire	Empirique	Simple mais il ne peut pas refléter la non-linéarité : saturation [38][39].
Proportionnel non-linéaire	Semi-empirique	Il peut correctement ajuster les courbes [40][41][42].
Bicycle	Analytique	Il ne reflet pas les forces de traction directement [43][44].

II-3-1. LA FORMULE MAGIQUE :

Dans ce travail on a choisi parmi ces modèles, le modèle Delft connu par la « Formule Magique ». Qui est un modèle élégant semi empirique basé sur les techniques d'ajustement de courbe. Il a été largement accepté dans l'industrie et le secteur académique. Il a été introduit par Bakker, Nyborg & Pacejka en 1987 [29] et depuis lors il a été révisé plusieurs fois [30][31][46][47]. Ses avantages sont la justesse, la simplicité et l'aptitude d'être interprété physiquement, en outre il est composé de fonctions dérivables continues.

L'une des raisons pour lesquels cette formule est dite magique est qu'elle permet à partir de la même équation et un ensemble de six paramètres d'obtenir une représentation des forces longitudinales, et latérales ainsi que celle du couple d'auto-alignement. Les paramètres de taux de glissement ou l'angle de dérive forment l'entrée de la formule. Les autres paramètres (la charge verticale, l'angle de carrossage, l'adhérence du sol, la pression de gonflage du pneumatique) entrent dans la détermination des macro-coefficients, qui gouvernent la forme des courbes générées, comme montre la figure (Fig. II.2).

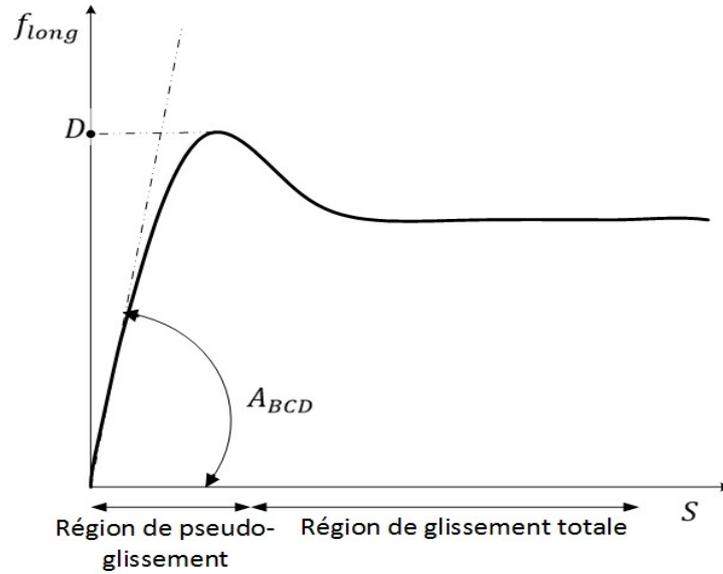


Figure II.2 Quelques paramètres indiqués sur la courbe de force longitudinale

La forme générale de la formule magique pour une charge et un angle de carrossage donnée est la suivante [48] :

$$y = D \sin \left(C \arctan(Bx - E(Bx - \arctan Bx)) \right) \quad (2.3a)$$

Avec

$$Y(X) = y(x) + S_v \quad (2.3b)$$

$$x = X + S_h \quad (2.3c)$$

Où :

- Y peut représenter la force latérale F_{lat} ou la force longitudinale F_{long} .
- X peut représenter l'angle de dérive du pneu δ lorsqu'on cherche à obtenir la force latérale, ou le taux de glissement s lorsqu'on cherche à obtenir la force longitudinale.

« D, C, B, E » sont les macro-coefficients et leurs définitions :

- D : coefficient d'amplitude correspond au maximum de la courbe d'adhérence.
- C : coefficient de forme permet d'ajuster l'allure de la courbe.
- B : coefficient de rigidité permet d'ajuster la pente à l'origine.
- E : coefficient de courbure influe la courbure de la caractéristique.

S_v et S_h sont pour introduire un décalage vertical et horizontal respectivement par rapport à l'origine.

On utilise la formule magique pour calculer les valeurs des efforts de tractions longitudinaux et latéraux qui sont essentiels pour notre modèle du robot. On calcule à chaque instant le taux de glissement Eq. (2.1), l'angle de dérive Eq. (2.2) et on les substitue dans les équations Eq.(2.3) .

II-3-1-1. FORCE DE TRACTION LONGITUDINALE :

La formule magique pour les forces de traction longitudinales des roues est [48] :

$$f_{longi} = D \sin \left(C \arctan(Bx_i - E(Bx_i - \arctan Bx_i)) \right) \quad (2.4a)$$

$$x_i = 100 * s_i \quad (2.4b)$$

Où

$$\mu_{long} = a_1 F_z + a_2 \quad (2.4c)$$

$$D = \mu_{long} F_z * 1000 \quad (2.4d)$$

$$C = a_0 \quad (2.4e)$$

$$B = \frac{A_{BCD}}{CD} \text{ avec } A_{BCD} = (a_3 F_z^2 + a_4) \exp(-a_5 F_z) \quad (2.4f)$$

$$E = a_6 F_z^2 + a_7 F_z + a_8 \quad (2.4g)$$

Où s_i est le taux de glissement de la i^{eme} roue, le taux doit être en pourcentage (%), c'est pourquoi la multiplication par 100 dans l'équation Eq. (2.4b). F_z est la charge nominale du robot en KN . μ_{long} peut être vu comme un coefficient de frottement. A_{BCD} est connu par le coefficient d'adhésivité. Les coefficients $a_0 a_1 \dots a_8$ sont des micro-coefficients et sont utilisés pour représenter les macro-coefficients D, C, B, E en fonction de la charge nominal F_z . Leur bon choix fait la différence entre les types des surface (tarmac sec , tarmac mouillé, la glace ,gravier meuble ...). La figure (Fig. II.3) montre la courbe de la force de traction longitudinale (Pour une charge $F_z = 154.644 N$. Les micro-coefficients utilisés dans la formule magique sont du tableau II.2 et II.3).

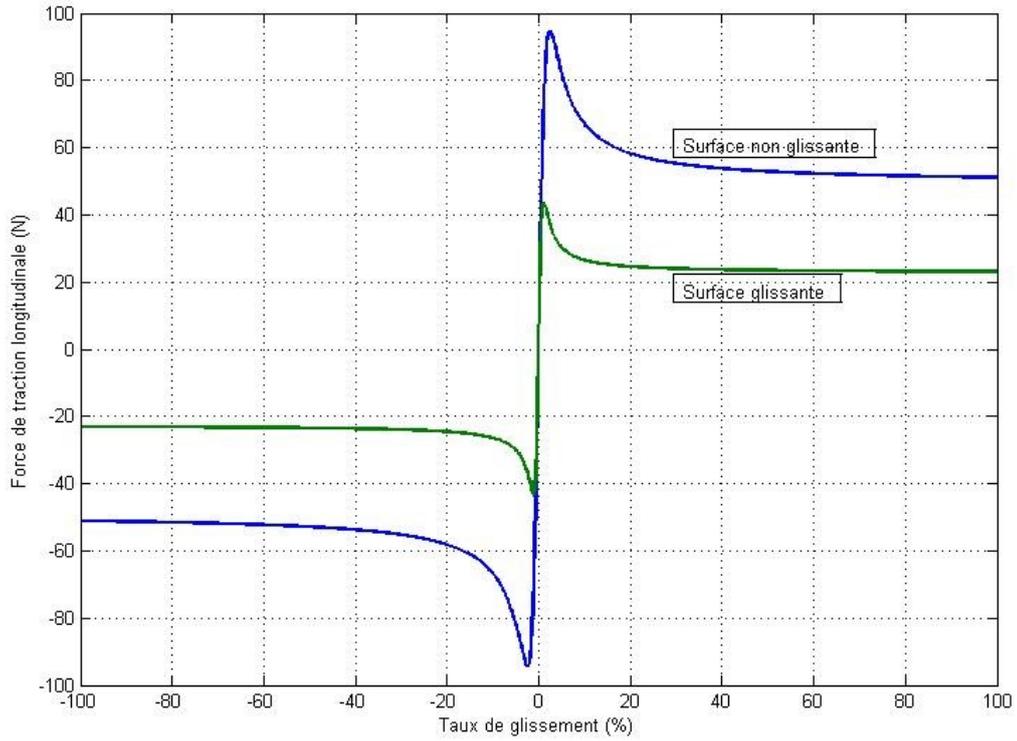


Figure II.3 Quelques exemples de courbes des forces de traction longitudinales.

II-3-1-2. FORCE DE TRACTION LATÉRALE :

La formule magique pour les forces de traction latérales des roues est [48]:

$$f_{lati} = D \sin \left(C \arctan \left(B x_i - E \left(B x_i - \arctan B x_i \right) \right) \right) \quad (2.5a)$$

$$x_i = \delta_i \frac{180}{\pi} \quad (2.5b)$$

Où

$$\mu_{lat} = b_1 F_z + b_2 \quad (2.5c)$$

$$D = \mu_{lat} F_z * 1000 \quad (2.5d)$$

$$C = b_0 \quad (2.5e)$$

$$B = \frac{B_{BCD}}{CD} \text{ avec } B_{BCD} = b_3 \sin \left(2 \arctan \left(\frac{F_z}{b_4} \right) \right) \quad (2.5f)$$

$$E = b_6 F_z + b_7 \quad (2.5g)$$

Où δ_i est l'angle de dérive de la i^{eme} roue. L'angle est en degré donc la conversion est nécessaire dans Eq. (2.5d) . F_z est la charge nominale du robot en KN. μ_{lat} peut être vu

comme un coefficient de frottement. B_{BCD} est connu par le coefficient d'adhésivité. b_0, b_1, \dots, b_8 sont des micro-coefficients et sont utilisés pour représenter les macro-coefficients D, C, B, E en fonction de la charge nominale F_z . Leur bon choix notamment b_1 et b_2 fait la différence entre les types des surfaces (chaussée sèche, chaussée mouillée, la glace, gravier meuble ...). La figure Fig. II.4 montre la courbe de la force de traction latérales (Pour une charge $F_z = 154.644 N$. Les micro-coefficients utilisés dans la formule magique sont du tableau II.2 et II.3).

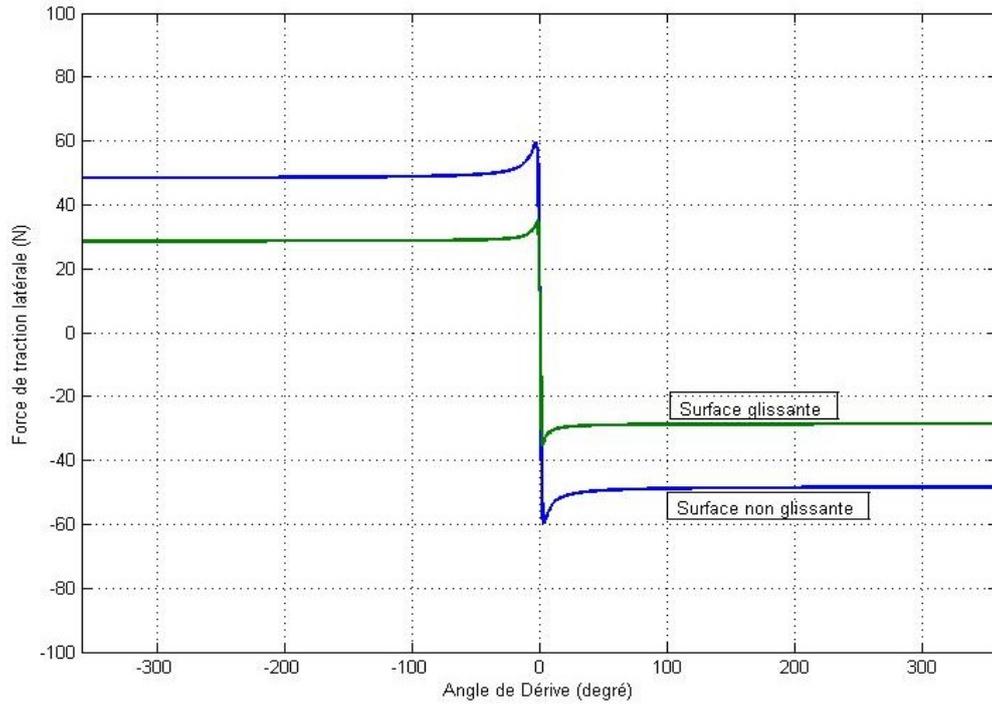


Figure II.4 Quelques exemples de courbes des forces de traction latérales.

Le tableau II.2 présent des exemples de valeurs des micro-coefficients. Cas d'une surface A : un béton sec et surface B : asphalte mouillée et ses interprétations [22] :

Tableau II.2 Quelques exemples de valeurs de micro coefficient de la formule magique.

Coefficients de Force de traction Longitudinale			
Surface	A	B	Commentaire
Coeff.	Valeurs		
a_0	1.65	1.65	facteur de forme de force longitudinale
a_1	-43.707	-42.026	Frottement longitudinal
a_2	1329.305	1280.406	Variation de frottement avec la charge
a_3	1285.465	1378.389	Rigidité de glissement longitudinal
a_4	8.875	8.847	Variation de rigidité avec la charge
a_5	0.017	0.016	Exposant dans la rigidité de glissement
a_6	-0.021	-0.025	Variation de courbure avec le carré de charge
a_7	-0.513	-0.428	Variation de courbure avec la charge
a_8	-0.111	-0.132	Courbure longitudinale

Coefficients de Force de traction Latérale			
Surface	A	B	Commentaire
Coeff.	Valeurs		
b_0	1.4	1.4	facteur de forme de force latérale
b_1	-33.015	-21.3	Frottement latéral
b_2	1153.2	744.0	Variation de frottement avec la charge
b_3	113.398	49.6	Rigidité de glissement latéral
b_4	516.693	226.0	Variation de rigidité avec la charge
b_5	0.3	0.3	Exposant dans la rigidité de glissement
b_6	-0.006	-0.006	Variation de courbure avec le carré de charge
b_7	-0.056	-0.056	Variation de courbure avec la charge
b_8	-0.486	-0.486	Courbure latéral

Note : les seuls coefficients qui ont un changement considérable selon le changement de la surface sont a_1 a_2 a_3 pour les forces longitudinales et b_1 b_2 b_3 pour les forces latérales. Parce qu'ils sont liés directement aux frottements longitudinaux et latéraux de la surface.

Dans ce travail, on a utilisé des coefficients de frottement longitudinaux et latéraux μ_{long} , μ_{lat} extraits des essais qui ont été fait par **Shahrul Naim Sidek** sur le Pioneer 3DX [21].

Tableau II.3 Coefficeients de frottement

Coefficient de friction	Surface non glissante	Surface glissante
μ_{long}	0.6107	0.2811
μ_{lat}	0.3856	0.2283

Dans ce chapitre, nous avons discuté de l'interaction pneu-chaussée et nous avons utilisé la formule magique pour modéliser la relation entre le glissement et la traction. L'interaction pneu-chaussée est essentielle et doit être prise avec précaution. Car les erreurs dans le calcul des forces de traction aura un effet dramatique sur l'exactitude de non seulement le modèle du robot, mais bien la stratégie de contrôle aussi. Pour cette raison, on doit noter que la détermination correcte des micro-coefficients de la formule magique pour le robot 3DX nécessite des expérimentations sur le robot lui-même.

Après qu'on a élaboré tout ce qui est lié à la modélisation et l'interaction roues-chaussée d'un robot mobile à roues soumis à un glissement, nous allons aborder la stratégie de la commande dans le chapitre suivant.

| CHAPITRE III : LA COMMANDE

III-1. Les taches de mouvement de base

III-2. Commande de suivi non-linaire

III-3. Commande adaptative neuronale

III-1.INTRODUCTION :

La navigation des robots mobiles peut se composer en trois problèmes : le suivi de chemin, la poursuite de trajectoire, la stabilisation de configuration fixe.

III-1-1. LES TACHES DE MOUVEMENT DE BASE :

Tout d'abord, nous devons rappeler que le mouvement d'un robot dans un plan cartésien est décrit par deux repères : le repère local lié au point référence du robot et le repère monde. Le robot est représenté par le vecteur $q = [x, y, \varphi]^T$ qui est la pose du robot dans le repère monde. Les taches de base dans la robotique mobile sont [49]:

III-1-1-1. SUIVI DE CHEMIN :

Le robot mobile doit suivre une courbe de référence C dans le plan cartésien à partir d'une configuration initial qui appartenant ou pas à la courbe à suivre. La vitesse de suivi dans ce cas est constante, fixée tout au début, et reste indépendante du chemin. La régulation se fait à travers la vitesse angulaire et la distance entre le robot et la courbe.

Le but dans ce problème est de trouver une loi de commande en vitesse:

$$v_c(t) = f_c(e_\varphi, v, b, K)$$

Tel que $\lim_{t \rightarrow \infty} e_\varphi = 0$ et $\lim_{t \rightarrow \infty} b(t) = 0$ tels que : e_φ et $b(t)$ sont l'erreur d'orientation φ et la distance entre le point référence du robot et la courbe C respectivement.

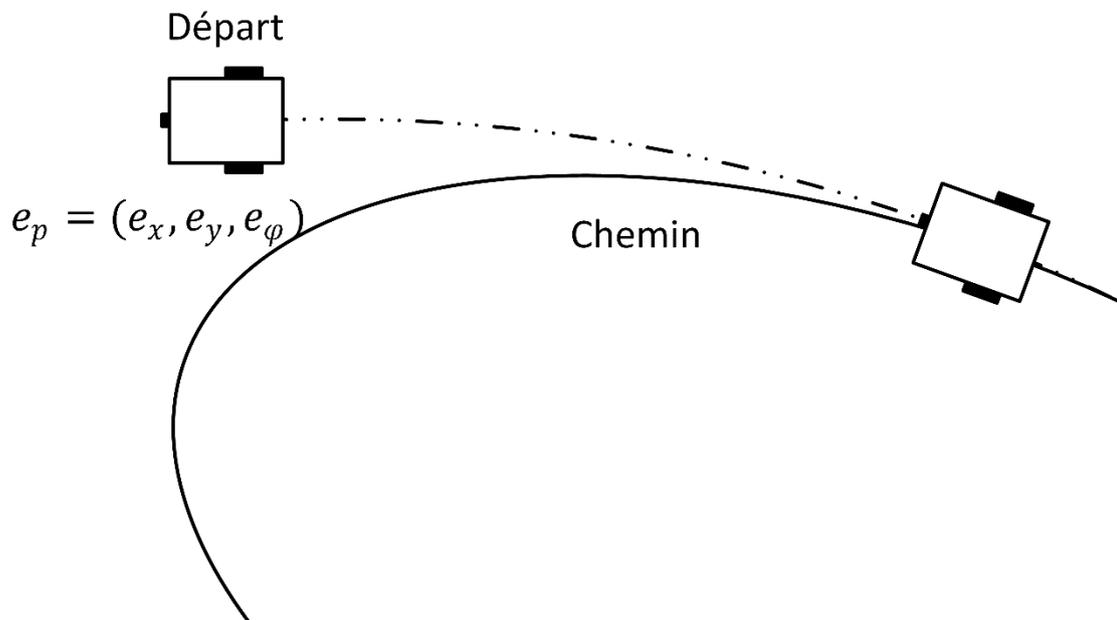


Figure III.1 Suivi de Chemin

III-1-1-2. POURSUITE DE TRAJECTOIRE :

Ce problème se distingue du suivi de chemin par le fait que la vitesse de déplacement n'est plus fixée à l'avance. Du fait que la trajectoire est un chemin plus une loi horaire associée. C'est-à-dire à chaque instant t on a une configuration qui doit être atteinte.

La trajectoire doit se conformer avec les contraintes cinématiques du robot. Ce qui rend la tâche une poursuite d'un robot non-holonome virtuel possédant la pose q_d et la vitesse V_d :

$$q_d = \begin{bmatrix} x_d \\ y_d \\ \varphi_d \end{bmatrix} \quad \text{et} \quad V_d = \begin{bmatrix} v_d \\ \omega_d \end{bmatrix}$$

Tels que

$$\dot{x}_d = v_d \cos(\varphi_d)$$

$$\dot{y}_d = v_d \sin(\varphi_d)$$

$$\dot{\varphi}_d = \omega_d$$

v_d et ω_d sont la vitesse linéaire et la vitesse angulaire du robot virtuel, avec $v_d > 0$ pour tout t . Le but est de trouver une loi de commande régulière en vitesse :

$$V_c(t) = f_c(e_p, V_d, K)$$

Tel que $\lim_{t \rightarrow \infty} (q_d - q) = 0$ tels que e_p, V_d, K sont l'erreur du suivi, le vecteur de vitesse de référence et le vecteur gain de la commande respectivement.

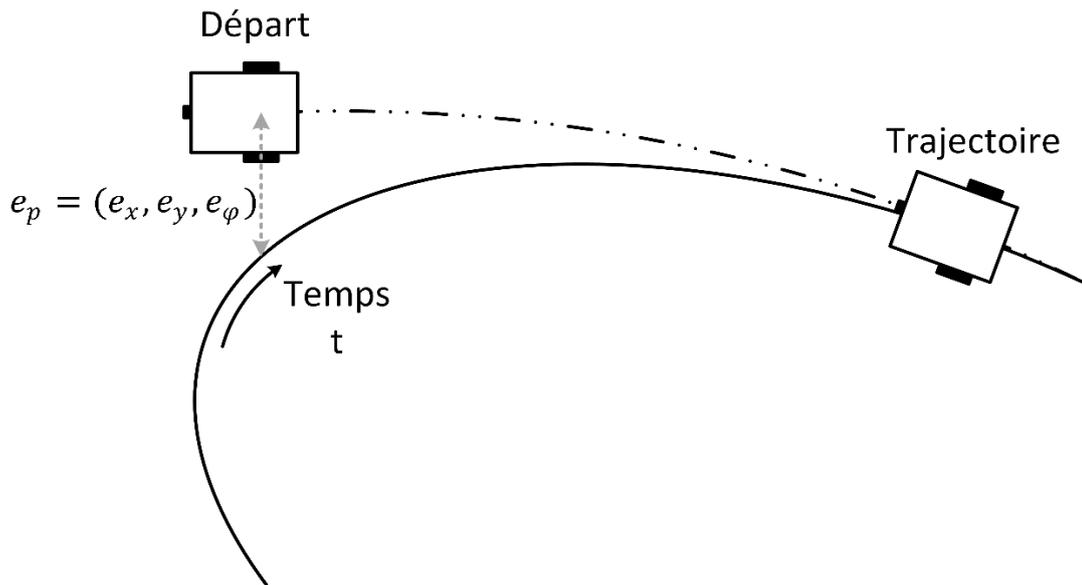


Figure III.2 Poursuite de Trajectoire

III-1-1-3. STABILISATION DE CONFIGURATION FIXE :

La stabilisation de configuration fixe autrement dit commande point à point. Étant donné une configuration arbitraire q_d comme référence, l'objectif est de trouver une loi de commande régulière de vitesse variante dans le temps :

$$V_c(t) = f_c(e_p, V_d, K, t)$$

Tel que $\lim_{t \rightarrow \infty} (q_d - q) = 0$.

Ce problème est celui pour lequel les propriétés de non-holonomie interviennent de la façon la plus contraignante, et pour lequel les méthodes classiques de l'automatique linéaire se montrent insuffisantes. Les applications principales de ce problème sont liées aux manœuvres de parking (créneau), et aux applications nécessitant une immobilisation du véhicule s'accompagnant d'un positionnement précis.

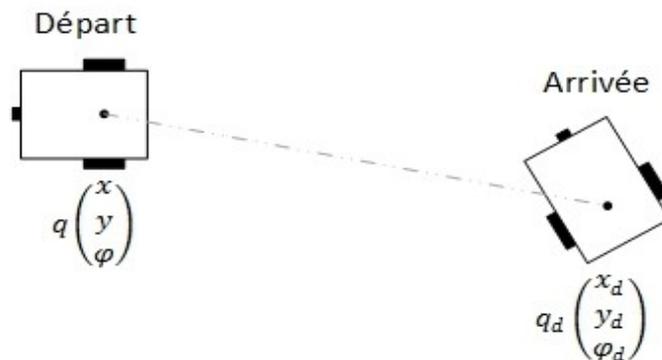


Figure III.3 Stabilisation autour d'une posture

Dans ce chapitre nous allons étudier plus avant le problème de la poursuite de trajectoire d'un robot mobile à roues non-holonome.

III-2. COMMANDE DE SUIVI NON-LINAIRE PAR BACKSTEPPING:

Le modèle du robot se compose principalement de deux parties : la cinématique et la dynamique. Les couples moteurs des roues droite et gauche sont les entrées de la partie dynamique. Ensuite les sorties seront utilisées comme des entrées de la partie cinématique, comme montre la figure Fig. III.4.

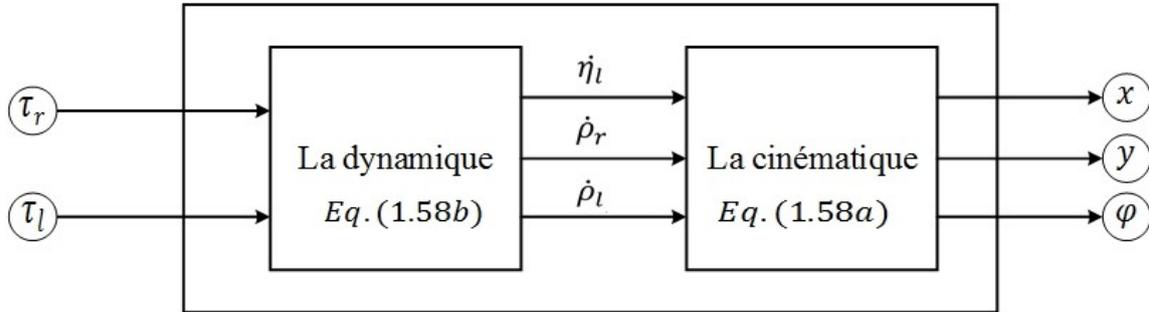


Figure III.4 Bloc-diagramme du robot

Beaucoup de chercheurs ont tendance de négliger la partie dynamique en supposant la poursuite parfaite « Perfect Velocity Tracking ». Cela rend la commande une commande de guidage « Steering control ». En réalité le « Perfect Velocity Tracking » ne se vérifie pas toujours. Donc c'est pratique d'utiliser un contrôleur cinématique dont les sorties seront utilisées comme des entrées références d'un autre contrôleur dynamique [20] Fig. III.5. La procédure de commande appartient à la classe générale des commandes « Backstepping » se déroule donc en deux étapes :

- Commande de stabilisation cinématique.
- Commande de stabilisation dynamique.

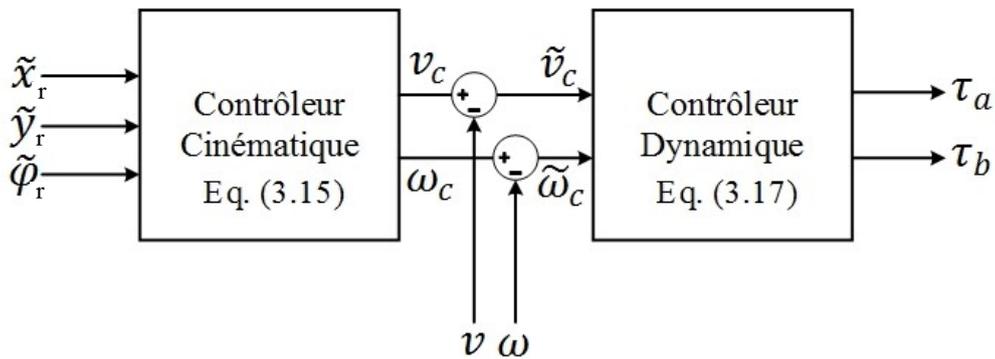


Figure III.5 Bloc-diagramme de la commande

Les erreurs présentées dans la figure Fig. III.5 peuvent être regroupées en deux groupes: les erreurs liées à la cinématique (fondamentalement sont les erreurs de position $\tilde{x}_r, \tilde{y}_r, \tilde{\varphi}_r$) et les erreurs de vitesse liées directement à la dynamique $\tilde{v}_c, \tilde{\omega}_c$.

La commande de stabilisation cinématique a l'objectif de minimiser les erreurs de position dans une boucle fermée globale. Alors que la commande de stabilisation dynamique a deux fonctions. Premièrement elle minimise les erreurs de vitesse dans une boucle fermée locale, comme il est indiqué dans la figure Fig. III.6, où la dynamique du robot est considérée comme un sous-système. Deuxièmement elle calcule les couples moteurs τ_a, τ_b qui agissent sur le robot mobile et assurent les corrections de position et de vitesses.

La stabilité est assurée pour chaque boucle fermée de manière individuelle. Mais si la stabilité de toutes les boucles est assurée cela signifie que le système est globalement stable.

Comme les entrées du modèle du robot sont les couples moteurs, et la commande est une commande de vitesse nous avons besoins d'obtenir les couples à partir des vitesses de commande. Cela se fait à l'aide des équations Eq. (1.34) avec :

$$F = F_r + F_l \quad (3.1a)$$

$$N = (F_r - F_l)2a \quad (3.1b)$$

Et

$$\tau_r = rF_r \quad (3.2a)$$

$$\tau_l = rF_l \quad (3.2b)$$

On obtient

$$\dot{v} = \frac{1}{mr} (\tau_r + \tau_l) \quad (3.3a)$$

$$\dot{\omega} = \frac{2a}{lr} (\tau_r - \tau_l) \quad (3.3b)$$

Alors

$$\dot{v} = \frac{1}{m} \tau_a \quad (3.3c)$$

$$\dot{\omega} = \frac{1}{l} \tau_b \quad (3.3d)$$

Où

$$\tau_a = \frac{1}{r}(\tau_r + \tau_l) \quad (3.4a)$$

$$\tau_b = \frac{2a}{r}(\tau_r - \tau_l) \quad (3.4b)$$

Donc

$$\tau_r = \frac{(2a\tau_a + \tau_b)r}{4a} \quad (3.5a)$$

$$\tau_l = \frac{(2a\tau_a - \tau_b)r}{4a} \quad (3.5b)$$

Tels que τ_r, τ_l sont les couples moteurs des roues droite et gauche respectivement. τ_a, τ_b sont les couples dérivés des vitesses de commande qui agissent sur le robot mobile. a, r sont des paramètres dimensionnels. m, I sont la masse et le moment d'inertie du robot respectivement.

Nous savons aussi de la cinématique que la vitesse du point de référence (G : centre de masse) par rapport aux deux axes du repère monde est :

$$\dot{x} = v \cos(\varphi) \quad (3.6a)$$

$$\dot{y} = v \sin(\varphi) \quad (3.6b)$$

Dans cette commande nous cherchons à trouver les vitesses de commande qui nous donnent les couples τ_a, τ_b appropriés qui nous amènent à la poursuite parfaite de la trajectoire.

III-2-1. CONTROLEUR DE POURSUITE CINEMATIQUE NON-LINAIRE :

Nous prenons q comme vecteur d'état, le problème est la poursuite d'une trajectoire d'état :

$$q_d = \begin{bmatrix} x_d(t) \\ y_d(t) \\ \varphi_d(t) \end{bmatrix} \quad (3.7)$$

Avec une erreur qui tend vers zéro asymptotiquement. D'une manière réalisable la trajectoire désirée doit satisfaire la cinématique aussi bien les contraintes non-holonomes :

$$\begin{bmatrix} \dot{x}_d \\ \dot{y}_d \\ \dot{\varphi}_d \end{bmatrix} = \begin{bmatrix} v_d \cos(\varphi_d) \\ v_d \sin(\varphi_d) \\ \omega_d \end{bmatrix}, \quad \dot{x}_d \sin(\varphi_d) = \dot{y}_d \cos(\varphi_d) \quad (3.8)$$

Les erreurs de poursuite exprimés dans le repère monde sont définis comme [50] :

$$\tilde{x} = (x_d - x) \quad (3.9a)$$

$$\tilde{y} = (y_d - y) \quad (3.9b)$$

$$\tilde{\varphi} = (\varphi_d - \varphi) \quad (3.9c)$$

On les transfère au repère local à l'aide de l'inverse de la matrice de rotation orthogonal Eq. (1.3), et on trouve les erreurs de poursuite dans le repère local du robot:

$$e_p = \begin{bmatrix} \tilde{x}_r \\ \tilde{y}_r \\ \tilde{\varphi}_r \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) & 0 \\ -\sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{\varphi} \end{bmatrix} \quad (3.10)$$

De Eq. (3.10) et Eq. (3.8) on obtient le model cinématique des erreurs $[\tilde{x}_r \quad \tilde{y}_r \quad \tilde{\varphi}_r]^T$ suivant [50] :

$$\dot{\tilde{x}}_r = v_d \cos(\tilde{\varphi}_r) - v + \tilde{y}_r \omega \quad (3.11a)$$

$$\dot{\tilde{y}}_r = v_d \sin(\tilde{\varphi}_r) - \tilde{x}_r \omega \quad (3.11b)$$

$$\dot{\tilde{\varphi}}_r = \omega_d - \omega \quad (3.11c)$$

Et

$$\dot{e}_p = \begin{bmatrix} \dot{\tilde{x}}_r \\ \dot{\tilde{y}}_r \\ \dot{\tilde{\varphi}}_r \end{bmatrix} \quad (3.12)$$

Le bon choix des vitesses v, ω assure que $e_p = 0$ soit un équilibre stable de la dynamique des erreurs Eq. (3.11). Les vitesses v, ω sont les lois de la commande cinématique. On peut les écrire v_c, ω_c en supposant que l'atteignabilité de ces vitesses est assurée via le contrôleur dynamique.

En se basant sur la méthode de Lyapunov pour choisir la loi de commande v_c, ω_c qui assure la stabilité de la commande cinématique.

METHODE DE LYAPUNOV :

Soit $e_p = 0 \in D \subset \mathbb{R}^n$ un point d'équilibre pour la dynamique $\dot{e}_p = f(e_p)$. Soit $V: D \rightarrow \mathbb{R}$ une fonction continue différentiable dite fonction candidate de Lyapunov telle que :

$$i) V(0) = 0 \text{ et } V(e_p) > 0, \quad \forall e_p \in D \setminus \{0\}$$

$$ii) \dot{V}(e_p) \leq 0, \quad \forall e_p \in D$$

Alors, $e_p = 0$ est stable, en outre si

$$iii) \dot{V}(e_p) < 0, \quad \forall e_p \in D \setminus \{0\}$$

$e_p = 0$ est asymptotiquement stable.

La fonction candidate de Lyapunov est choisie comme [20] :

$$V(e_p) = \frac{1}{2}(\tilde{x}_r^2 + \tilde{y}_r^2) + \left(\frac{1}{K_y}\right)(1 - \cos(\tilde{\varphi}_r)) \quad (3.13)$$

Tels que $\tilde{x}_r, \tilde{y}_r, \tilde{\varphi}_r$ sont les erreurs de la poursuite.

Cette fonction est continue différentiable et satisfait la condition i). Donc on doit vérifier sous quelle condition, ii) et iii) peuvent être satisfaits. En dérivant Eq. (3.13) par rapport au temps on trouve :

$$\dot{V}(e_p) = (-v + v_d \cos(\tilde{\varphi}_r)) \tilde{x}_r + \left(-\frac{\omega}{K_y} + v_d \tilde{y}_r + \frac{\omega_d}{K_y}\right) \sin(\tilde{\varphi}_r) \quad (3.14)$$

Si on prend v_c et ω_c tel que $\dot{V}(e_p) \leq 0$:

$$v_c = K_x \tilde{x}_r + v_d \cos(\tilde{\varphi}_r) \quad (3.15a)$$

$$\omega_c = K_\varphi \sin(\tilde{\varphi}_r) + K_y v_d \tilde{y}_r + \omega_d \quad (3.15b)$$

Eq. (3.14) devient :

$$\dot{V}(e_p) = -(K_x \tilde{x}_r^2 + \frac{K_\varphi}{K_y} \sin^2(\tilde{\varphi}_r)) \quad (3.16)$$

Tels que K_x, K_y et K_φ sont des gains de commande. Il est claire que $\dot{V}(e_p) \leq 0$ pour $K_x > 0$, $K_y > 0$ et $K_\varphi > 0$. et $\dot{V}(e_p) = 0$ pour $\tilde{x}_r = 0$ et $\tilde{\varphi}_r = 0$. Donc le contrôleur cinématique présenté par les équations Eq. (3.15) garantit la stabilité asymptotique totale de la commande cinématique.

III-2-2. CONTROLEUR DE POURSUIT DYNAMIQUE:

Après avoir choisi la loi de commande v_c et ω_c comme dans les équations Eq. 3.15, on doit assurer que le robot atteigne ces vitesses. Cela se fait par le bon choix des entrées des couples τ_a et τ_b en se basant sur la dynamique du robot mobile Eq. (3.3). Si on prend les couples comme suit:

$$\tau_a = m\dot{v}_c + K_a \tilde{v}_c \quad (3.17a)$$

$$\tau_b = I\dot{\omega}_c + K_b \tilde{\omega}_c \quad (3.17b)$$

Où les erreurs de vitesses sont :

$$\tilde{v}_c = v_c - v \quad (3.18a)$$

$$\tilde{\omega}_c = \omega_c - \omega \quad (3.18b)$$

Tels que v et ω sont les vitesses du robot, v_c, ω_c sont considérées comme référence.

En introduisant Eq 3.18 dans Eq 3.17 on obtient les équations des erreurs [20] :

$$\dot{\tilde{v}}_c + \left(\frac{K_a}{m}\right) \tilde{v}_c = 0 \quad (3.19a)$$

$$\dot{\tilde{\omega}}_c + \left(\frac{K_b}{I}\right) \tilde{\omega}_c = 0 \quad (3.19b)$$

Ansı pour $K_a > 0$ et $K_b > 0$, $\tilde{v}_c, \tilde{\omega}_c$ convergent asymptotiquement vers zéro.

En conséquence le choix d'une loi de commande par retour d'état comme dans Eq. (3.17) avec v_c et ω_c sont données dans Eq. (3.15) assure la stabilité du système et la convergence d'états du robot mobile vers les états désirée $[x_d(t) \ y_d(t) \ \varphi_d(t)]^T$.

La figure Fig.III.6 présent le schéma-bloc de la commande « backstepping » de poursuite étudiée :

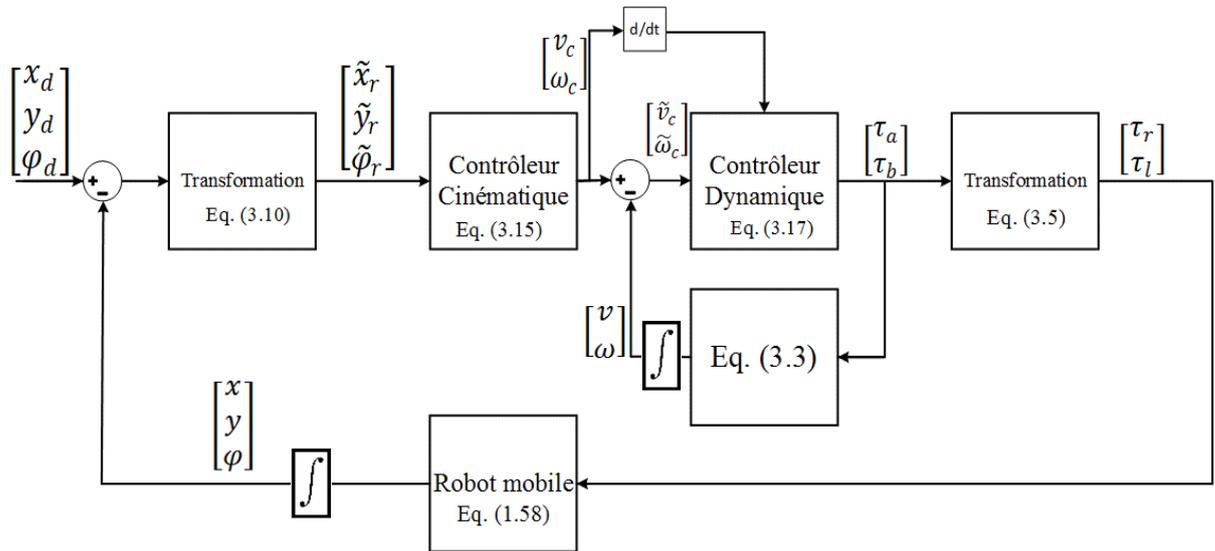


Figure III.6 Bloc-diagramme de la commande backstepping

Cette commande dépend d'une connaissance complète de la modélisation dynamique (m : la masse, I : moment d'inertie du robot). Seulement, en pratique le système est confronté à la présence des dynamiques et des perturbations inconnues. Ce qui conduit à une modélisation incomplète. Cependant, ce problème peut être résolu par un paradigme capable d'approximer ces inconnues dans le modèle dynamique.

III-3. COMMANDE ADAPTATIVE NEURONALE :

Grâce à leur propriété d'approximation universelle, les réseaux de neurones ont été utilisés dans des nombreuses applications, dans l'identification, la planification et la commande des robots. Ils sont capables d'approximer n'importe quelle fonction non linéaire avec n'importe quelle précision souhaitée. Pour cela l'introduction d'un réseau de neurones dans la partie dynamique de la commande non linéaire « Backstepping », peut l'améliorer et résoudre ces inconvénients, concernant les perturbations et les dynamiques inconnues. Avant de passer à la conception d'une telle commande nous devons donner une présentation préliminaire sur les réseaux de neurones.

III-3-1. LES RESEAUX DE NEURONES ARTIFICIELS :

Le réseau de neurones artificiels est une collection de simples processeurs mathématiques reliés ensemble. Chaque processeur ne peut effectuer qu'une seule tâche mathématique primitive. Mais un réseau complexe, possède de grandes capacités et peut faire beaucoup de choses.

L'inspiration derrière les réseaux de neurones artificiels est le cerveau. Il est constitué d'environ 100 milliards d'unités de traitement reliés entre eux de manière à former un réseau. Ces unités processeurs sont appelées « cellules cérébrales » ou « Neurones », chacune d'elle est une cellule vivante.

Le concept des réseaux de neurones artificiels a été défini par Mc Culloch et Pitts en 1943, à la suite de leurs travaux sur les cellules cérébrales humaines qu'ils ont nommés « Neurons ». Le pas suivant a été fait en 1949 par Hebb, qui a défini le concept des « poids synaptiques » [20].

Pour comprendre le fonctionnement du réseau de neurones artificiels, il est préférable de présenter d'abord le fonctionnement des réseaux de neurones biologiques.

III-3-1-1. LE NEURONE BIOLOGIQUE :

Les neurones ou les cellules cérébrales sont les éléments constitutifs du système nerveux. Bien qu'ils aient la même organisation générale et l'équipement biochimique des autres cellules, ils possèdent des caractéristiques uniques. Ils ont une forme particulière, une membrane extérieure capable de générer des impulsions électriques, et une structure unique : la synapse pour transférer les informations d'un neurone à un autre. Il est possible de distinguer trois régions de cette cellule spécialisée [51]:

- Le corps cellulaire.
- Les dendrites.
- L'axone.

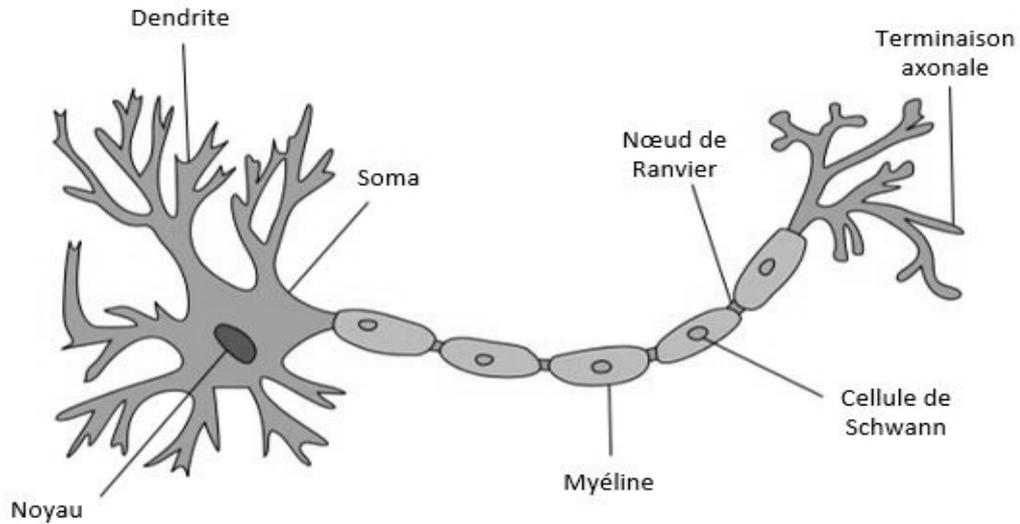


Figure III.7 Le neurone biologique

Le corps cellulaire ou soma assure les fonctions de soutien et de structure de la cellule. Il recueille et traite les informations provenant d'autres neurones. L'axone s'étend loin du corps cellulaire et assure le chemin sur lequel les informations voyagent. Les dendrites sont des extensions de forme tubes et se ramifient à maintes reprises pour former un arbre touffu autour du soma. Ils assurent les chemins principaux sur lesquels les neurones reçoivent les informations. Le point de jonction entre l'axone et une dendrite d'un autre neurone s'appelle une synapse, là où les informations sont transportées d'un neurone à un autre au moyen des transmetteurs chimiques qui sont libérés par l'arrivée d'une impulsion nerveuse.

En générale, on peut soutenir que le rôle de neurones dans les processus de raisonnements cérébraux est analogue au rôle des commutateurs logiques dans un ordinateur numérique. Toutefois cette analogie est très simple, un neurone contient un seuil de sensibilité, une amplification ajustable des signaux dans chaque synapse, et une structure interne qui permet l'intégration des signaux nerveux entrants dans les deux, l'espace et le temps.

Du point de vue mathématique, il peut être conclu que le traitement des informations dans un neurone implique les deux opérations mathématiques suivantes [51] :

- *Opération Synaptique*: La vigueur (le poids) d'une synapse est une représentation de stockage de connaissance ainsi que la mémoire de la connaissance précédente. L'opération synaptique attribue un poids relatif (importance) pour chaque signal entrant selon l'expérience passée (connaissance) stockée dans la synapse.
- *Opération Somatique*: L'opération somatique assure diverses opérations mathématiques telles que l'agrégation, seuillage, activation non linéaire et un traitement dynamique des

entrées synaptiques. Si l'agrégation pondérée des entrées de neurones dépasse certain seuil, le soma va générer un signal de sortie à son axone.

Voici la représentation simplifiée de ces opérations neuronales :

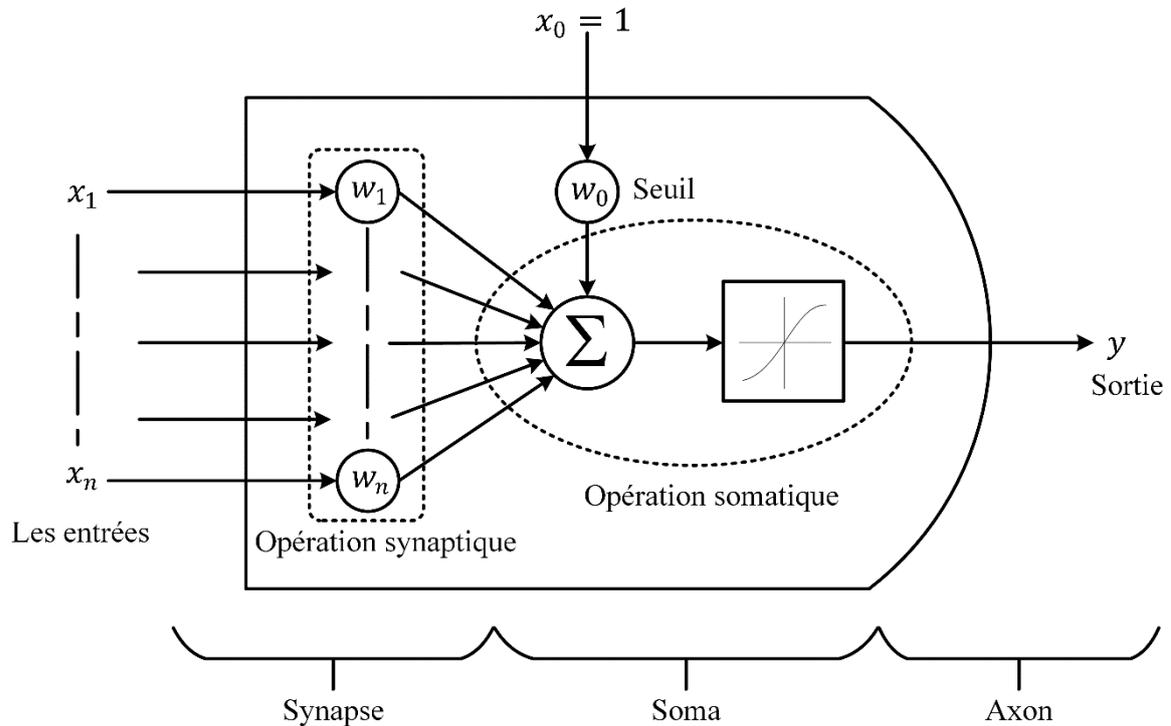


Figure III.8 Diagramme opérationnel d'un neurone biologique

III-1-1-2. LE NEURONE ARTIFICIEL :

Inspirés par le traitement mathématique dans les cellules cérébrales, Mc Culloch-Pitts ont proposés le premier modèle d'un neurone sous forme d'une unité logique à seuil, qui a un processus de base de trois éléments [20] :

- Un ensemble des connexions entrées (synapses).
- Un nœud de sommation linéaire.
- Une fonction d'activation non linéaire.

Chaque branchement de connexion a un poids associé (force), qui est positif s'il a un rôle excitateur ou négative s'il a un rôle inhibiteur. Le nœud de sommation additionne les signaux d'entrée multipliés par leurs poids synaptiques respectifs. Finalement, la fonction d'activation limite l'amplitude admissible du signal de sortie à une valeur finie généralement située dans l'intervalle normalisé $[0, 1]$ ou bien $[-1, 1]$. Le modèle de neurone a également un seuil θ qui est appliqué extérieurement, pratiquement il abaisse le net (somme) des entrées pour la fonction d'activation. La figure Fig. III.9 montre le neurone de base artificiel :

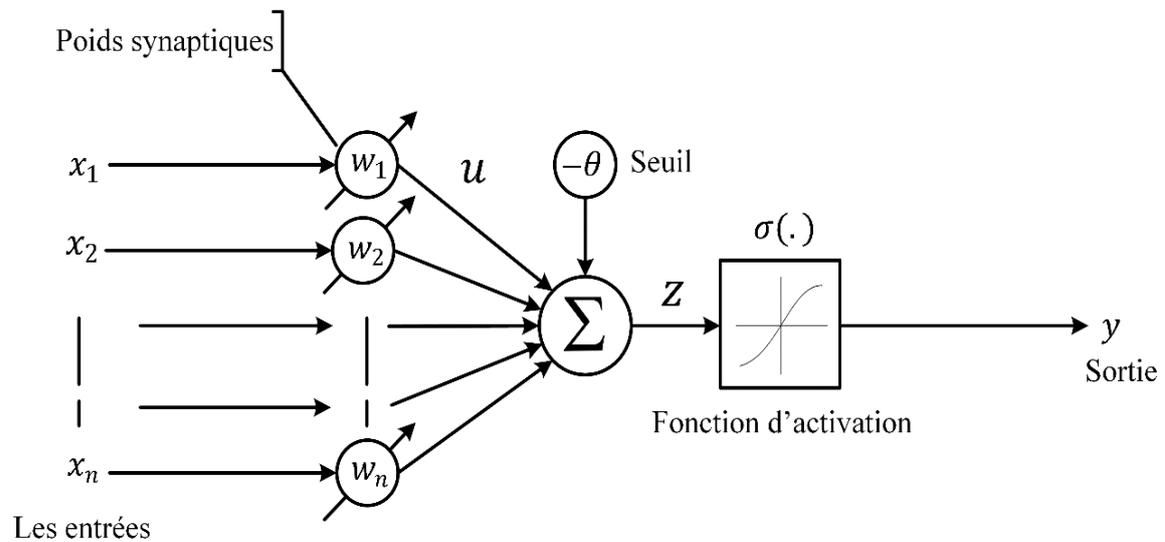


Figure III.9 Le neurone artificiel

De la Fig. III.9 il est clair que le neurone est décrit par les équations suivantes :

$$u = \sum_{i=1}^n w_i x_i \quad (3.20a)$$

$$y = \sigma(z) \quad (3.20b)$$

$$z = u - \theta, \quad \theta > 0 \quad (3.20c)$$

Si le seuil est $-\theta$, il est considéré comme une entrée normale $x_0 = -1$ avec un poids correspondant $w_0 = \theta$, alors le modèle du neurone prend la forme :

$$y = \sigma(z) \quad (3.21a)$$

$$z = \sum_{i=0}^n w_i x_i \quad (3.21b)$$

La fonction d'activation non linéaire $\sigma(x)$ peut-être de type « tout ou rien », ou une fonction de type saturation, ou une fonction sigmoïde [56] avec des valeurs soit dans l'intervalle $[0, 1]$ ou bien $[-1, 1]$ comme il est montré dans les Fig. III.10-11 .

Les deux fonctions sigmoïdes les plus utilisées sont :

- La fonction logistique:

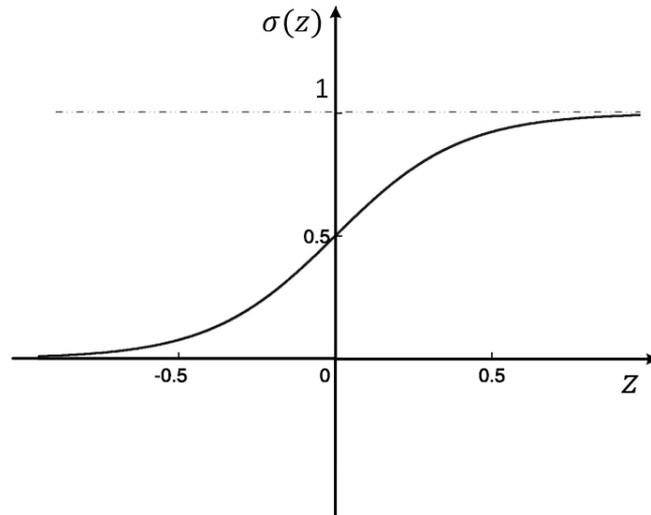


Figure III.10 Fonction sigmoïde logistique

$$y = \sigma(z) = \frac{1}{1 + e^{-z}} , \quad y \in [0,1] \quad (3.22a)$$

- La fonction hyperbolique tangentielle :

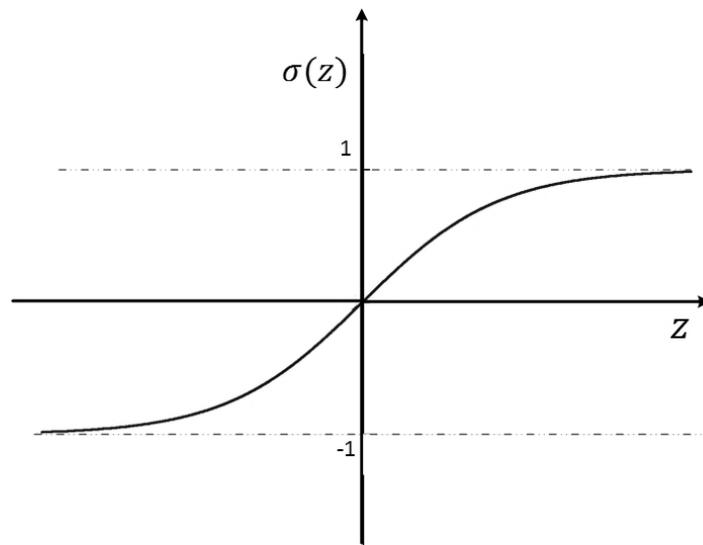


Figure III.11 Fonction hyperbolique tangentielle

$$y = \sigma(z) = \tan\left(\frac{z}{2}\right) = \frac{1 - e^{-z}}{1 + e^{-z}} , \quad y \in [-1,1] \quad (3.22b)$$

III-1-2. LE PERCEPTRON MULTICOUCHES :

Le réseau de neurones MLP a été développé par Rosembat en 1958. Il s'agit de trois couches interconnectées de nœuds, chaque nœud est un neurone artificiel. Ces couches sont : une couche d'entrée, une couche de sortie et une couche intermédiaire appelée couche cachée. La figure Fig. III.12 présent la structure de base d'un MLP :

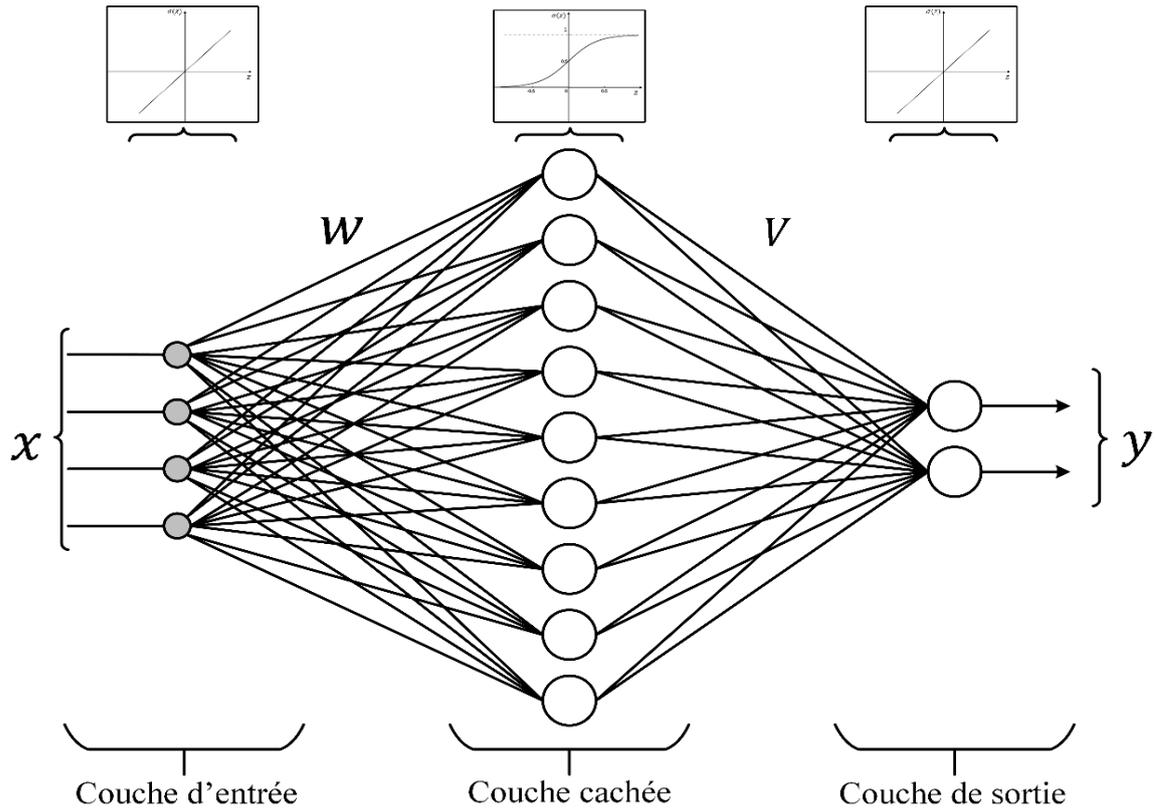


Figure III.12 Structure d'un réseau de neurones MLP

C'est possible d'utiliser plusieurs couches cachées et cela selon les besoins de précision. Mais Kolmogorov dans son théorème d'approximation universel [52] a indiqué qu'une seule couche de neurones cachée est suffisante pour que le MLP accompli les opérations, qui peuvent être réalisées avec plusieurs couches cachées [20].

La sortie d'un réseau de neurones MLP comme le montre la Fig. III.12 est donnée par la relation :

$$y_i = \sum_{j=1}^L \left[v_{ij} \sigma \left(\sum_{k=0}^n w_{jk} x_k \right) \right], i = 1, 2, \dots, m \quad (3.23)$$

Où $x_k (k = 0, 1, 2, \dots, n)$ sont les entées présenté au réseau de neurones y compris les seuils. w_{jk} sont les poids associés aux interconnexions entre la couche d'entrée et la couche cachée. v_{ij} sont les poids associés aux interconnexions entre la couche cachée et la couche de sortie. L représente le nombre de neurones dans la couche cachée. m représente le nombre de neurones dans la couche de sortie.

Sous une forme compacte l'équation Eq. (3.23) peut être écrite comme suit :

$$y = V^T \sigma(W^T x) \quad (3.24)$$

Où

$$x = [x_0, x_1, \dots, x_n]^T, \quad y = [y_1, y_2, \dots, y_m]^T$$

Et

$$V^T = \begin{bmatrix} V_1^T \\ V_2^T \\ \vdots \\ V_m^T \end{bmatrix}, \quad \sigma(W^T X) = \begin{bmatrix} \sigma(W_1^T X) \\ \sigma(W_2^T X) \\ \vdots \\ \sigma(W_L^T X) \end{bmatrix}$$

$$V_i^T = [v_{i1} \quad v_{i2} \quad \dots \quad v_{iL}], \quad W_i^T = [w_{i0} \quad w_{i1} \quad \dots \quad w_{in}]$$

III-1-3. APPRENTISSAGE PAR BACKPROPAGATION :

L'algorithme de retro-propagation (Back-propagation) est un algorithme d'apprentissage supervisé. Il met à jour les poids synaptiques telle que l'erreur entre les sorties désirées et réelles soit minimisée. Les erreurs sont retro-propagées via les couches du réseau afin de calculer les corrections de tous les poids du réseau.

Les poids seront mis à jour de manière à minimiser le critère suivant :

$$E_p(t) = \frac{1}{2} \sum_k e_k^2(t) \tag{3.25a}$$

$$e_k(t) = d_k(t) - y_k(t) \tag{3.25b}$$

Où :

$d_k(t)$ est la réponse désirée de la $k^{\text{ème}}$ neurones de sortie a l'instant t .

$y_k(t)$ est la réponse réelle obtenue après la présentation de chaque vecteur d'entrée x à la couche des entrées à l'instant t .

Cela signifie que la minimisation de $E_p(t)$ doit se faire successivement "pattern" par "pattern".

Où $E_p(t)$ et $y_k(t)$ et $e_k(t)$ sont données par :

$$y_k(t) = \sigma(z_k(t)), \quad e_k(t) = d_k(t) - \sigma_k(z_k(t)), \quad z_k = \sum_{i=0}^n w_{ki}(t)y_i(t)$$

La minimisation de $E_p(t)$ peut se faire par la règle du gradients [20][51] :

$$\Delta w_{ki}(t) = -\gamma \frac{\partial E_p(t)}{\partial w_{ki}(t)} \tag{3.26}$$

Où γ est un paramètre d'apprentissage et :

$$\begin{aligned} \frac{\partial E_p(t)}{\partial w_{ki}(t)} &= \frac{\partial E_p(t)}{\partial e_k(t)} \cdot \frac{\partial e_k(t)}{\partial y_k(t)} \cdot \frac{\partial y_k(t)}{\partial z_k(t)} \cdot \frac{\partial z_k(t)}{\partial w_{ki}(t)} \\ &= e_k(t)(-1) \frac{\partial \sigma(z_k(t))}{\partial z_k(t)} y_i(t) \end{aligned}$$

Pour la fonction logistique $\sigma(\cdot)$:

$$\frac{\partial \sigma(z_k(t))}{\partial z_k(t)} = \frac{\partial}{\partial z_k(t)} \left[\frac{1}{1 + e^{-z_k}} \right] = - \frac{(-e^{-z_k})}{(1 + e^{-z_k})^2} = y_k(t)[1 - y_k(t)] \quad (3.27)$$

Alors

$$\Delta w_{ki}(t) = \gamma [d_k(t) - y_k(t)] y_k(t) [1 - y_k(t)] y_i(t) = \gamma \delta_k(t) y_i(t) \quad (3.28a)$$

Où

$$\delta_k(t) = [d_k(t) - y_k(t)] [1 - y_k(t)] y_k(t) \quad (3.28b)$$

Est le delta de la retro-propagation avec k le nombre des neurones de sortie et i le nombre des neurones cachés.

La mise à jour des poids de la couche cachée prend la forme suivante :

$$\Delta w_{ji}(t) = \gamma y_i(t) y_j(t) [1 - y_j(t)] \sum_m \delta_m(t) w_{mj}(t) \quad (3.29a)$$

$$\Delta w_{ji}(t) = w_{ji}(t+1) - w_{ji}(t) \quad (3.29b)$$

Où l'index i correspond aux neurones de la couche située après la couche considérée. Pour accélérer la convergence de l'algorithme un terme « Momentum » est ajouté comme suit :

$$w_{ji}(t+1) = \gamma \delta_j(t) y_i(t) + a [w_{ji}(t) - w_{ji}(t-1)] + w_{ji}(t) \quad (3.29c)$$

Avec a est le taux d'influence du momentum qui prend une valeur dans l'intervalle $0 \leq a \leq 1$.

III-1-4. L'APPROXIMATION UNIVERSELLE DES RESEAUX DE NEURONES :

Le réseau de neurones MLP possédant au moins une seule couche cachée est caractérisé par la propriété de l'approximation universelle. Cela découle de théorème de Kolmogorov qui stipule [52] :

Etant donné n'importe quelle fonction continue :

$$F: [0,1]^n \rightarrow F(x) = y, y \in R^m$$

Ou $I = [0,1]^n$ est l'unité cube de dimension n , F peut-être approximée précisément avec un perceptron MLP de trois couches, avec n nœuds (neurones) dans la couche d'entrée, $L = 2n + 1$ nœuds dans la couche cachée et m nœuds dans la couche de sortie.

La fonction non linéaire $f(z)$ utilisée dans les neurones artificiels peut être sélectionnée afin de satisfaire la condition de Lipschitz : $|f(z_1) - f(z_2)| \leq c|z_1 - z_2|^\mu$ pour $0 < \mu < 1$ et c est une constante. N'importe quelle fonction de type sigmoïde ou radiale de base (RBF) satisfaire cette condition.

Dans la pratique, la représentation de $y = F(x)$ est une approximation avec une erreur \mathcal{E} après un nombre d'itération telle que :

$$V^T \sigma(W^T x) = F(x) - \mathcal{E} \quad (3.30)$$

La sélection de nombre des neurones cachés $L \geq 2n + 1$, peut être fait de plusieurs manières de telle sorte que :

$$\|\mathcal{E}\| < \mathcal{E}_0$$

Avec \mathcal{E}_0 est la précision d'approximation.

III-2. COMMANDE NEURONALE :

La commande neuronale se base sur des réseaux de neurones bien définis pour générer les signaux de commande. Les réseaux de neurones sont capables d'apprendre et généraliser à partir des exemples et pour cela ils conviennent à la commande des systèmes non linéaire.

La commande neuronale peut être classée en [53][54] :

- commande neuronale par apprentissage supervisé.
- commande neuronale par apprentissage non supervisé.
- commande neuronale par apprentissage renforcé.

La commande neuronale par apprentissage supervisé est la plus simple Fig. III.13. La fonction du réseau de neurones dans ce cas est d'approximer la loi de commande. Le réseau de neurones se base sur un superviseur pour vérifier la justesse de la loi approximée. Le superviseur peut être un humain ou un contrôleur classique. Après que le réseau de neurones apprend la génération de la loi de commande le superviseur est déconnecté, et le réseau de

neurones joue le rôle d'un contrôleur. Le réseau de neurones recopie le système de commande.

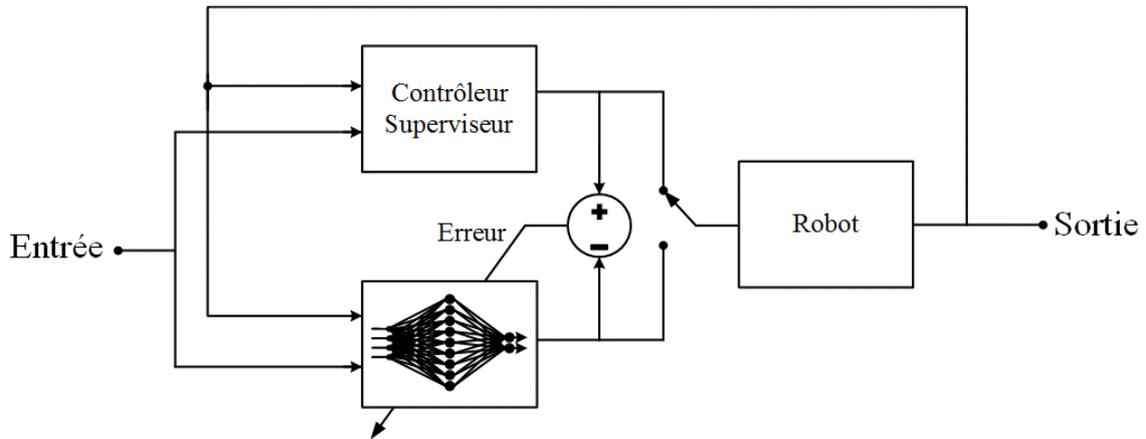


Figure III.13 Commande neuronale par apprentissage supervisé

La commande neuronale la plus répandue est la commande directe par un modèle inverse (direct inverse control) Fig. III.14 dans laquelle le réseau de neurones apprend la dynamique inverse du robot, puis il est utilisé directement comme un contrôleur en boucle ouverte [55].

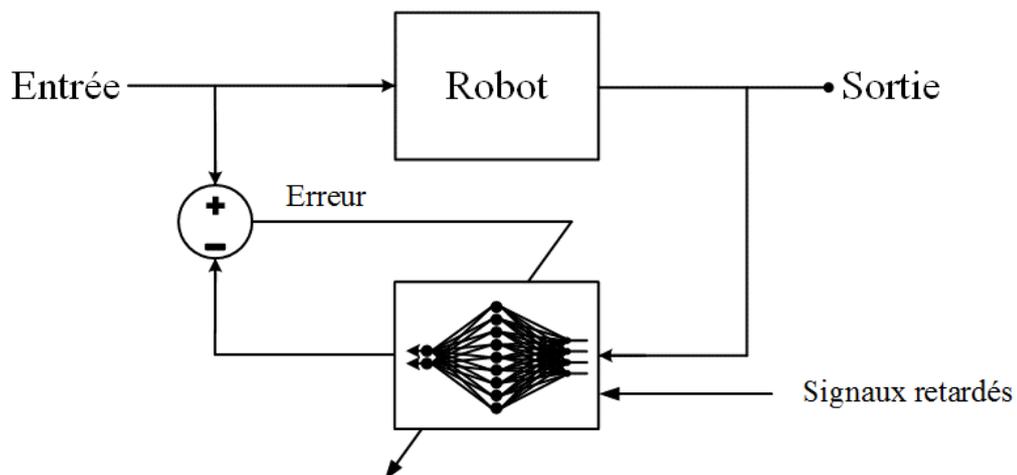


Figure III.14 Apprentissage du modèle inverse

Autre variation est la commande neuronale spécialisée par un modèle inverse Fig III.15. Dans cette commande le réseau de neurones est entraîné « en ligne ». L'erreur de la boucle fermée dans ce cas est le critère d'apprentissage du réseau de neurones. L'apprentissage dans la commande par un modèle inverse direct se fait hors ligne [55].

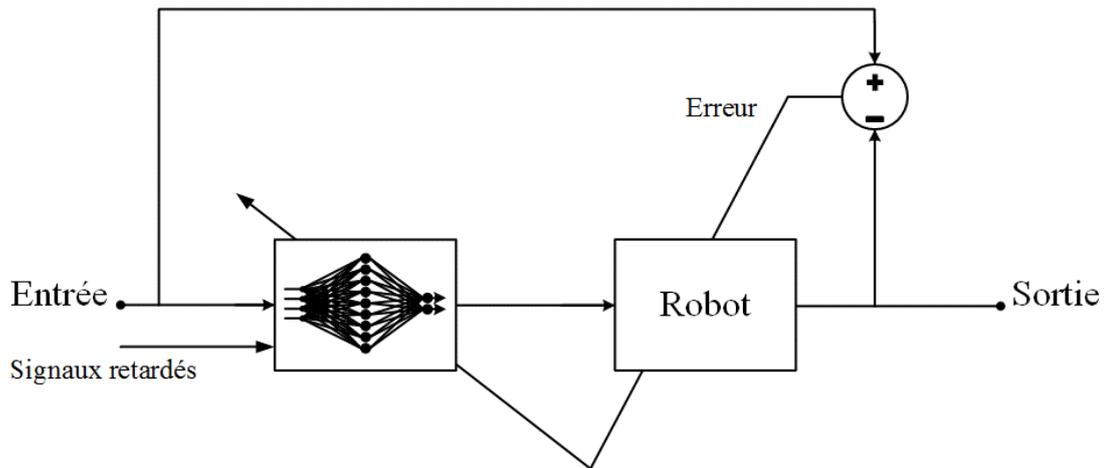


Figure III.15 Commande neuronale spécialisée par un modèle inverse

Autre type des commandes neuronales est la commande neuronale indirect Fig .III.16. Cette commande utilise deux réseaux de neurones : le premier joue le rôle d'un simulateur qui peut être entraîné en ligne où hors ligne, pour apprendre la dynamique du robot. Le deuxième a le rôle d'un contrôleur [55].

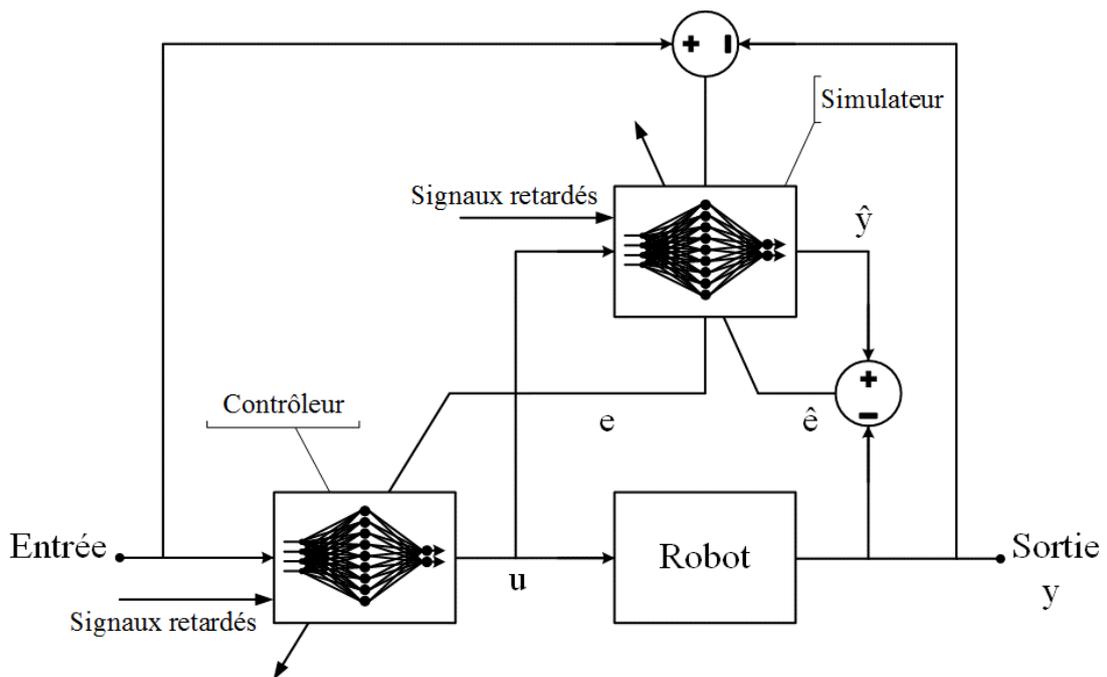


Figure III.16 le neuro-contrôle indirect

Tous ces neuro-contrôleurs utilisent le réseau MLP et peuvent être entraînés avec l'algorithme de retro-propagation [20].

III-2-1. COMMANDE NEURONALE ADAPTATIVE :

La même stratégie de la commande Backstepping sera adoptée ici :

- Un contrôleur cinématique qui génère les vitesses de commande et assure la convergence asymptotique de la trajectoire du robot à la référence.
- Un contrôleur dynamique qui génère les couples agissants sur le robot mobile et qui assurent l'atteignabilité des vitesses générées par la commande cinématique.

Donc de la section (III-2-1) le contrôleur cinématique est de la forme:

$$\begin{aligned} v_c &= K_x \tilde{x}_r + v_d \cos(\tilde{\varphi}_r) \\ \omega_c &= K_\varphi \sin(\tilde{\varphi}_r) + K_y v_d \tilde{y}_r + \omega_d \end{aligned}$$

Pour la dynamique, un contrôleur neuronal est utilisé. Sa fonction est d'approximer tous les inconnus dans la dynamique du robot mobile. Ce qui nous donne un contrôleur dynamique capable de surmonter le problème de glissement [57][58].

Sachant que la dynamique du robot est :

$$\bar{D}(q)\dot{V} + \bar{C}(q, \dot{q})V + \bar{g}(q) = \bar{E}\tau \quad (3.31a)$$

Le terme de gravitation $\bar{g}(q)$ du robot mobile est nul, en ajoutant un terme de frottement linéaire, l'équation Eq. (3.31a) devient :

$$\bar{D}(q)\dot{V} + \bar{C}(q, \dot{q})V + \bar{B}(V) = \bar{\tau} \quad (3.31b)$$

Où $\bar{\tau} = \bar{E}\tau$ et $V = [v, \omega]^T$

Si on prend $V_c = [v_c, \omega_c]^T$ comme vecteur d'entrée de référence du modèle dynamique, l'erreur de poursuite s'écrit:

$$e_v = V_c - V \quad (3.32a)$$

$$e_v = \begin{bmatrix} \tilde{v}_c \\ \tilde{\omega}_c \end{bmatrix} = \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} - \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (3.32b)$$

La dérivée de l'erreur des vitesses est :

$$\dot{e}_v = \dot{V}_c - \dot{V} \quad (3.33)$$

Sachant que

$$\bar{D}(q)\dot{V}_c + \bar{C}(q, \dot{q})V_c + \bar{B}(V) = \bar{\tau}_{commande} \quad (3.34)$$

En soustrayant Eq. (3.31) de Eq. (3.34) on trouve la dynamique des erreurs suivante:

$$\bar{D}(q)\dot{e}_v = -\bar{C}(q, \dot{q})e_v + F(x) - \bar{\tau} \quad (3.35a)$$

Ou :

$$F(x) = \bar{D}(q)\dot{V}_c + \bar{C}(q, \dot{q})V_c + \bar{B}(V) \quad (3.35b)$$

Le vecteur nécessaire pour le calcul de $F(x)$ est :

$$x = \begin{bmatrix} V \\ V_c \\ \dot{V}_c \end{bmatrix}$$

Ce qui est mesurable, et la fonction $F(x)$ contient tous les paramètres dynamiques du robot (les masses, les moments d'inertie, les coefficients de frottement, etc.) qui sont en pratique inconnues ou comprenant des incertitudes.

On pose la loi de commande en boucle fermée comme suit :

$$\bar{\tau} = \hat{F}(x) + K_V e_v - \mu \quad (3.36)$$

Où $\hat{F}(x)$ est une approximation de la fonction $F(x)$. $K_V = \text{diag}[K_v, K_\omega]$ avec $K_v > 0$ et $K_\omega > 0$. μ est un terme de robustesse nécessaire pour compenser n'importe quel perturbation non-structurée.

En introduisant Eq. (3.36) dans Eq. (3.35a) nous obtenons le système à boucle fermée :

$$\bar{D}(q)\dot{e}_v = -\bar{C}(q, \dot{q})e_v - K_V e_v + F(x) - \hat{F}(x) + \mu \quad (3.37)$$

Les paramètres K_V , $\hat{F}(x)$ et μ doivent être bien choisis pour avoir une stabilisation asymptotique autour de $e_v = 0$. L'estimation de $F(x)$ est fournie par une approximation neuronale [31] :

$$\hat{F}(x) = \hat{V}^T \sigma(\hat{W}^T x) + \varepsilon \quad (3.38)$$

Alors la loi de commande Eq. (3.34) devient :

$$\bar{\tau} = \hat{V}^T \sigma(\hat{W}^T x) + K_V e_v - \mu + \varepsilon \quad (3.39)$$

Donc le système à boucle fermée Eq. (3.35) devient :

$$\bar{D}(q)\dot{e}_v = -\bar{C}(q, \dot{q})e_v - K_V e_v + V^T \sigma(W^T x) - \hat{V}^T \sigma(\hat{W}^T x) + \mu - \varepsilon \quad (3.40)$$

On choisit :

$$\mu(t) = -\bar{K}_\mu e_v \quad (3.41a)$$

$$\bar{K}_\mu = k_Z (\|\hat{Z}\|_F + w_M - 1) \quad (3.41b)$$

Où $\hat{Z} = \text{diag}\{\hat{V}, \hat{W}\}$, un développement détaillé de la fonction $\mu(t)$ est disponible dans [58].

Alors :

$$\bar{D}(q)\dot{e}_v = -(\bar{C}(q, \dot{q}) + K_V + K_\mu)e_v + V^T \sigma(W^T x) - \hat{V}^T \sigma(\hat{W}^T x) - \varepsilon \quad (3.42)$$

Les lois d'adaptation des poids de réseau de neurones sont choisies comme suit :

$$\dot{\hat{V}} = F \hat{\sigma} e_v^T - F \hat{\sigma}' \hat{W}^T x e_v^T - F \lambda \|e_v\| \hat{V} \quad (3.43a)$$

$$\dot{\hat{W}} = G x (\hat{\sigma}'^T \hat{V} e_v)^T - G \lambda \|e_v\| \hat{W} \quad (3.43b)$$

Ou V et W sont les vecteurs des poids de la couche de sortie et la couche cachée respectivement. $\hat{\sigma}'$ est le gradient correspondant au chaque neurones de la couche cachée.

Pour une fonction sigmoïde σ' , $\hat{\sigma}'$ est donné par :

$$\hat{\sigma}'(\hat{S}) = \left[\frac{\partial \sigma(S)}{\partial S} \right]_{S=\hat{S}} = \text{diag}\{\sigma(\hat{W}^T x)\} [I - \text{diag}\{\sigma(\hat{W}^T x)\}] \quad (3.44)$$

F et G sont des matrices positives définies et représentent le taux d'apprentissage, $\lambda > 0$ stabilise l'erreur e_v autour d'une région $e_v = 0$.

La loi d'adaptation Eq. (3.41) est composée essentiellement de trois termes : Le premier est rien qu'une back-propagation normale. Le dernier est le terme e-modifié vient de la théorie de contrôle adaptatif [60]. Le terme médian est un terme ajouté pour prouver la stabilité de la commande [58].

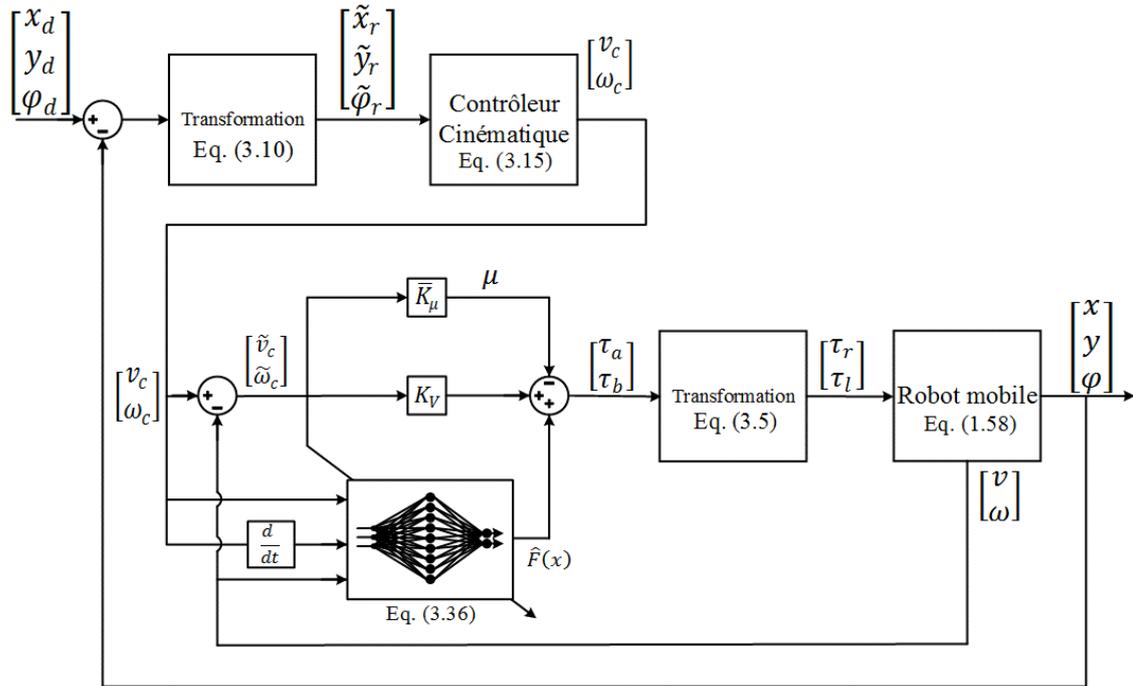


Figure III.17 Bloc-diagramme de la commande neuronal étudiée.

La stabilité de cette commande est assurée via la méthode de Lyapunov, où la fonction de Lyapunov candidate est :

$$V = k_x(\tilde{x}_r^2 + \tilde{y}_r^2) + 2k_\varphi v_d(1 - \cos\tilde{\varphi}_r) + V_1 \quad (3.45a)$$

Où V_1 est :

$$V_1 = \frac{1}{2} [e_v^T \bar{D} e_v + tr\{\tilde{V}^T F^{-1} \tilde{V}\} + tr\{\tilde{W}^T G^{-1} \tilde{W}\}] \quad (3.45b)$$

Tels que

$$\tilde{V} = V - \hat{V}$$

$$\tilde{W} = W - \hat{W}$$

La dérivée de V est donnée par :

$$\dot{V} = 2k_x(\tilde{x}_r \dot{\tilde{x}}_r + \tilde{y}_r \dot{\tilde{y}}_r) + 2k_\varphi v_d \dot{\tilde{\varphi}}_r \sin\tilde{\varphi}_r + \dot{V}_1 \quad (3.46)$$

La condition :

$$\|e_v\| < b_v(k_v^{-1}) \quad (3.47)$$

Où b_v est une limite correctement sélectionnée assure que $\dot{V} < 0$. Ce qui signifie que l'erreur $\|e_v\|$ et les erreurs des poids du réseau de neurones sont bornées. Il est noté que l'erreur $\|e_v\|$ peut-être gardée arbitrairement petite par l'augmentation de K_v .

Plus de détail est trouvé dans [57][58][59].

Dans ce chapitre nous avons présenté une étude préliminaire et les étapes de conception de deux commandes, la première est basée sur le backstepping, et la deuxième sur les réseaux de neurones et le backstepping. Dans le chapitre suivant on va voir les résultats de simulation réalisés pour étudier la performance de ces deux commandes et mettre en évidence le phénomène de glissement.

| CHAPITRE IV : SIMULATION

IV-1. Suivi de trajectoire par backstepping.

IV-2. Suivi de trajectoire par réseaux de neurones.

Dans ce chapitre, nous allons présenter les résultats de la simulation pour toutes les commandes étudiées dans les chapitres précédents. La simulation est réalisée en utilisant l'outil MATLAB / SIMULINK. Nous avons choisi les paramètres du robot comme suit:

$$m_b = 9 \text{ kg}$$

$$m_w = 0.5 \text{ kg}$$

$$r = 0.095 \text{ m}$$

$$a = 0.24 \text{ m}$$

$$d = 0.05 \text{ m}$$

$$I_{bz} = 0.537 \text{ kg.m}^2$$

$$I_{wz} = 0.0011 \text{ kg.m}^2$$

$$I_{wy} = 0.0023 \text{ kg.m}^2$$

Nous allons présenter pour chaque commande trois scénarios :

- Un mouvement de roulement pur (on néglige tous les glissements).
- Un mouvement avec glissement sur une surface (1) non glissante.
- Un mouvement avec glissement sur une surface (2) glissante.

Les trajectoires utilisées sont :

- Une ligne droite.
- Un cercle.
- Un virage à 90° degrés.

IV-1. SUIVI DE TRAJECTOIRE PAR BACKSTEPPING :

IV-1-1. TRAJECTOIRE LIGNE DROITE :

Suivi d'une trajectoire ligne cas d'un roulement pur (sans glissement):

Condition initial : $x = 2$, $y = 0$, $\varphi = \pi/4$.

Gains du contrôleur cinématique : $k_x = 8$, $k_y = 4$, $k_\varphi = 2$.

Gains du contrôleur dynamique : $k_a = 15$, $k_b = 15$.

Temps de simulation : 10s .

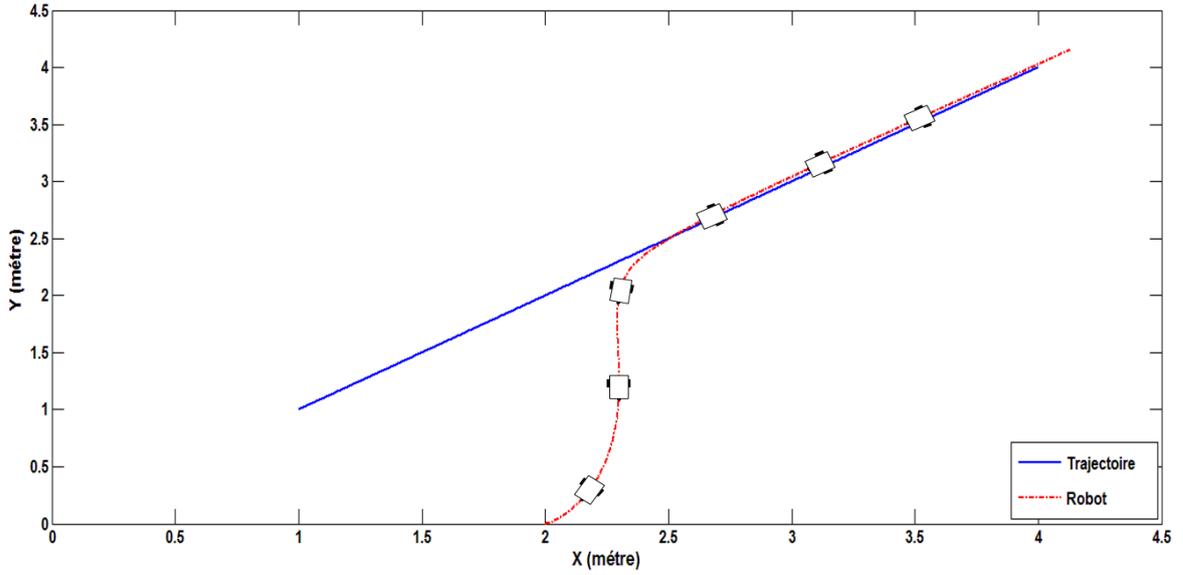


Figure IV.1 Suivi de la trajectoire ligne (roulement pur)

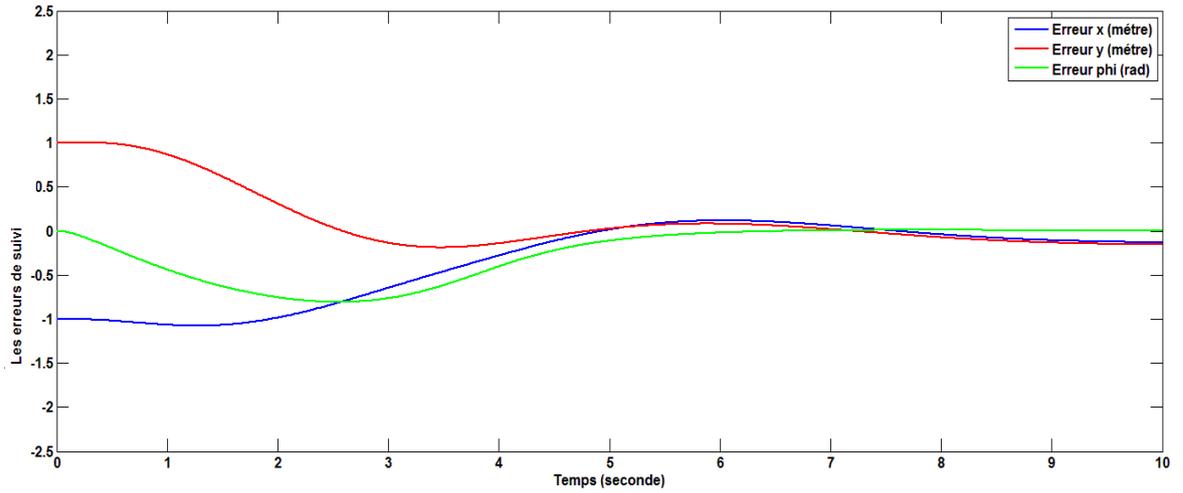


Figure IV.2 Les erreurs du suivi (ligne-roulement pur)

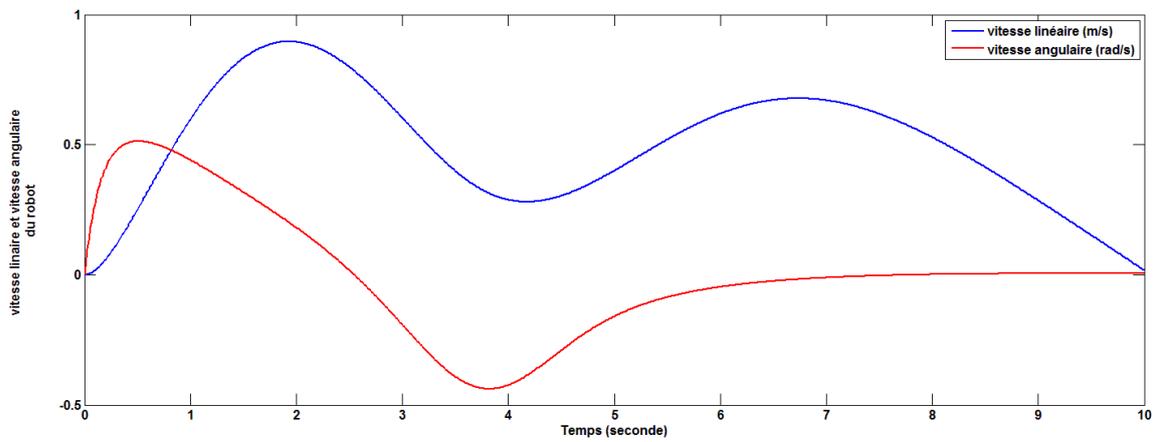


Figure IV.3 Les vitesses du robot (ligne-roulement pur)

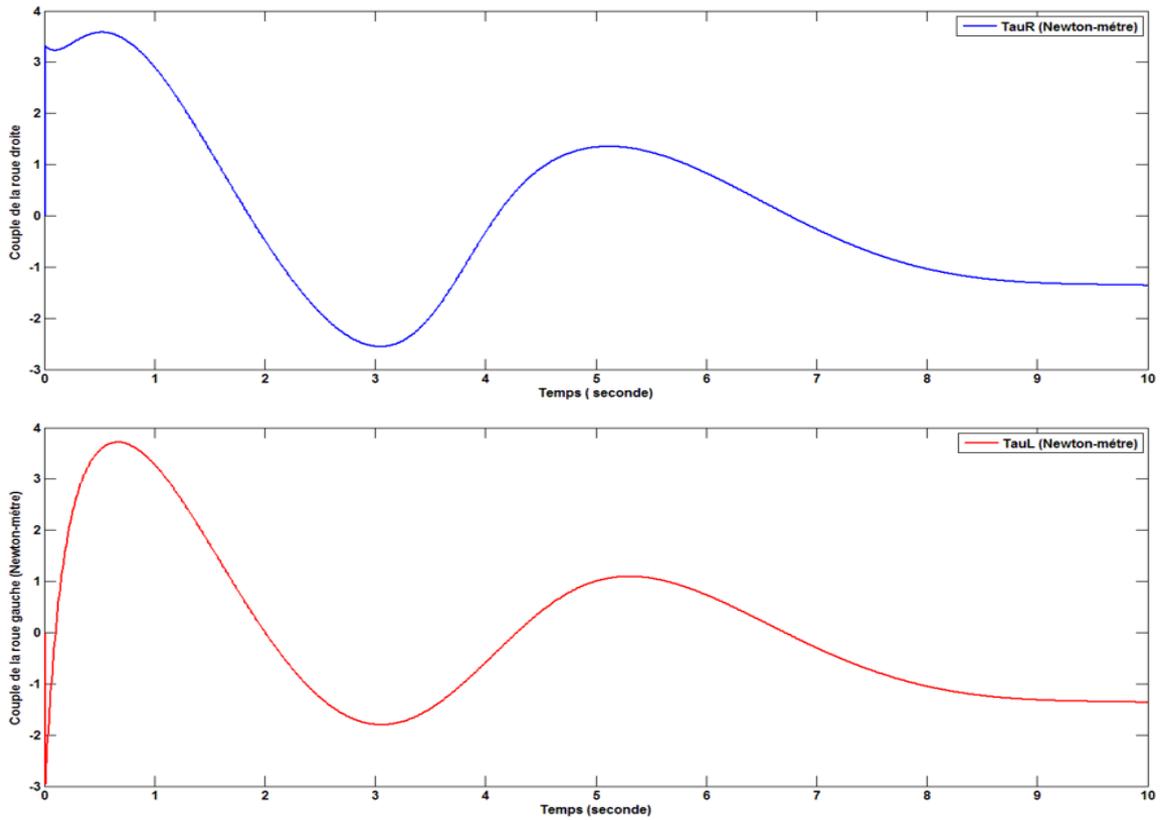


Figure IV.4 Les couples des roues droite et gauche (ligne-roulement pur)

Suivi d'une trajectoire ligne avec glissement cas de surface (1) :

Il est clair que lorsqu'on prend le glissement en considération, le suivi est affecté sévèrement. Mais la commande continue à rectifier les erreurs du suivi tant que le glissement est dans un intervalle admissible Fig IV.5.

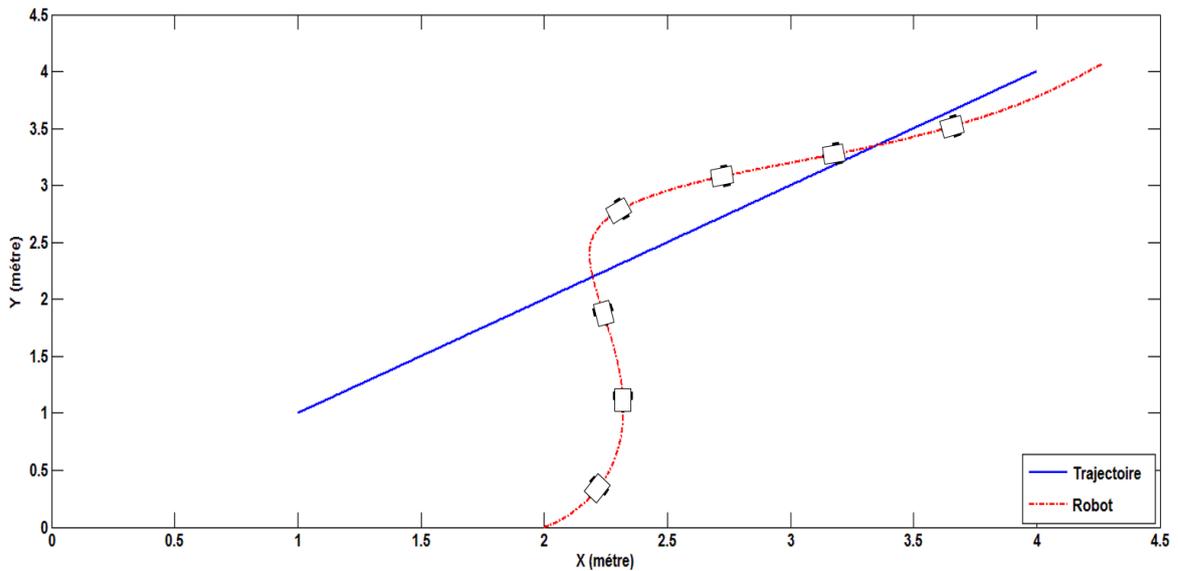


Figure IV.5 Suivi de la trajectoire ligne cas de surface (1)

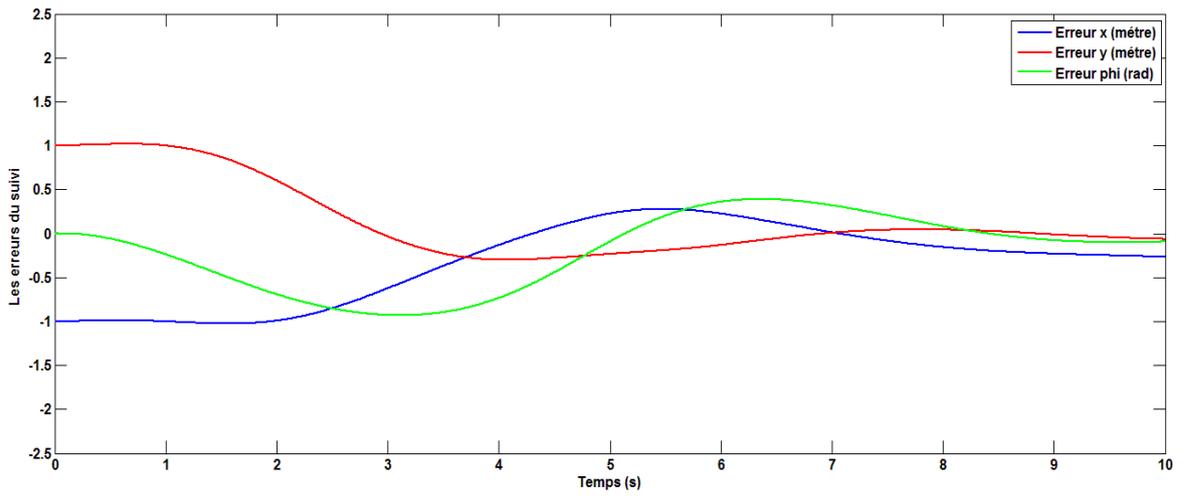


Figure IV.6 Les erreurs de suivi pour la surface (1)

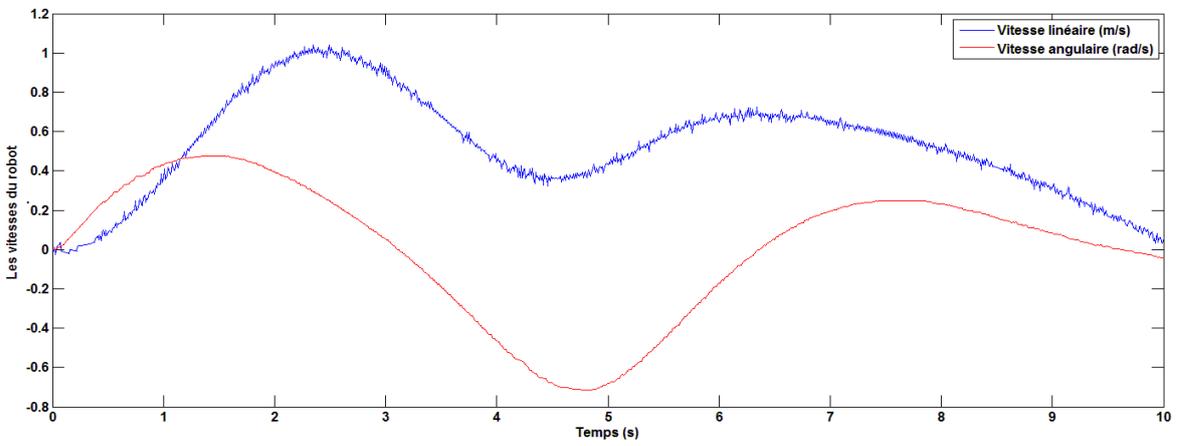


Figure IV.7 Les vitesses du robot surface (1)

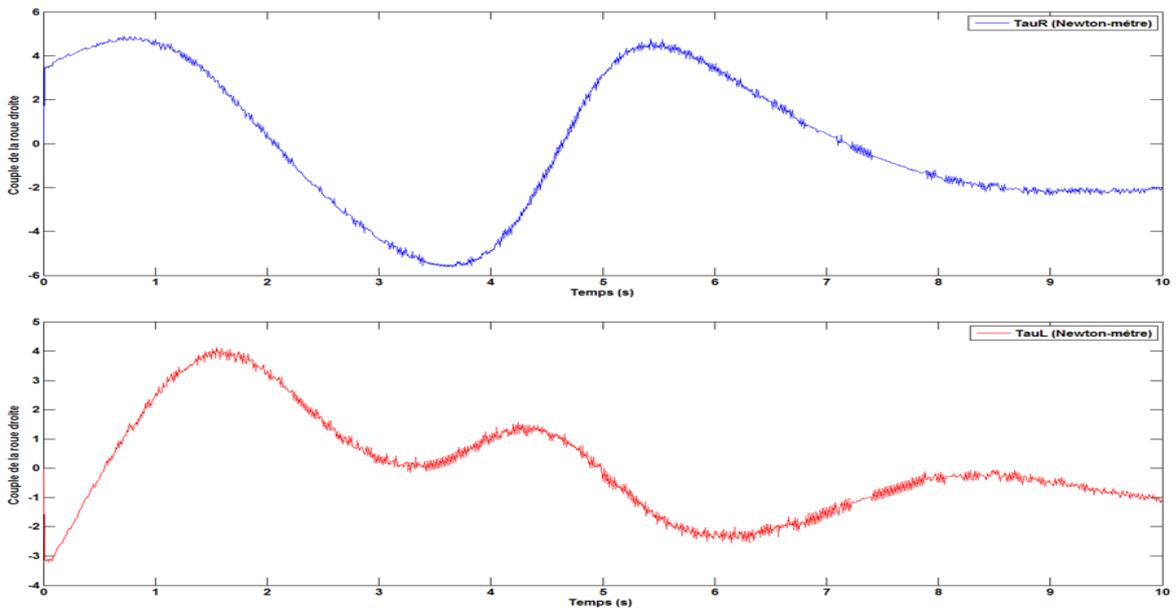


Figure IV.8 Les couples moteurs droite et gauche surface (1)

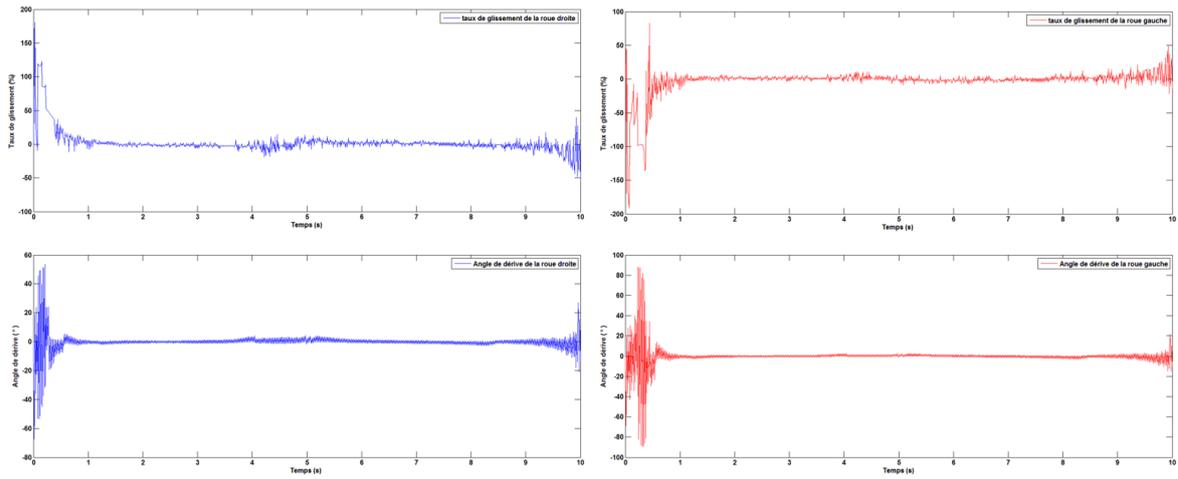


Figure IV.9 Les taux de glissement et les angles de dérive gauche et droite surface (1)

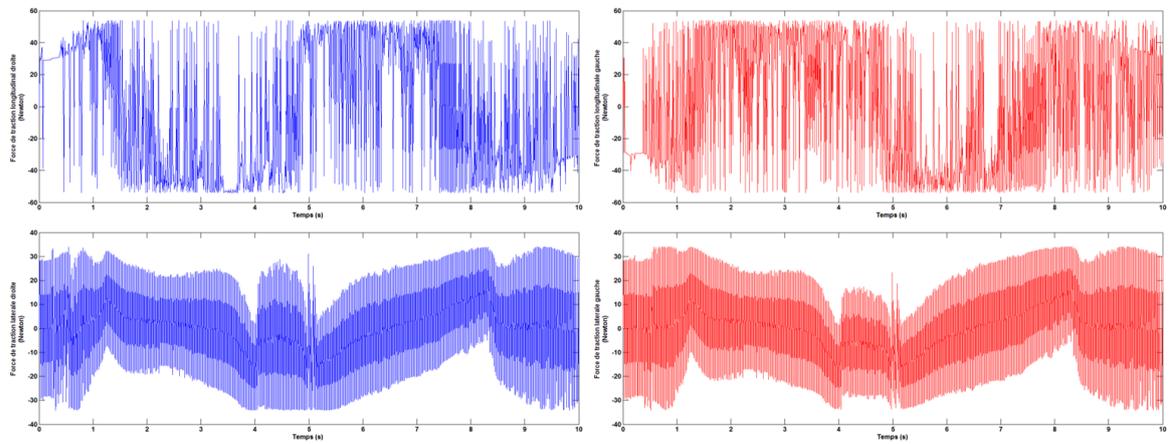


Figure IV.10 Les forces de traction longitudinales et latérales droite et gauche surface (1)

Suivi d'une trajectoire ligne avec glissement cas de surface (2) :

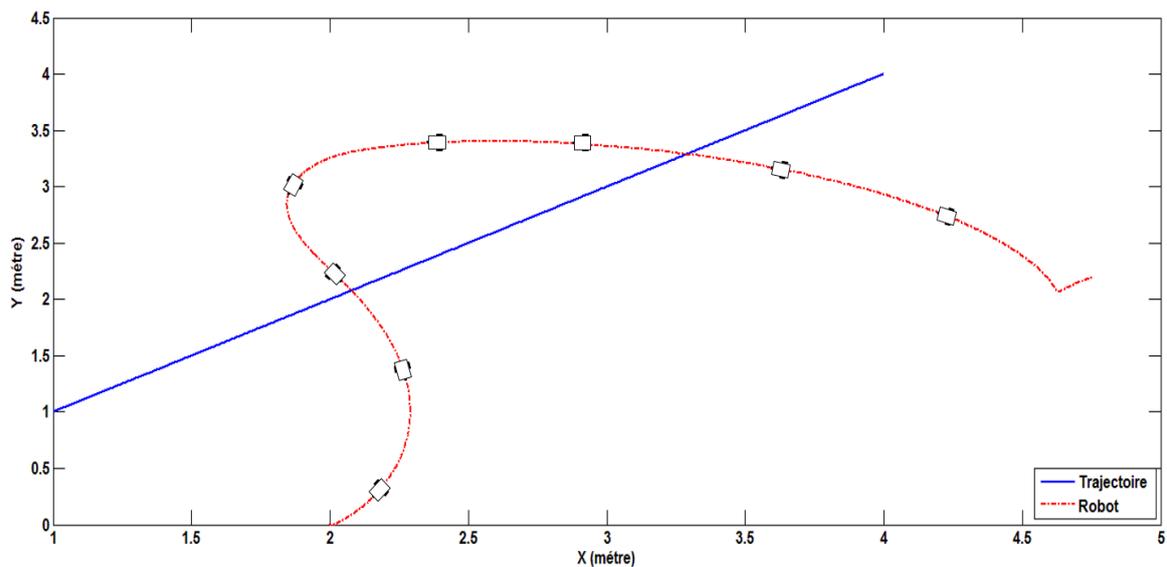


Figure IV.11 Suivi de trajectoire ligne cas de surface (2)

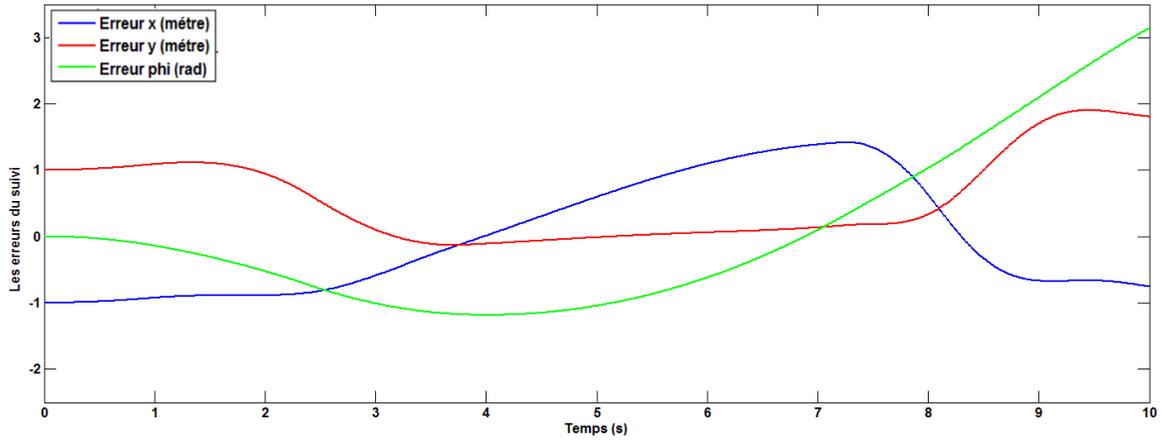


Figure IV.12 Les erreurs du suivi surface (2)

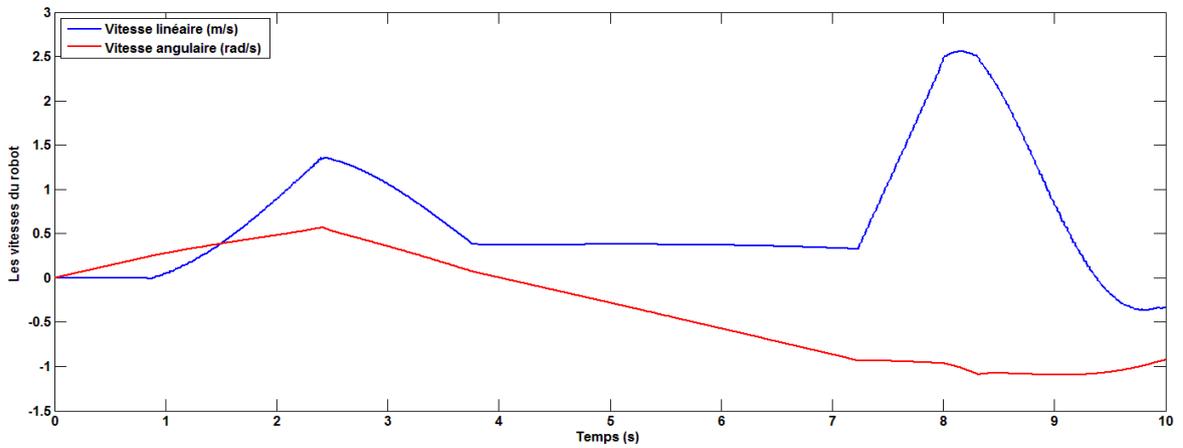


Figure IV.13 Les vitesses du robot surface (2)

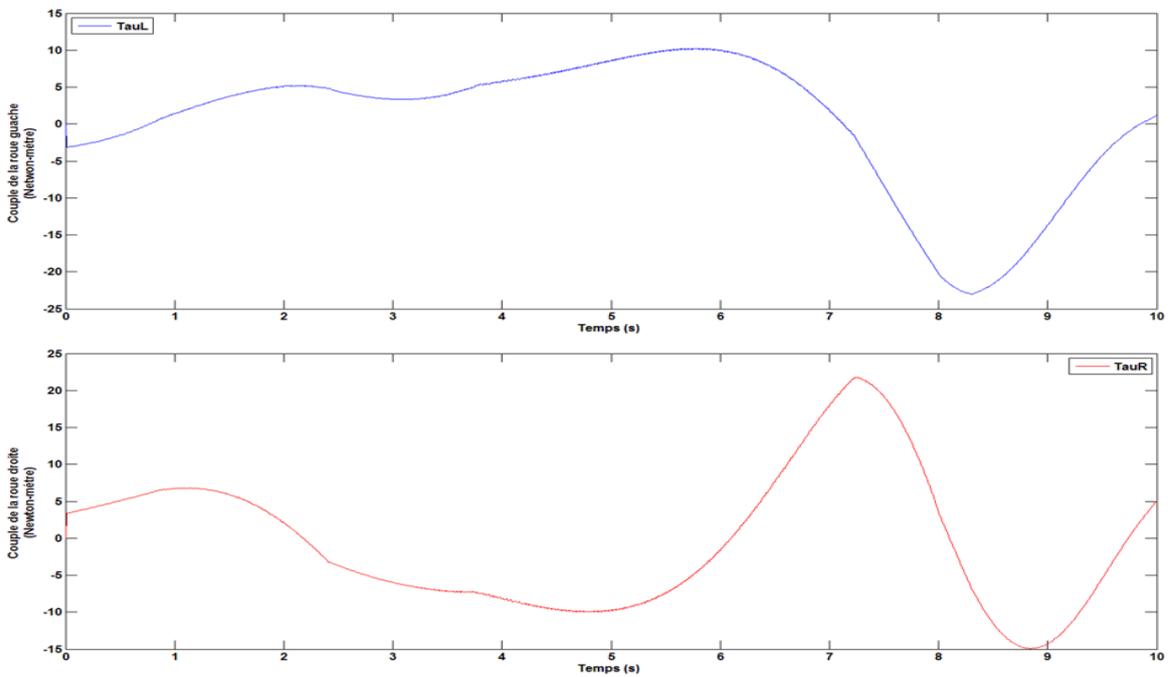


Figure IV.14 Les couples moteurs des roues droite et gauche surface (2)

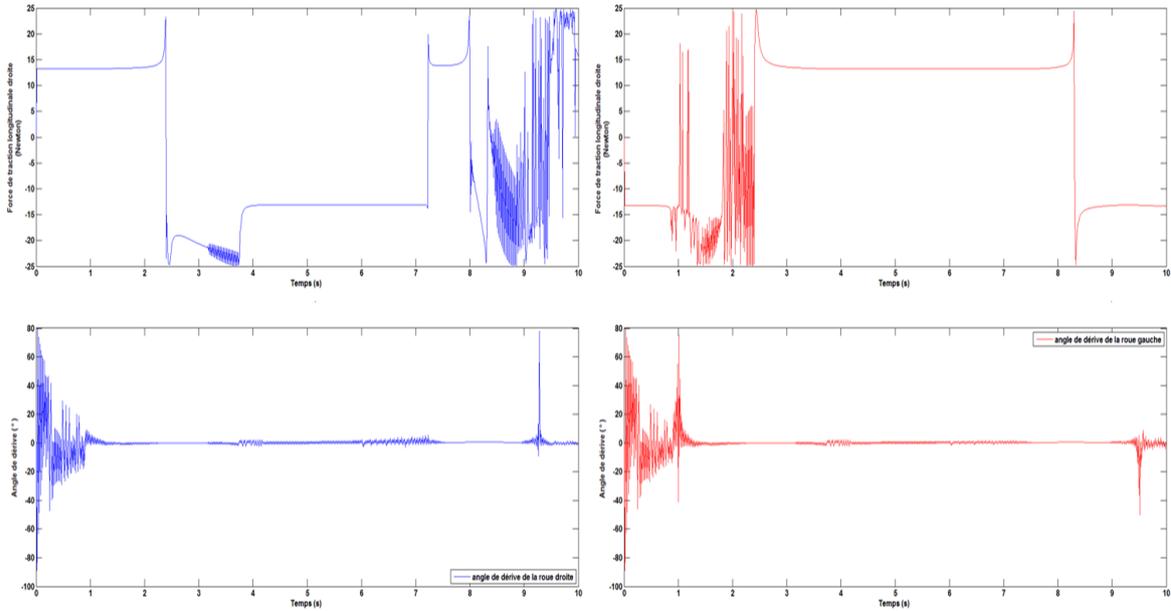


Figure IV.15 Les taux de glissement et les angles de dérive droites et gauches surface (2)

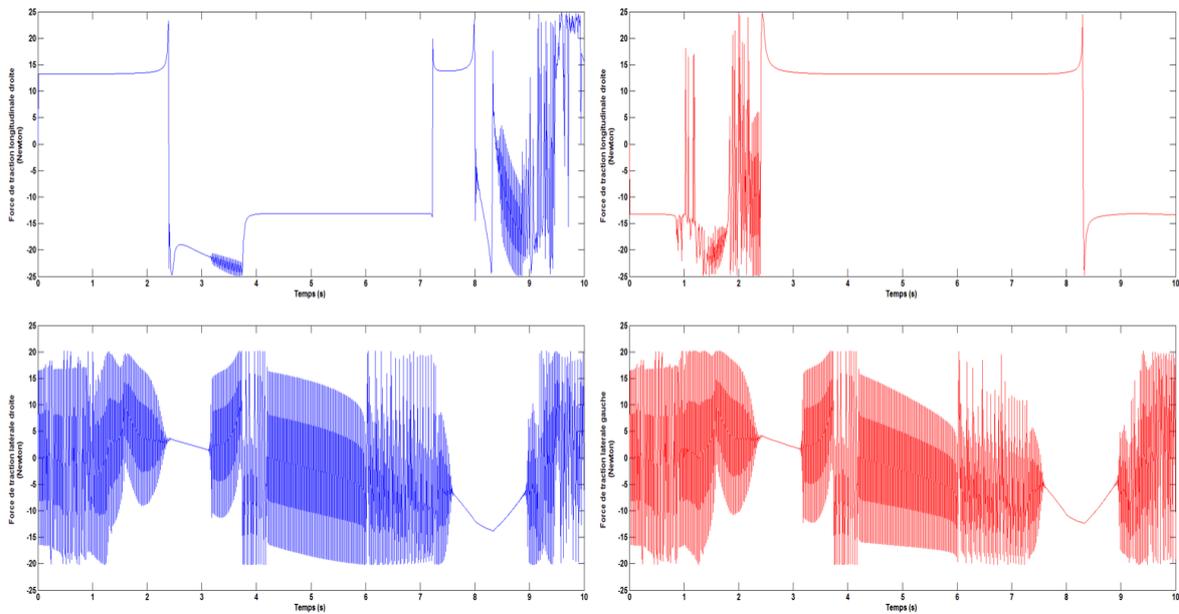


Figure IV.16 Les forces de traction longitudinales et latérale droite et gauche surface(2)

La différence entre la surface (1) et (2) est la diminution des forces de traction longitudinales et latérales Fig IV.10 et Fig IV.16 .Un max de presque 55 N et 30 N pour les forces de traction longitudinales et latérales respectivement cas de surface (1). 25 N -20 N pour la surface (2). La commande backstepping ne pouvait pas maintenir la stabilité du suivi pour la surface (2). Mais sa réponse partiellement lente (8 s) pour la surface (1) est satisfaisante Fig IV.6 .

IV-1-2. TRAJECTOIRE CERCLE :

Suivi d'une trajectoire cercle cas d'un roulement pur (sans glissement):

Condition initial : $x = 4, y = 0, \varphi = \pi/2$.

Gains du contrôleur cinématique : $k_x = 12, k_y = 8, k_\varphi = 3$.

Gains du contrôleur dynamique : $k_a = 15, k_b = 15$. Temps de simulation : 15s .

Il est vu que la commande backstepping est stable pour le cas de roulement pur Fig IV.17.

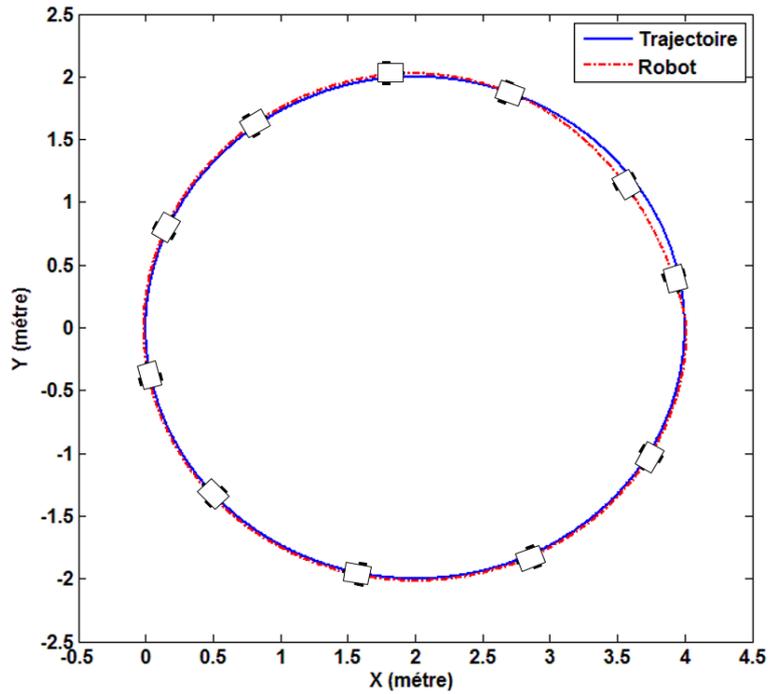


Figure IV.17 Suivi d'une trajectoire cercle sans glissement

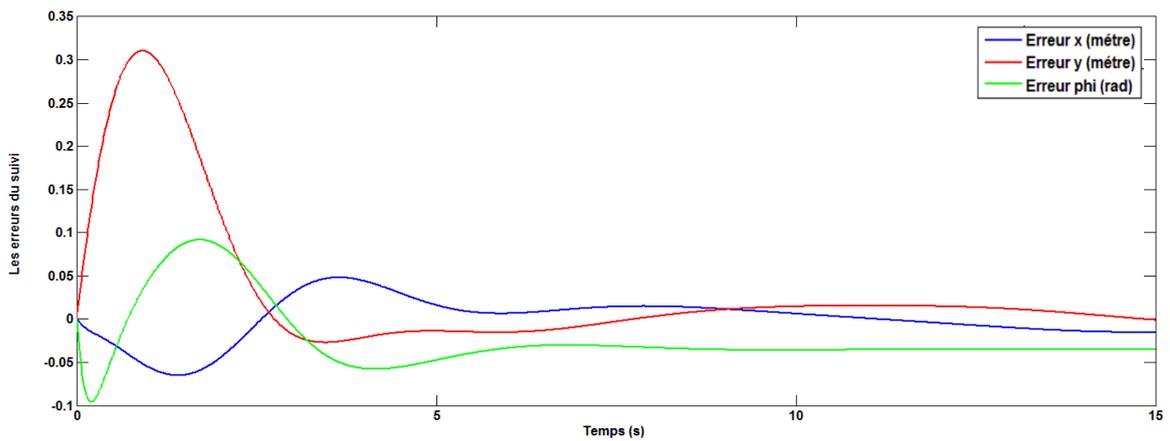


Figure IV.18 Les erreurs du suivi sans glissement

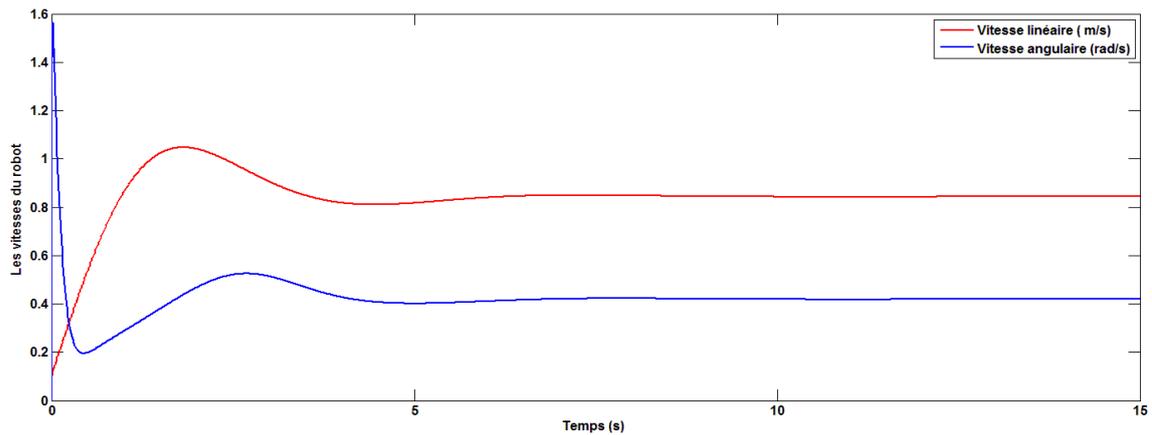


Figure IV.19 Les vitesses linéaire et angulaire du robot sans glissement

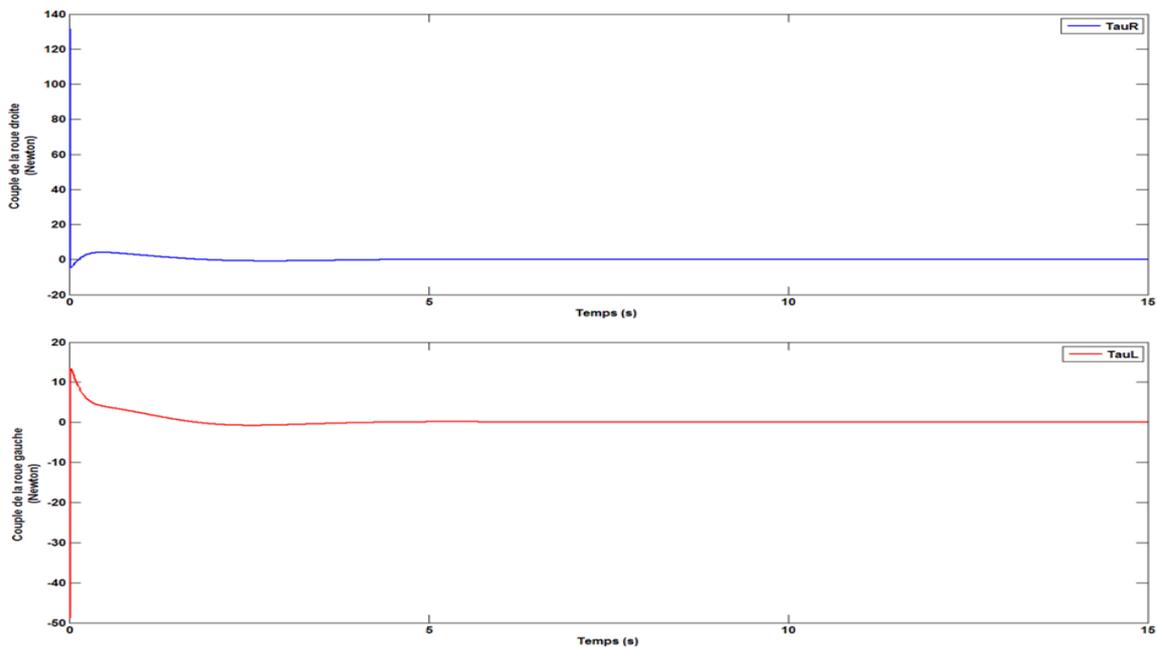


Figure IV.20 Les couples moteurs des roues droite et gauche sans glissement

Suivi d'une trajectoire cercle cas de surface (1):

Il est vu que la commande backstepping est capable de stabiliser le suivi de la trajectoire sauf que pour quelque parties ou le glissement latérale est très grand. Parce que la vitesse référence de démarrage été maximale on peut voir un glissement longitudinal sur la figure des vitesses au début Fig IV.23 et les taux de glissement Fig IV.25. La commande était capable de minimiser les erreurs de suivi qu'après 9 secondes.

La surface (2) était assez glissante pour que la commande soit stable. La diminution des forces de traction est claire dans les figures Fig IV.26 et Fig IV.32. On note aussi que le comportement du robot après son dérapage est incompréhensible. À cause de la contrainte horaire de la trajectoire le robot a été incapable de récupérer sa position de poursuite Fig IV.27.

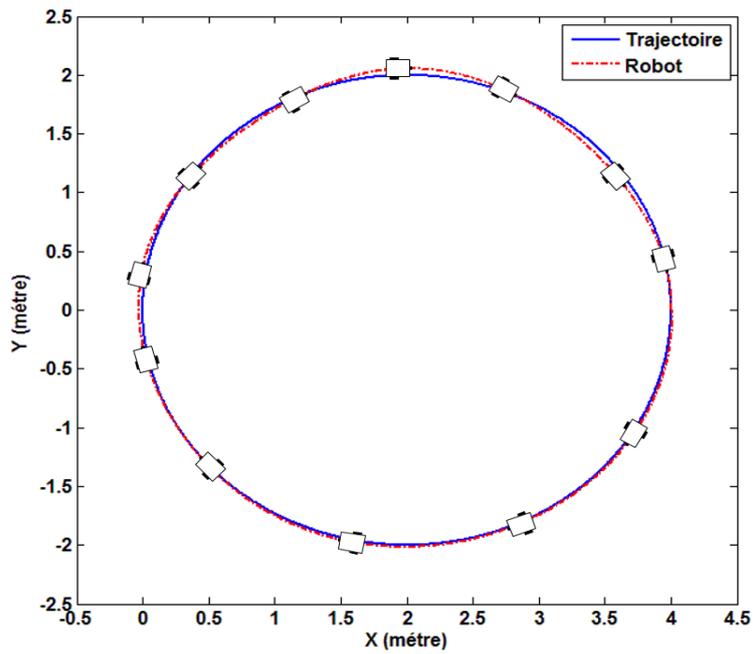


Figure IV.21 Suivi d'une trajectoire cercle surface (1)

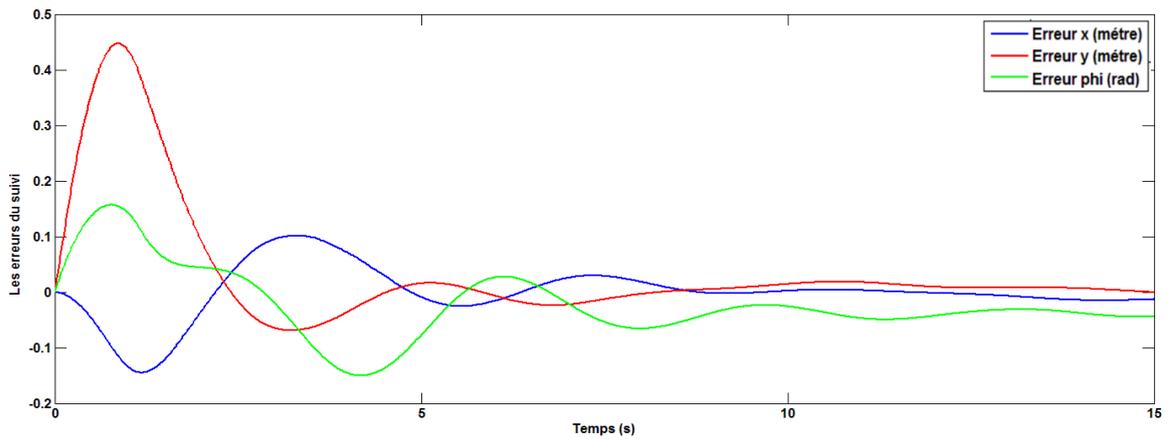


Figure IV.22 Les erreurs du suivi surface (1)

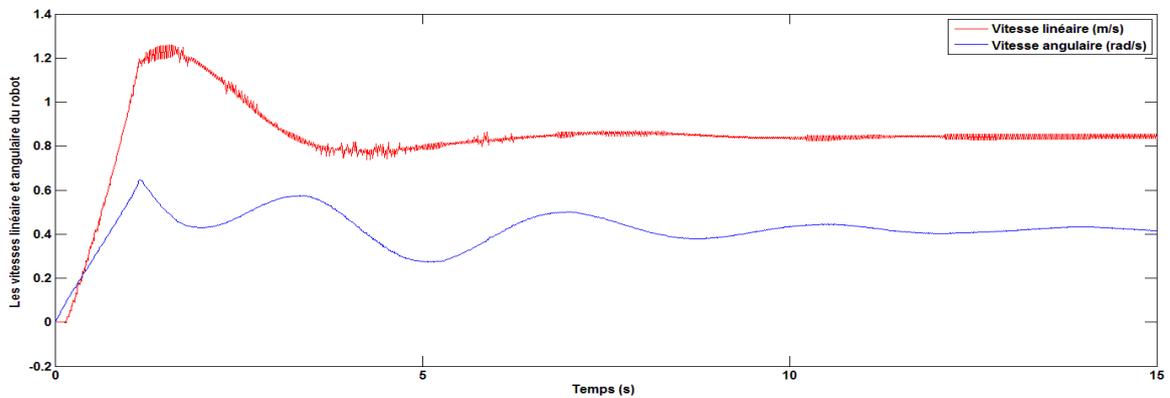


Figure IV.23 Les vitesses linéaire et angulaire du robot surface (1)

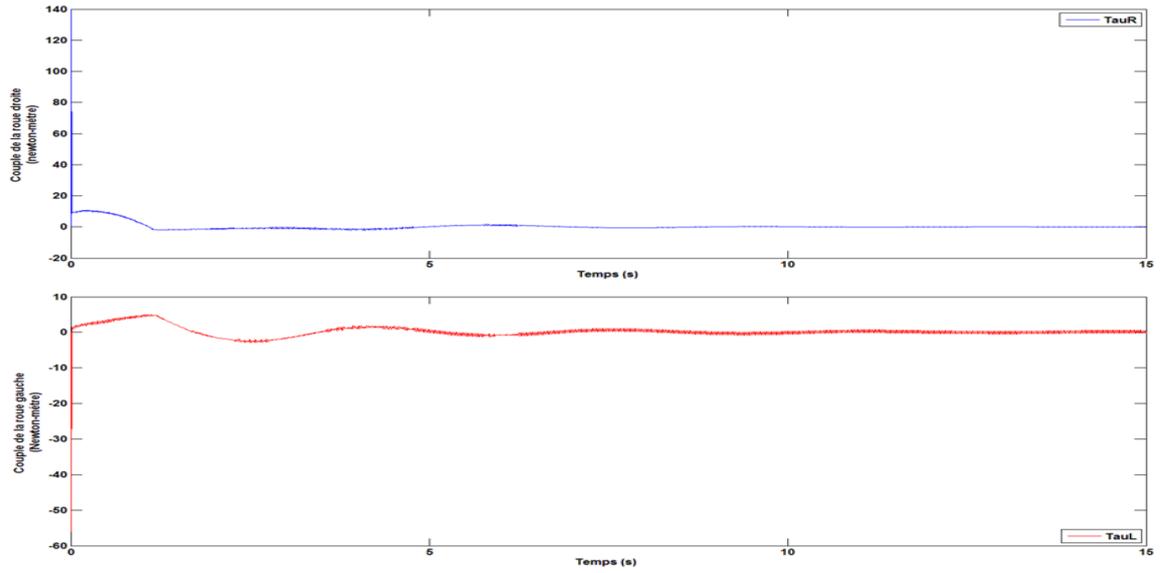


Figure IV.24 Les couples moteurs des roues droite et gauche surface (1)

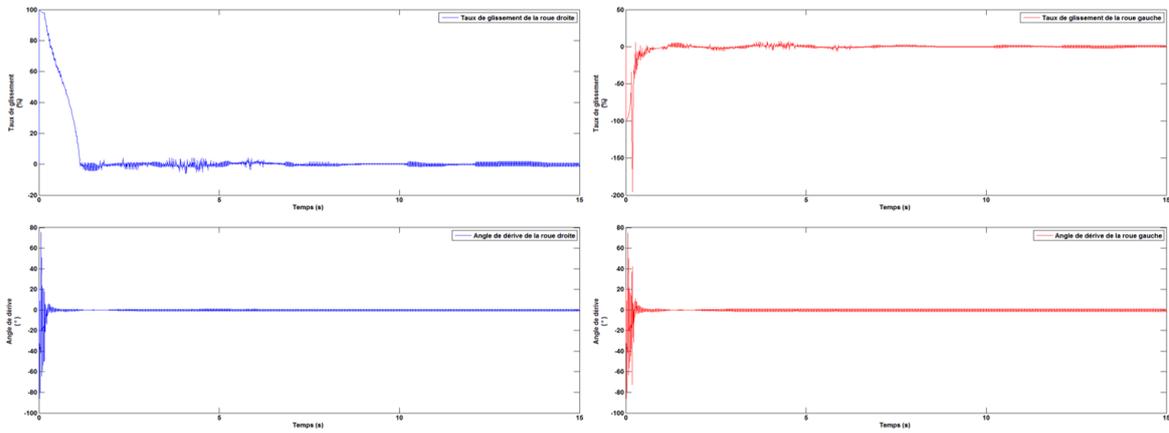


Figure IV.25 Les taux de glissement et les angles de dérive droite et gauche surface (1)

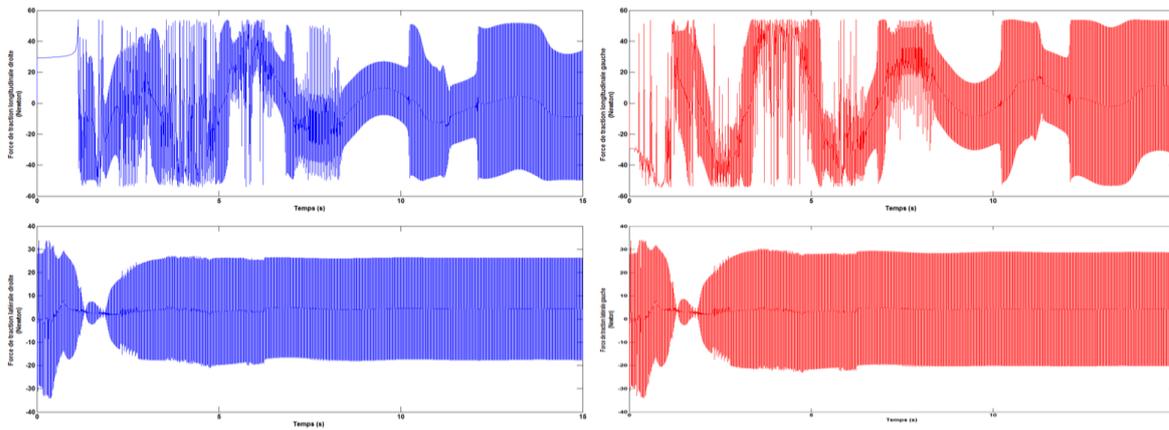


Figure IV.26 Les forces de traction longitudinales et latérales droite et gauche surface (1)

Suivi d'une trajectoire cercle cas de surface (2):

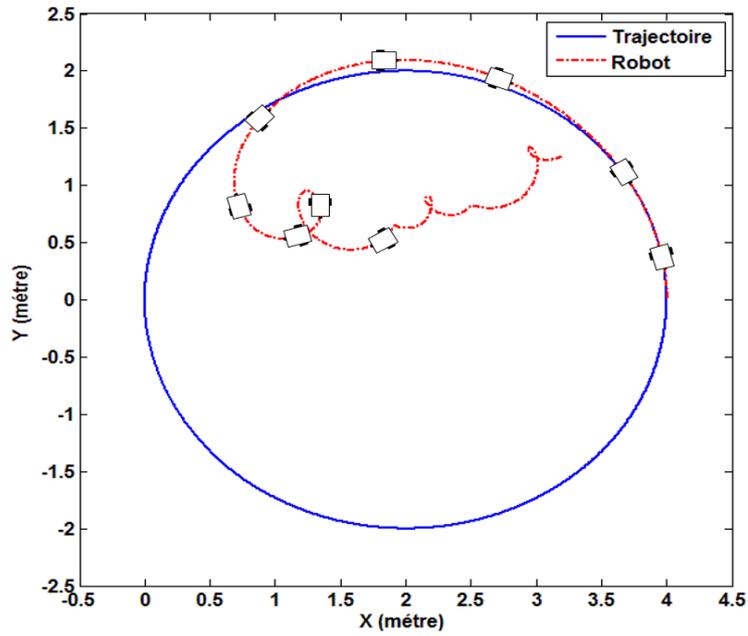


Figure IV.27 Suivi d'une trajectoire cercle surface (2)

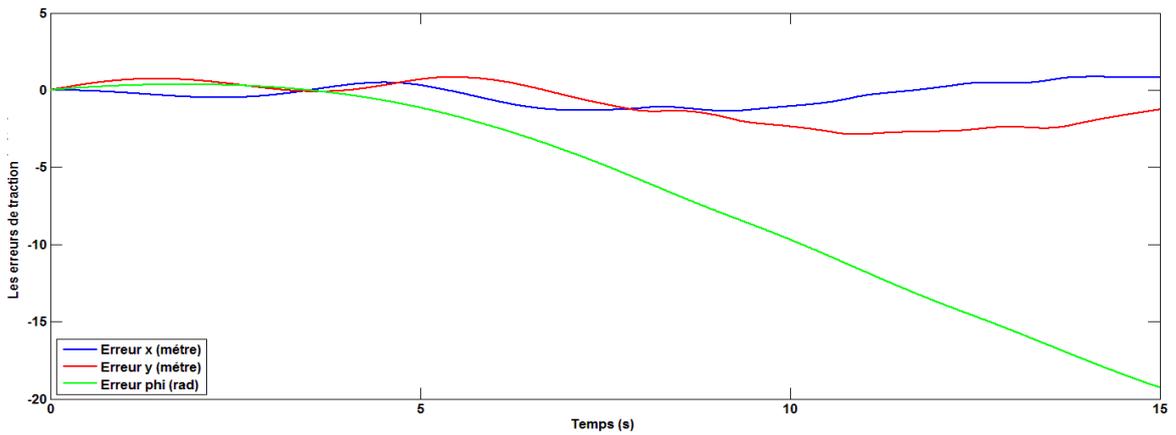


Figure IV.28 Les erreurs du suivi surface (2)

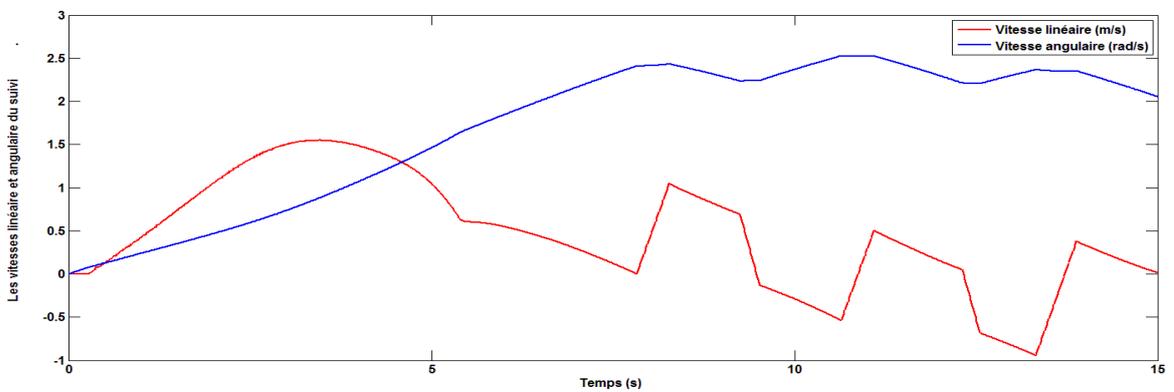


Figure IV.29 Les vitesses linéaire et angulaire du robot surface (2)

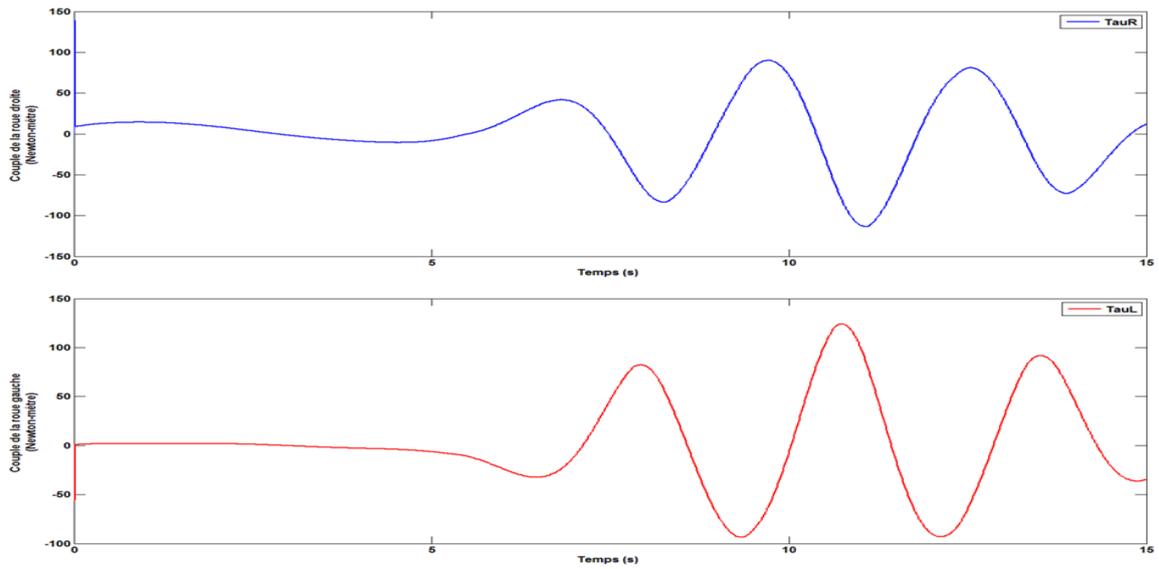


Figure IV.30 Les couple moteurs des roues droites et gauche surface (2)

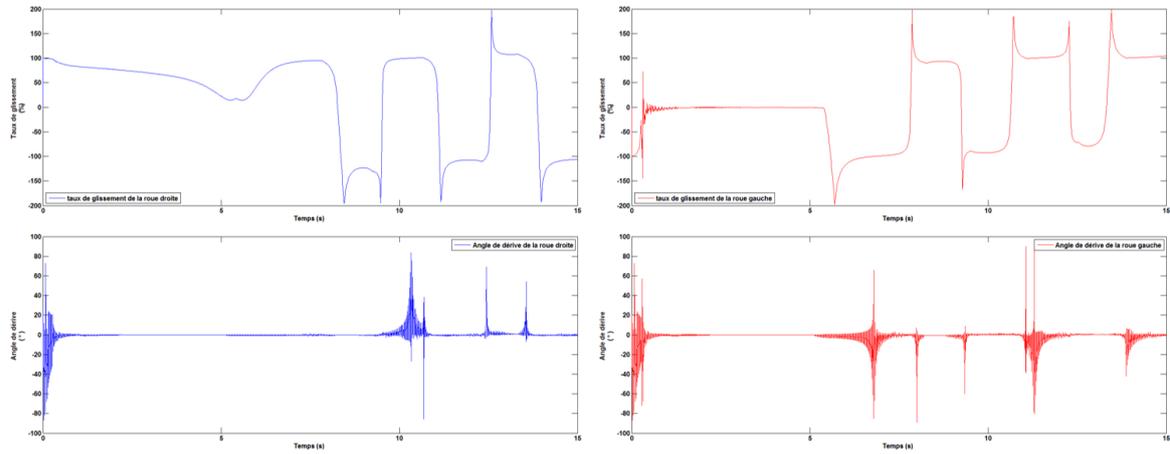


Figure IV.31 Les taux de glissement et les angles de dérive droite et gauche surface (2)

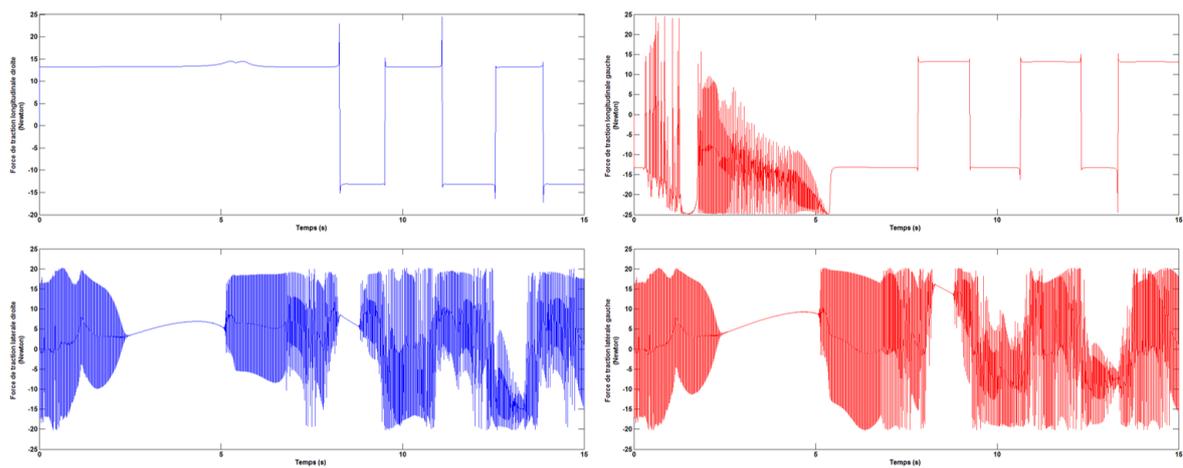


Figure IV.32 Les forces de traction longitudinales et latérale gauche et droite surface (2)

IV-1-3. VIRAGE A 90° :

Suivi d'une trajectoire virage à 90° cas d'un roulement pur (sans glissement):

Condition initial : $x = 0, y = 0, \varphi = 0$.

Gains du contrôleur cinématique : $k_x = 5, k_y = 5, k_\varphi = 5$.

Gains du contrôleur dynamique : $k_a = 15, k_b = 15$. Temps de simulation : 10s .

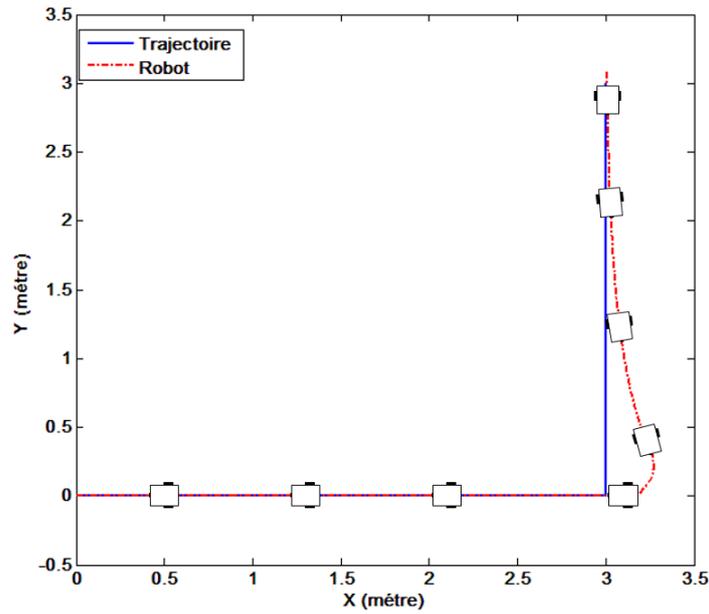


Figure IV.33 Suivi d'une trajectoire L (sans glissement)

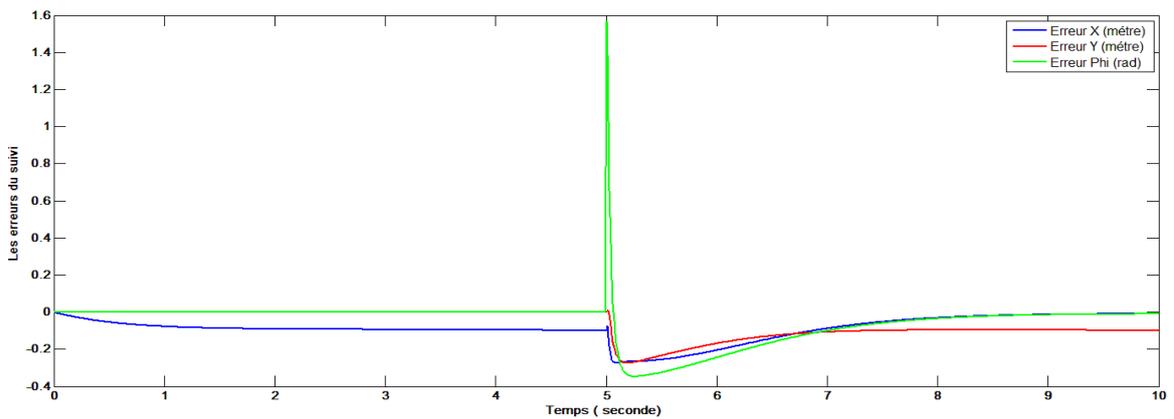


Figure IV.34 Les erreurs du suivi (sans glissement)

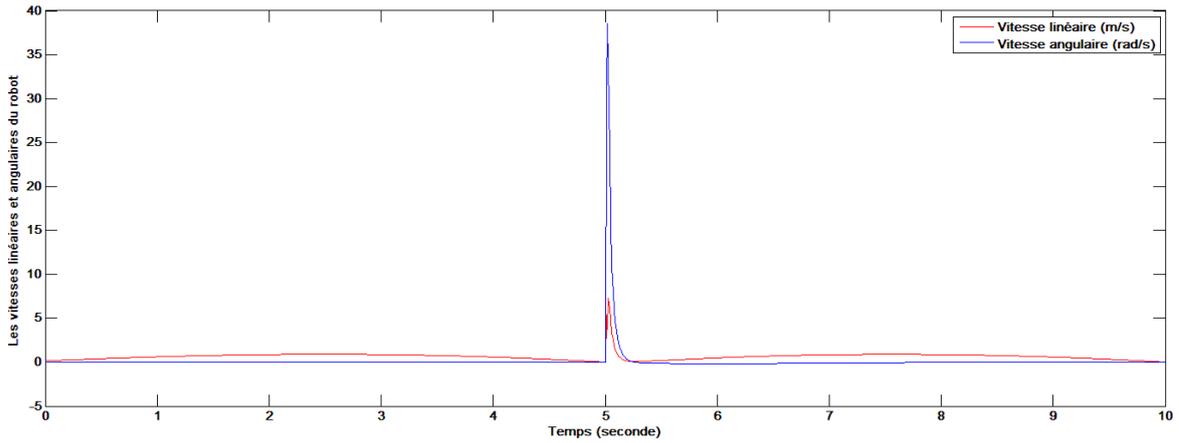


Figure IV.35 Les vitesses linéaire et angulaire du robot (sans glissement)

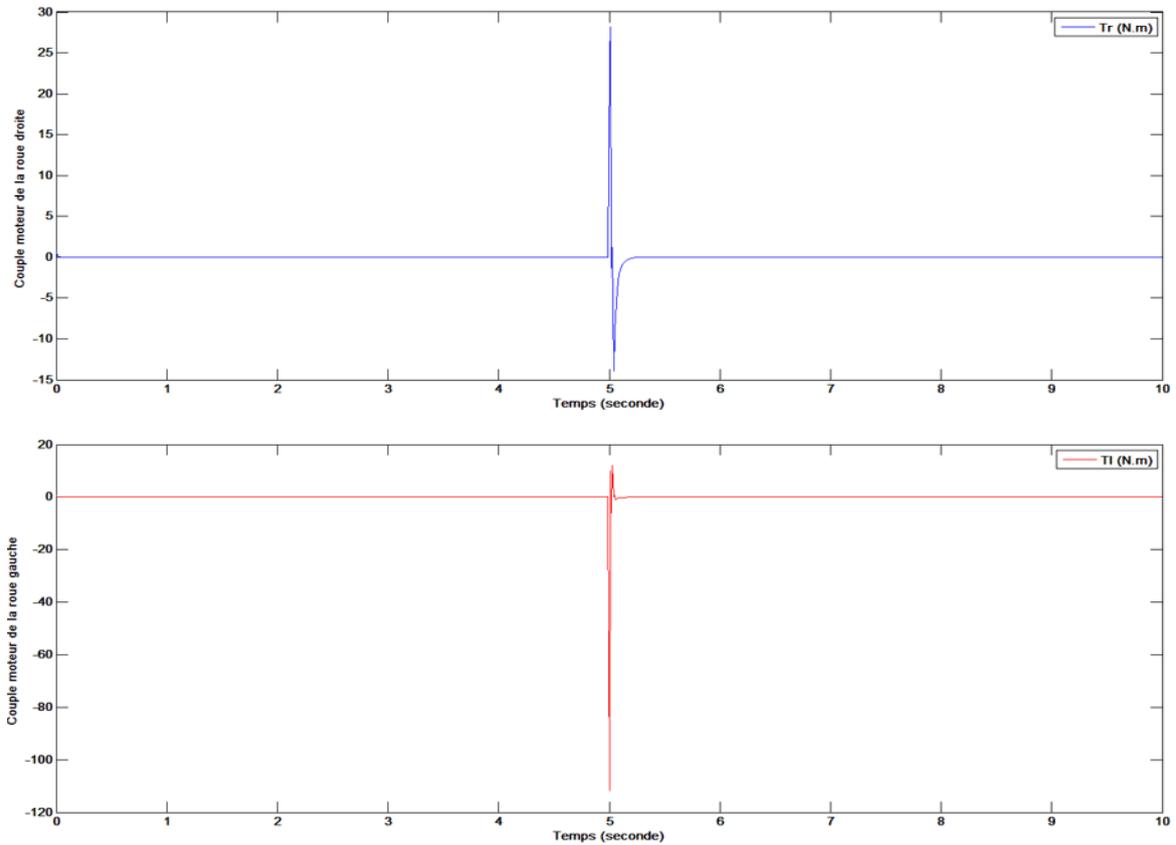


Figure IV.36 Les couples moteurs droite et gauche (sans glissement)

Pour la trajectoire L et pour le cas d'un mouvement sans glissement, la commande backstepping a été stable. L'erreur au virage est résultante d'un changement brusque de l'angle φ de 0 rad à $\frac{\pi}{2} \text{ rad}$ Fig IV.34. La commande a rectifiée ce changement en 2 s Fig IV.34. Même chose pour la surface (1), mais la commande n'a pas pu rectifier l'erreur au virage qu'après 3 s Fig IV.38.

Suivi d'une trajectoire virage à 90° cas de surface (1) :

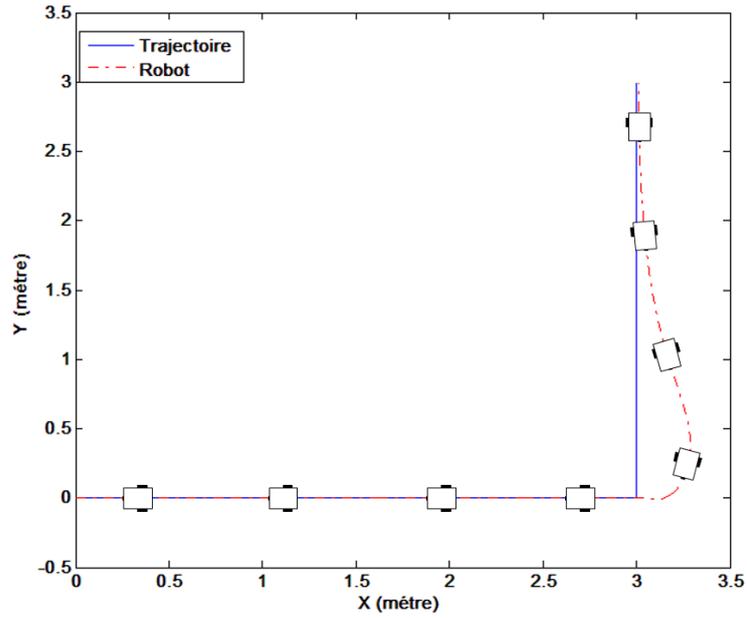


Figure IV.37 Suivi d'une trajectoire L (surface (1))

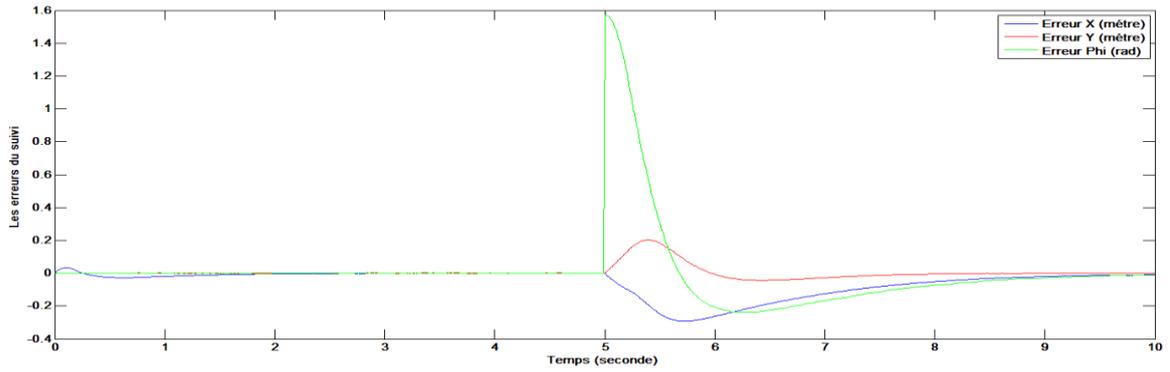


Figure IV.38 Les erreurs du suivi (surface (1))

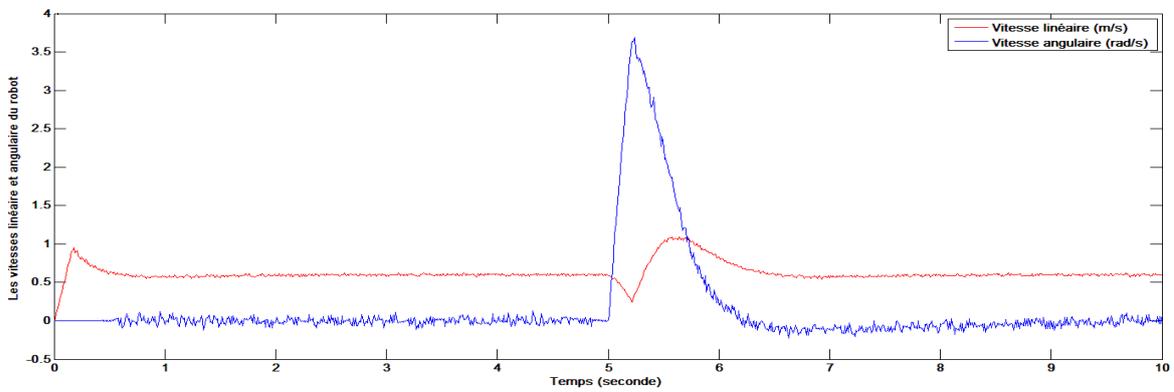


Figure IV.39 Les vitesses linéaire et angulaire du robot (surface (1))

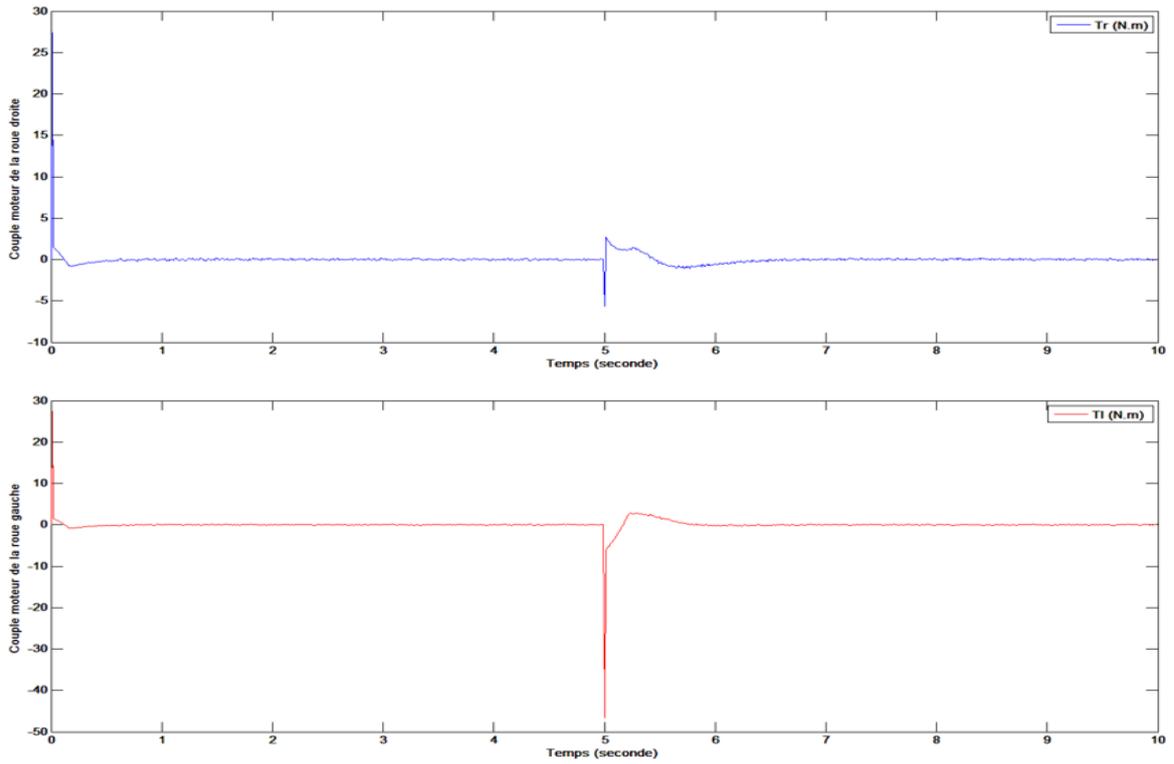


Figure IV.40 Les couples moteurs droite et gauche (surface (1))

Suivi d'une trajectoire virage à 90° cas de surface (2) :

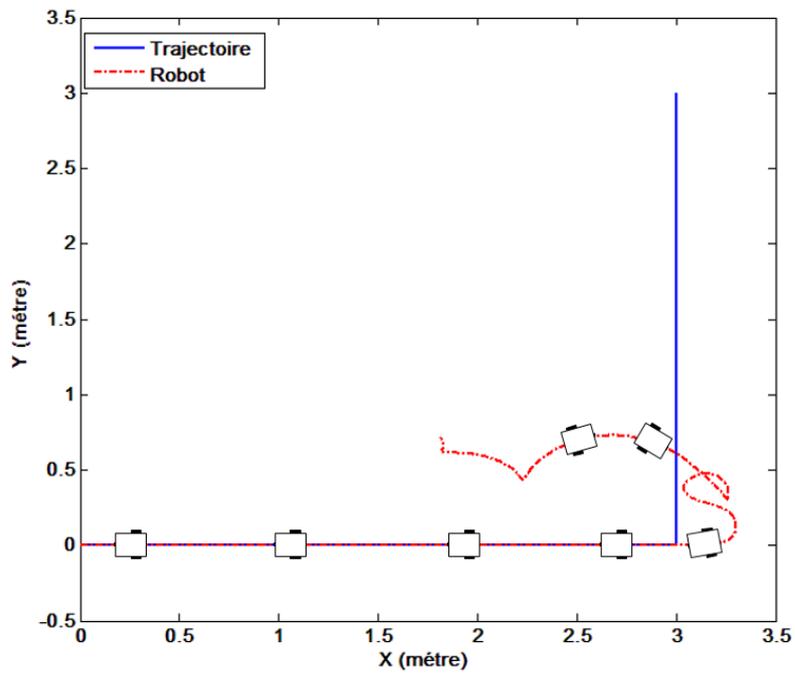


Figure IV.41 Suivi d'une trajectoire L (surface (2))

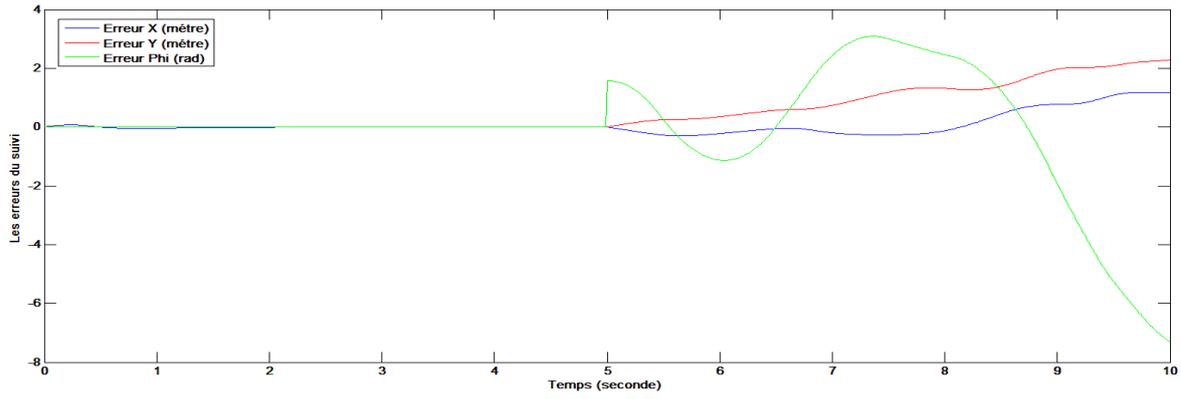


Figure IV.42 Les erreurs du suivi (surface (2))

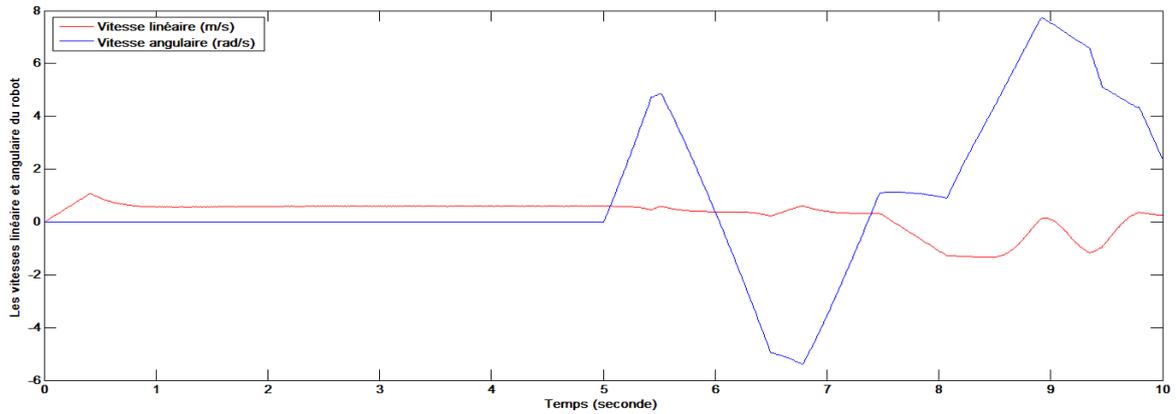


Figure IV.43 Les vitesses linéaire et angulaire du robot (surface (2))

Pour la surface (2) la commande a été capable de stabiliser le suivi le long de la partie droite de la trajectoire. Mais lorsque le virage, le robot a perdu son chemin sous l'effet du glissement Fig IV.41.

Il ressort de ses simulations que la commande backstepping a donné des bons résultats pour la surface (1) presque identiques aux résultats théoriques (mouvement sans glissement). Mais elle a été incapable de stabiliser la poursuite sur la surface (2) glissante pour toutes les trajectoires.

IV-2. SUIVI DE TRAJECTOIRE PAR COMMANDE NEURONALE :

IV-2-1. TRAJECTOIRE LIGNE DROITE :

Suivi d'une trajectoire ligne avec glissement cas de surface (1) :

Condition initial : $x = 2$, $y = 0$, $\varphi = \pi/4$.

Gains du contrôleur cinématique : $k_x = 1.5$, $k_y = 5$, $k_\varphi = 3$.

Gains d'apprentissage du réseau de neurones : $F = 10$, $G = 10$

Gains du terme adaptatif : $k = 0.01$. Gain de terme robuste : $k_z = 1$, $w_{max} = 5$.

Gains du PD : $k_v = 20$, $k_w = 20$. Temps de simulation : 10 s.

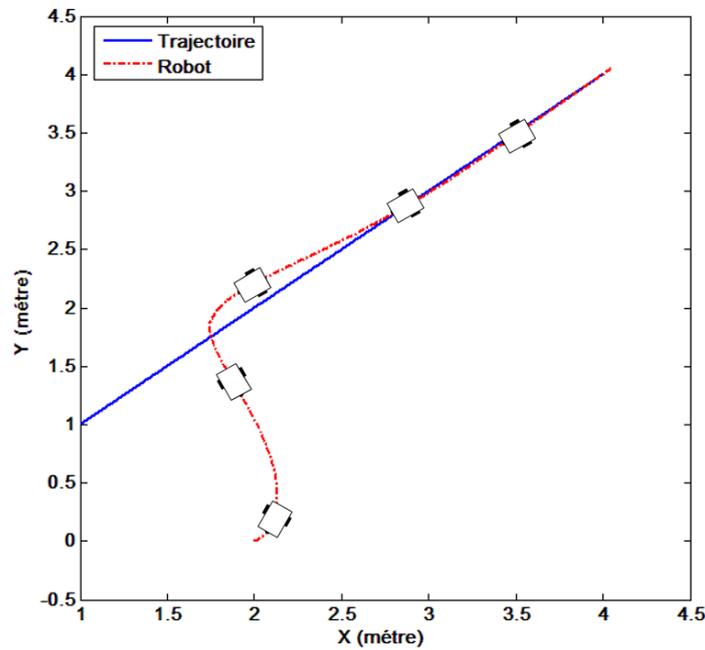


Figure IV.44 Suivi d'une trajectoire ligne droite (surface 1))

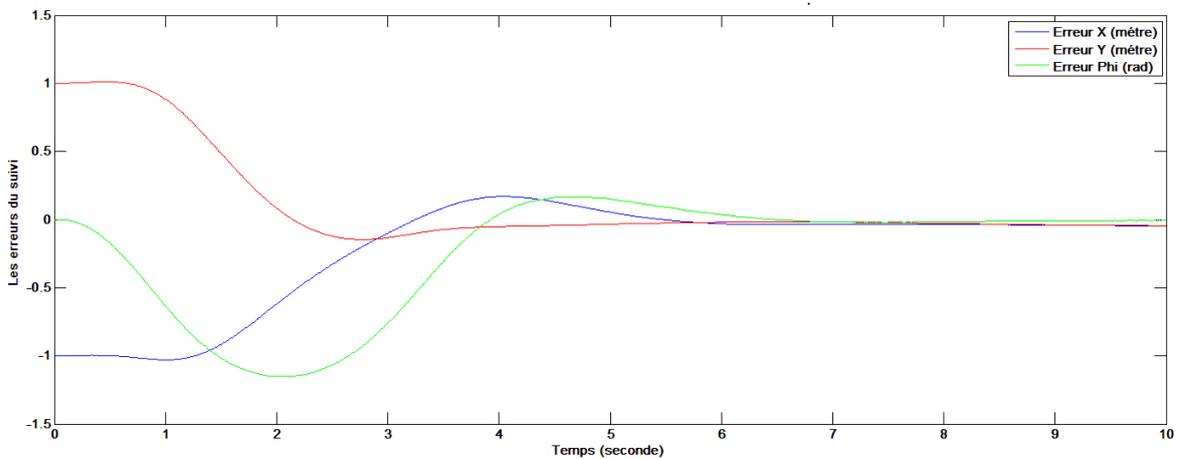


Figure IV.45 Les erreurs du suivi (surface 1))

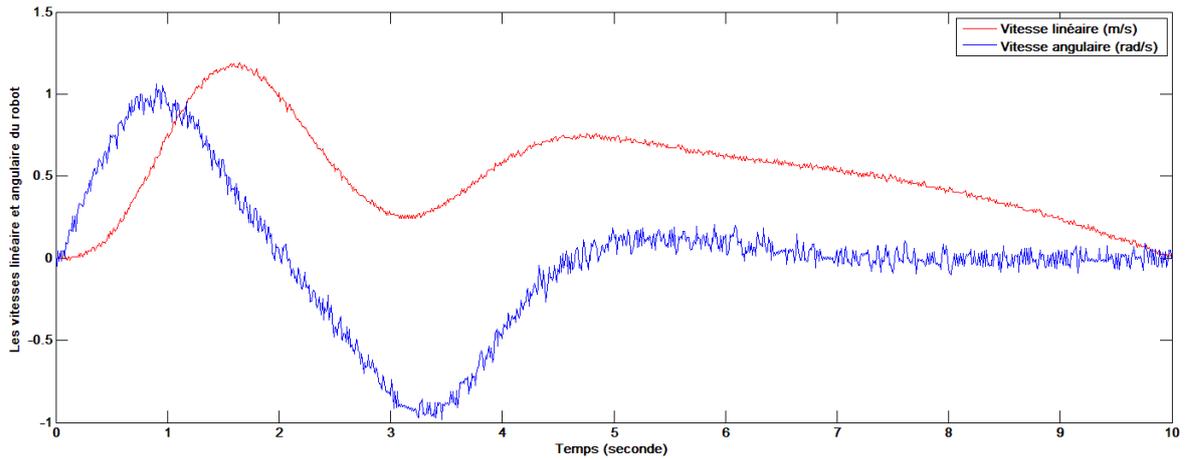


Figure IV.46 Les vitesses linéaire et angulaire du robot (surface (1))

On peut constater que la repense de la commande neuronale est plus vite (6 s) et a été capable de rendre l'erreur du suivi presque « 0 » Fig IV.45.

Suivi d'une trajectoire ligne avec glissement cas de surface (2) :

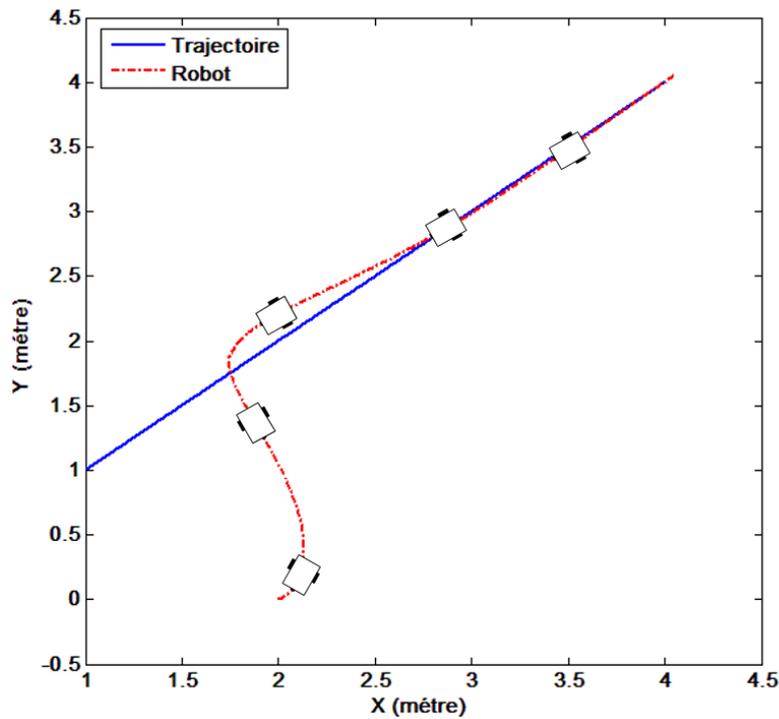


Figure IV.47 Suivi d'une trajectoire ligne droite (surface (2))

Les réponses de la commande neuronale pour les deux surfaces sont identiques, malgré la diminution des forces de tractions cas de la surface (2) Fig IV.52 . Le réseau de neurones a été stable, et ses poids convergent vers des valeurs stables Fig IV.54. La fonction F approximée est présentée dans la figure Fig IV. 53.

Le bon choix des gains du contrôleur cinématique peut donner des meilleurs résultats.

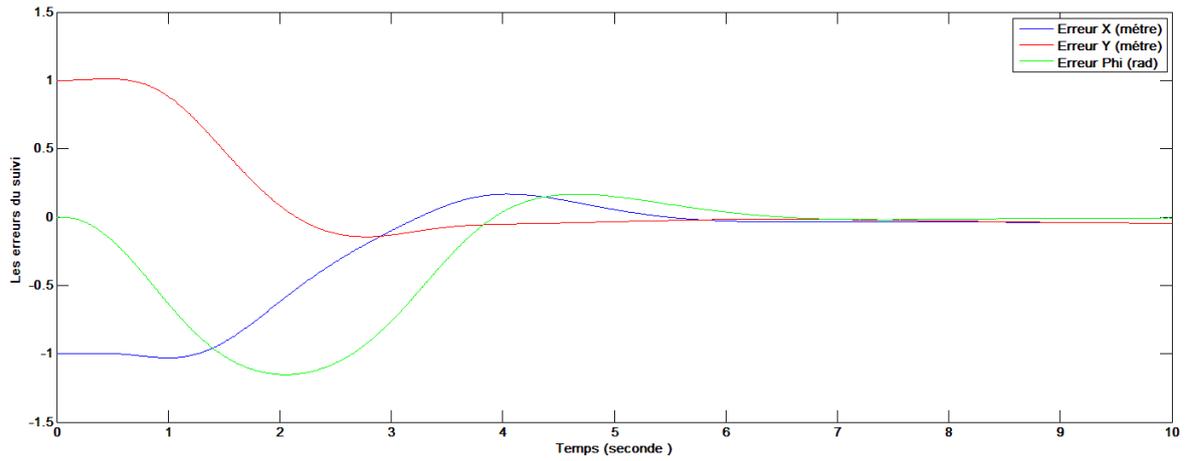


Figure IV.48 Les erreurs du suivi (surface (2))

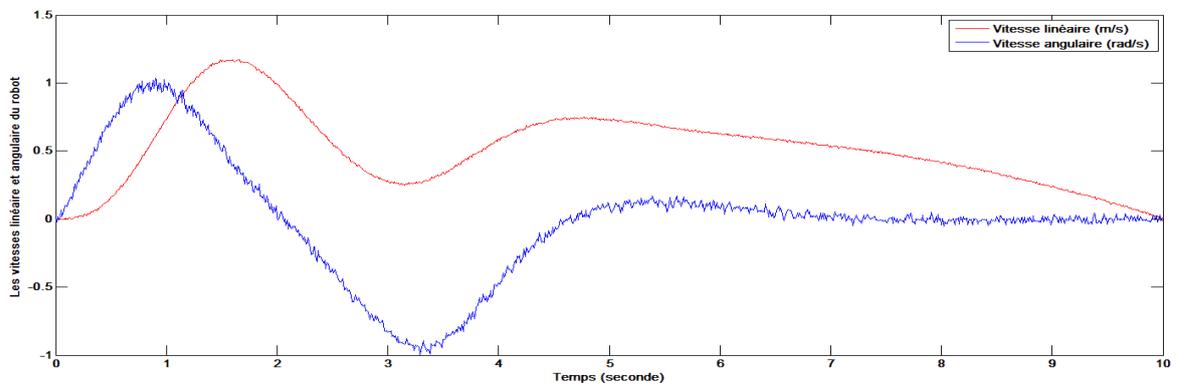


Figure IV.49 Les vitesses linéaire et angulaire du robot (surface (2))

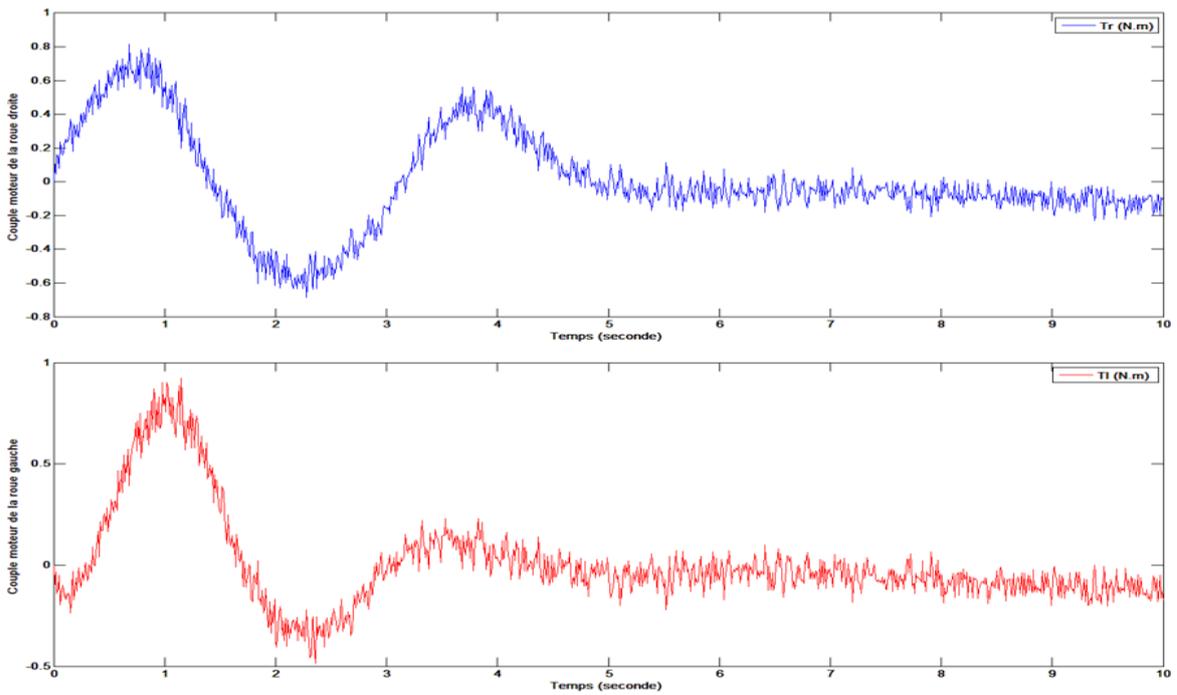


Figure IV.50 Les couples moteurs droite et gauche du robot (surface (2))

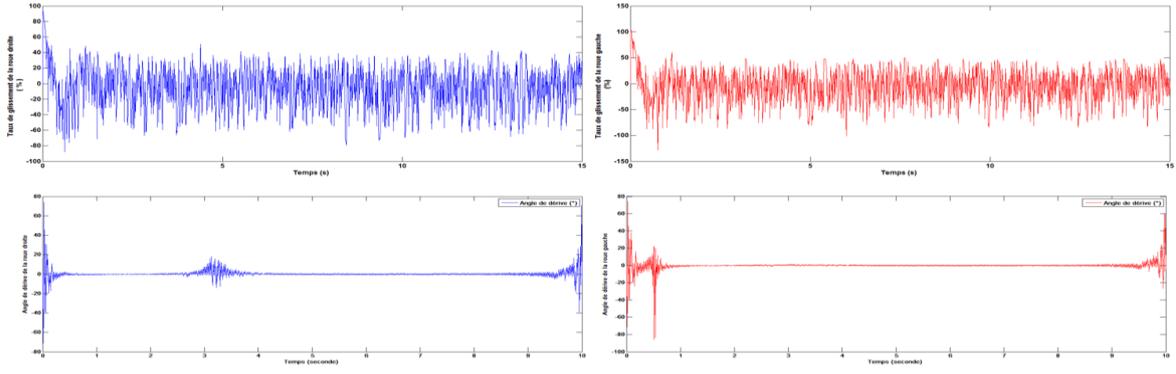


Figure IV.51 Les taux de glissement et les angles de dérive droite et gauche (surface (2))

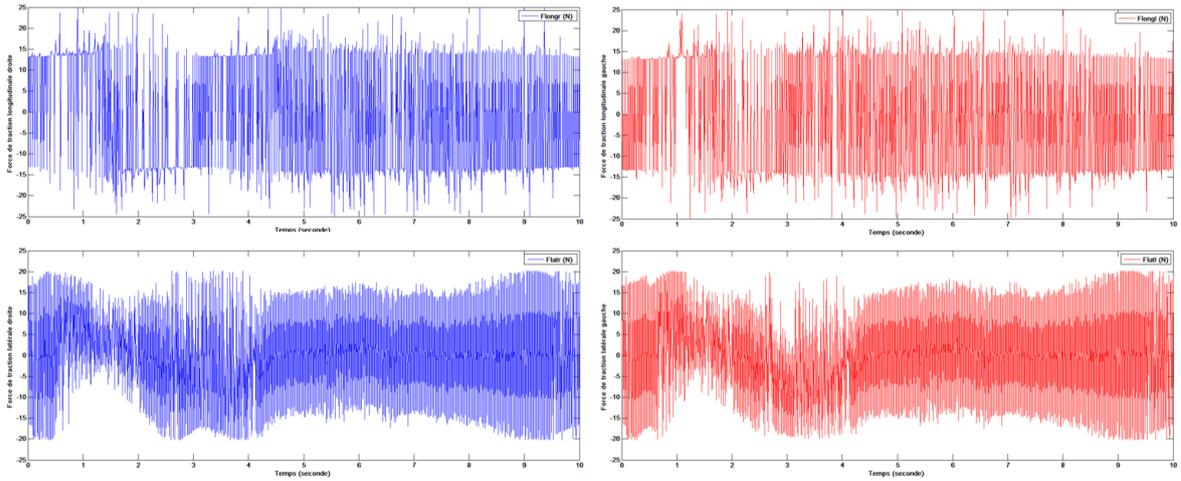


Figure IV.52 Les forces de traction longitudinale et latérales droite et gauche (surface (2))

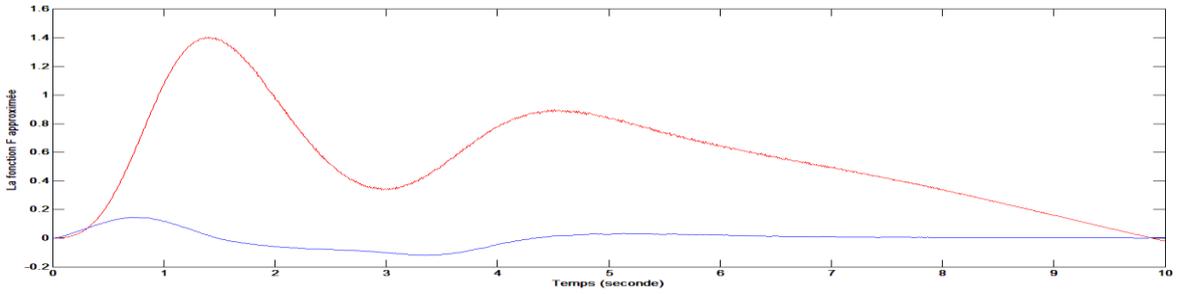


Figure IV.53 La fonction F approximée (surface (2))

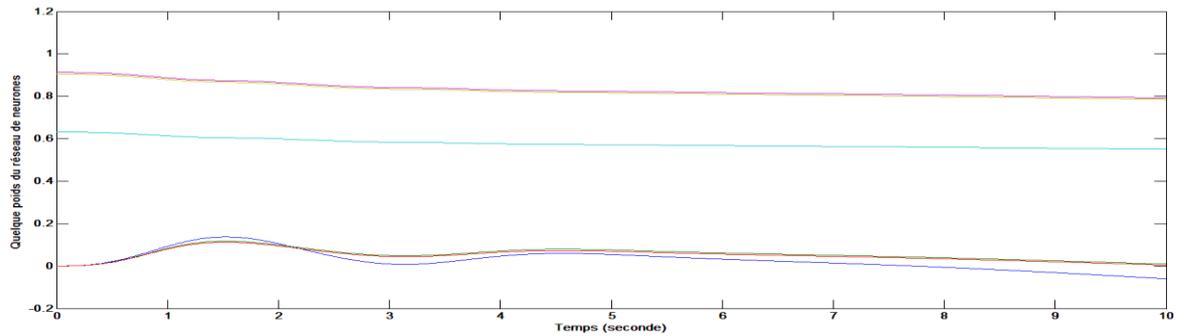


Figure IV.54 Quelques poids du réseau de neurones de la couche caché et de sortie (surface (2))

IV-2-2. TRAJECTOIRE CERCLE :

Suivi d'une trajectoire cercle cas de surface (1):

Condition initial : $x = 4$, $y = 0$, $\varphi = \pi/2$.

Gains du contrôleur cinématique : $k_x = 8$, $k_y = 5$, $k_\varphi = 3$.

Gains d'apprentissage du réseau de neurones : $F = 10$, $G = 10$

Gains du terme adaptatif : $k = 0.01$. Gain de terme robuste : $k_z = 1$, $w_{max} = 5$.

Gains du PD : $k_v = 40$, $k_w = 40$. Temps de simulation : 15s .

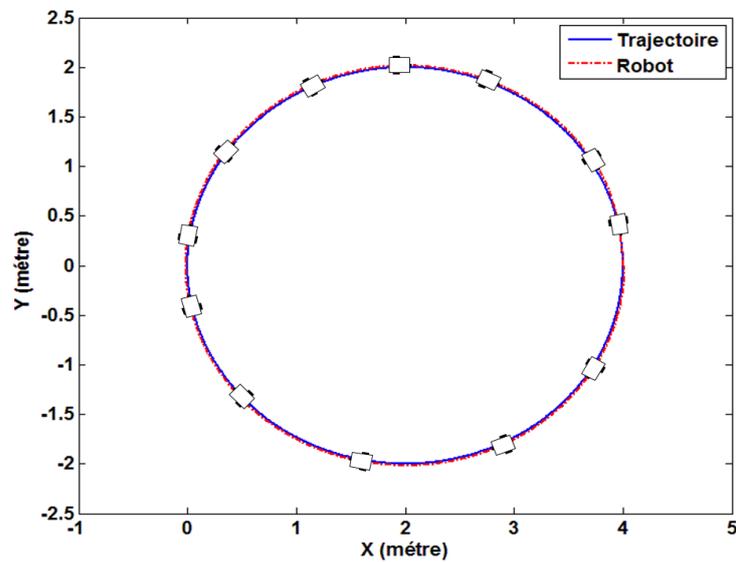


Figure IV.55 Suivi d'une trajectoire cercle (surface (1))

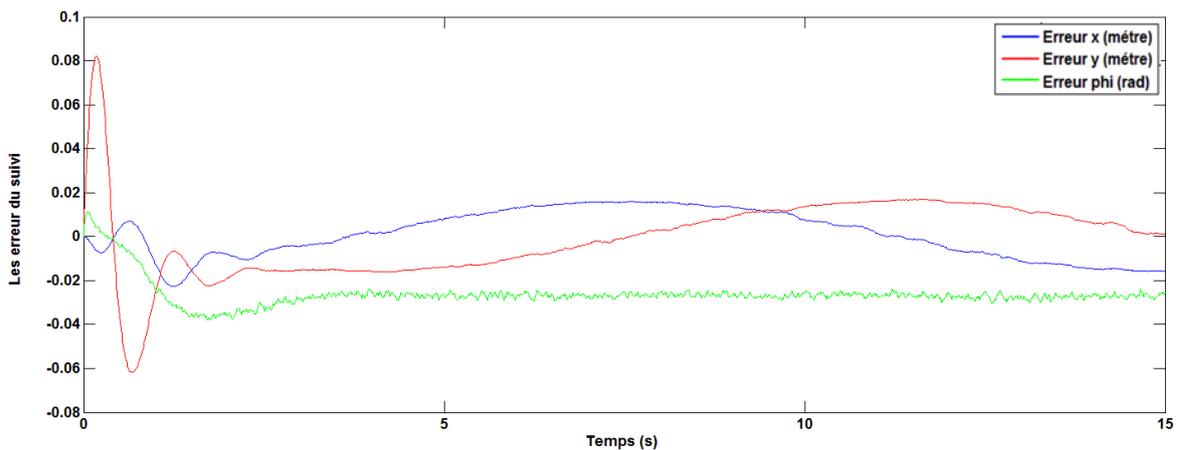


Figure IV.56 Les erreurs du suivi (surface (1))

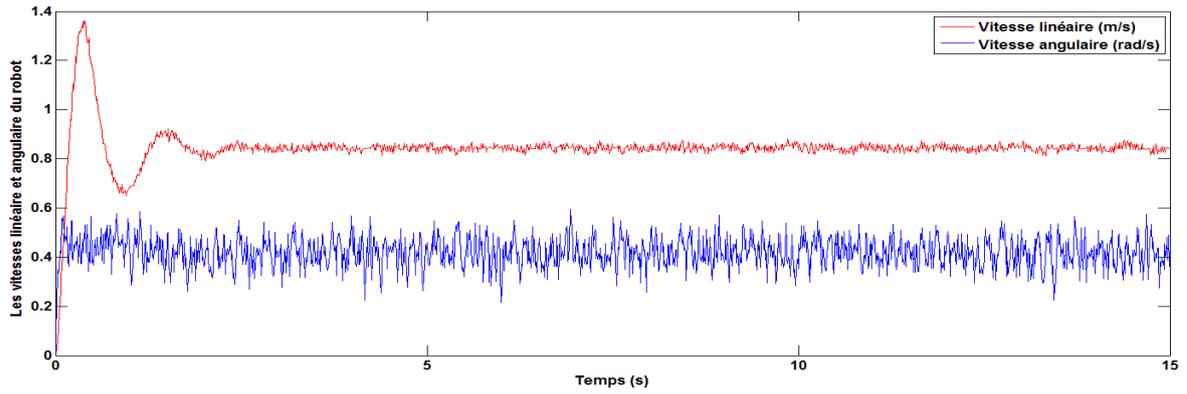


Figure IV.57 Les vitesses linéaire et angulaire du robot (surface (1))

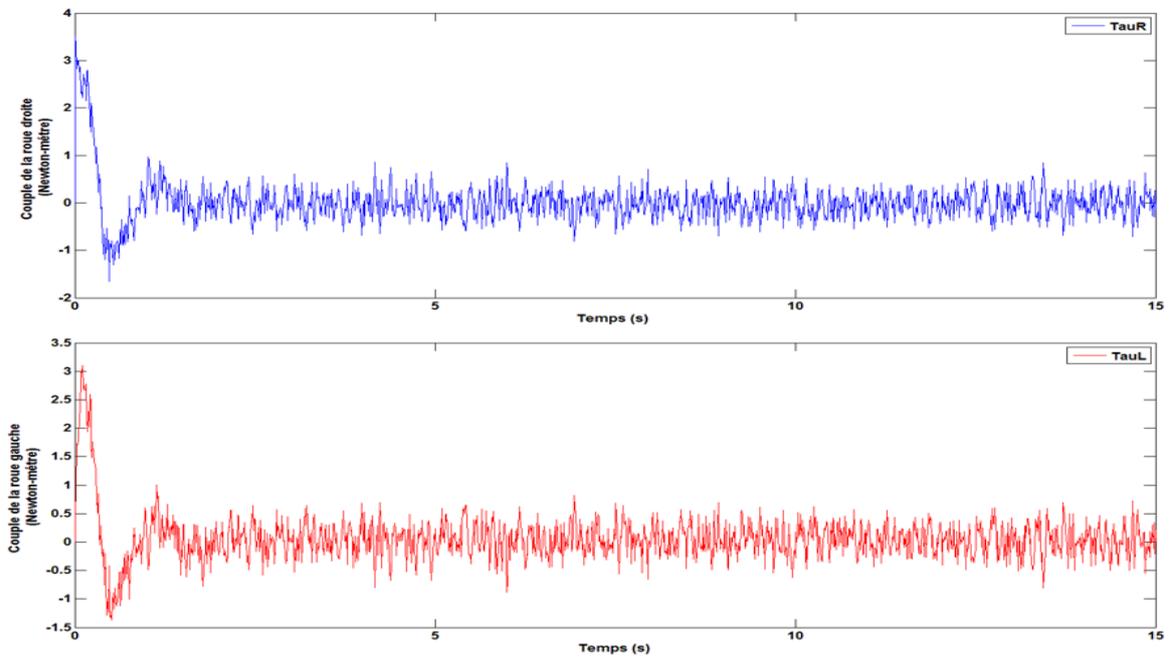


Figure IV.58 Les couples des roues droite et gauche (surface (1))

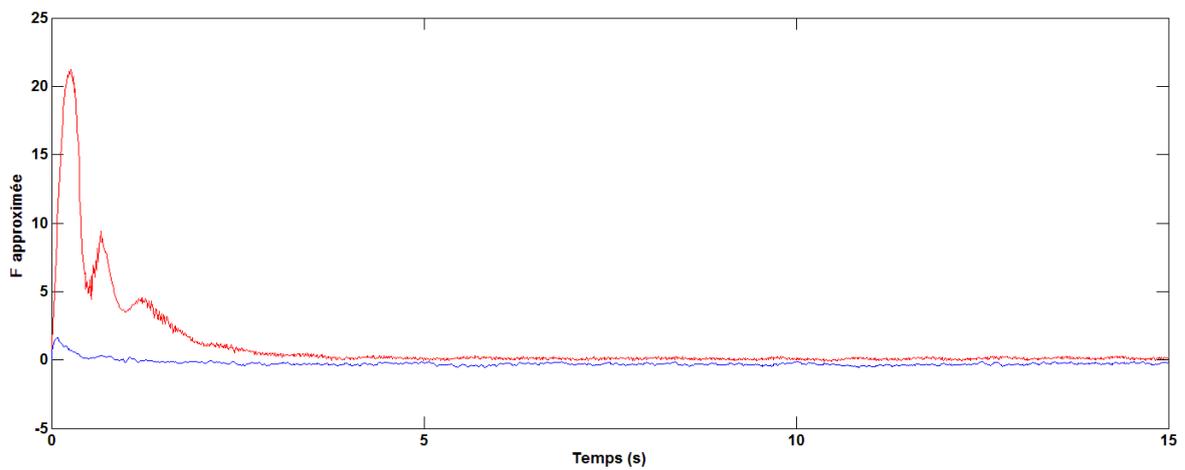


Figure IV.59 La fonction F approximée ((surface 1))

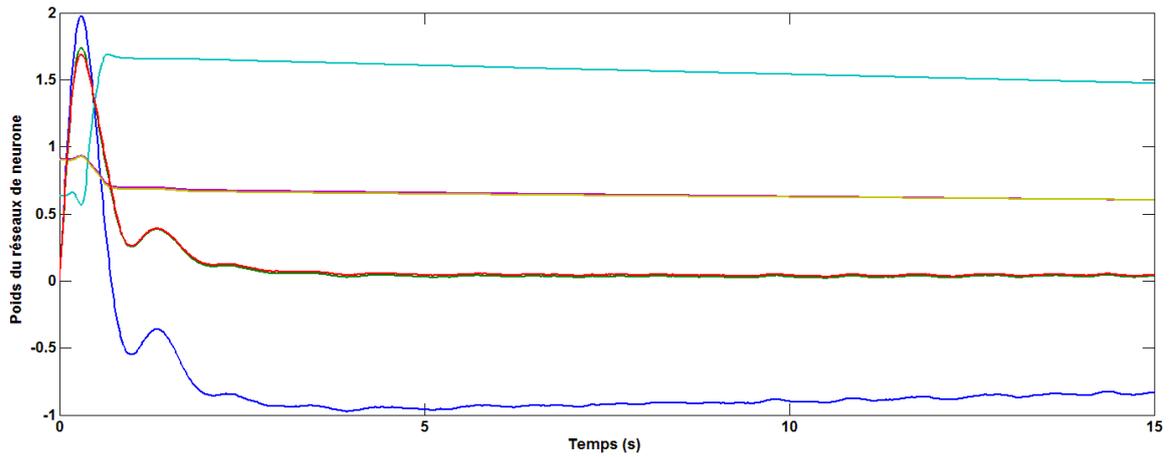


Figure IV.60 Quelques poids du réseau de neurones (surface (1))

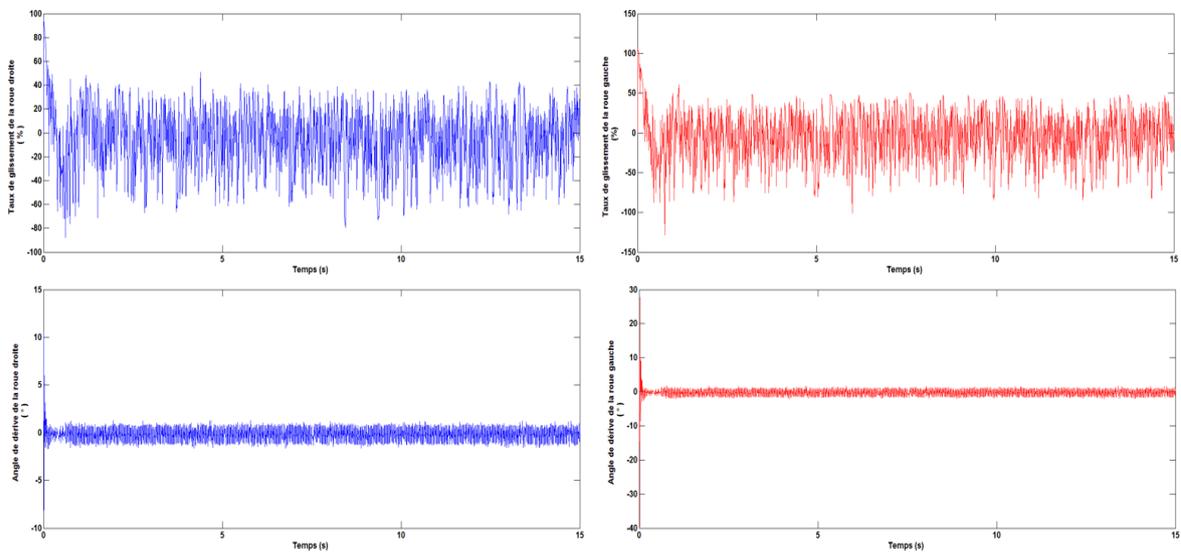


Figure IV.61 Les taux de glissement et les angles de dérive droite et gauche surface (1)

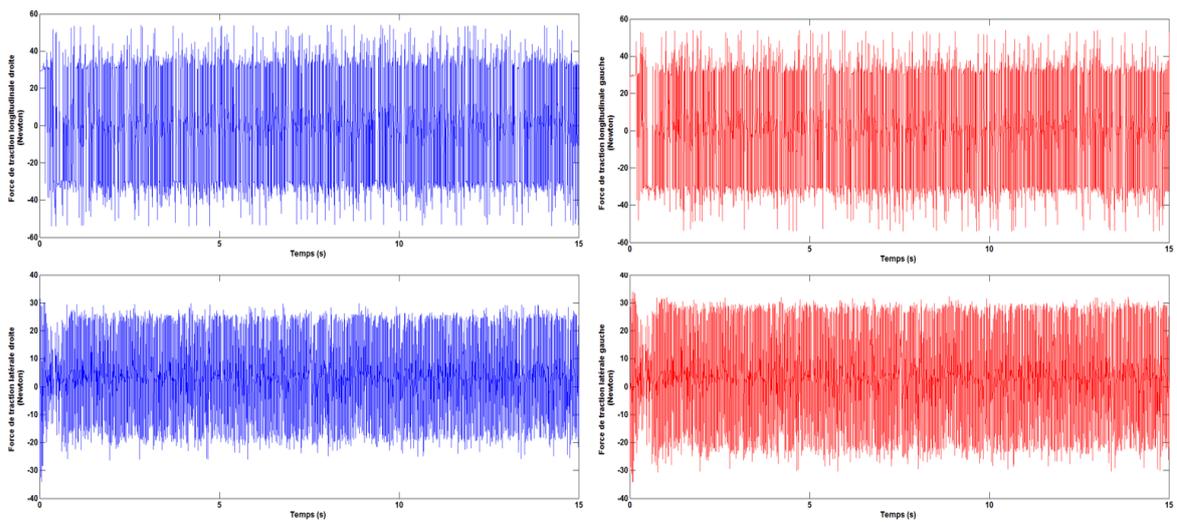


Figure IV.62 Les forces de traction longitudinales et latérale surface (1)

Il ressort que la réponse de la commande est mieux que celle de la commande Back-stepping. Les erreurs ont rectifiés presque en 2 seconde, avec des valeurs proche de zéro. Il est vu aussi que les poids du MLP tendent vers des valeurs stables tout en mouvement (en ligne).

Suivi d'une trajectoire cercle cas de surface (2):

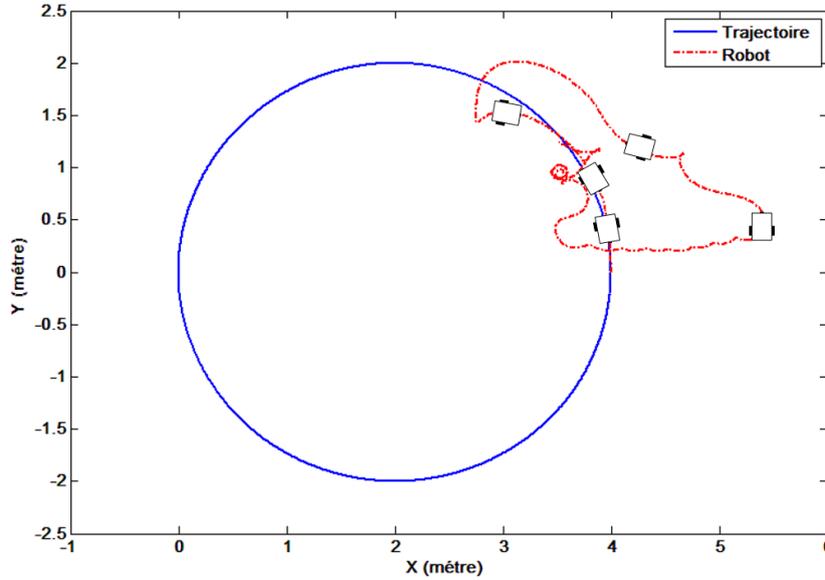


Figure IV.63 Suivi d'une trajectoire cercle (surface (2), $k_x=8, k_y=5, k_\phi=3$).

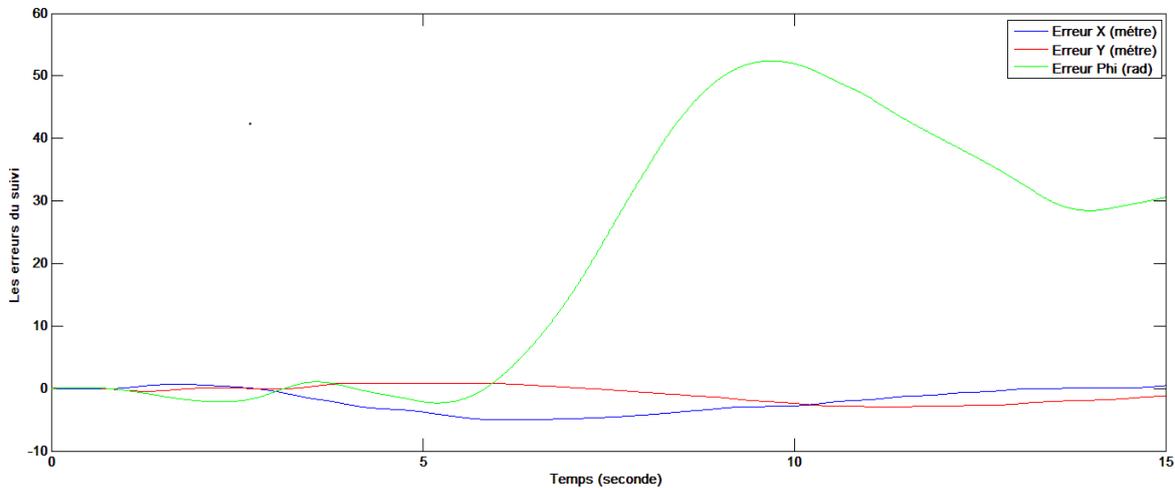


Figure IV.64 Les erreurs du suivi (surface (2), $k_x=8, k_y=5, k_\phi=3$).

Il est vu que la commande neuronale pour la trajectoire cercle surface (2) avec ces gains du contrôleur cinématique ne stabilise pas la poursuite. Car le glissement est lié directement à la vitesse du robot. Le choix de gains du contrôleur cinématique de façon à rendre la vitesse du robot immuable peu stabilisé le suivi et ne permis pas à la vitesse de passes un certain seuil, le seuil qui provoque le dérapage Fig IV.65 a la première seconde.

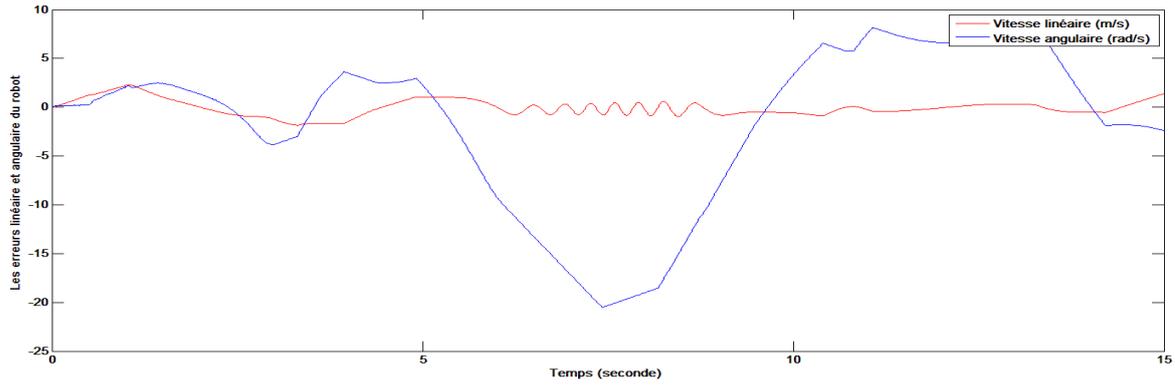


Figure IV.65 Les vitesses linéaire et angulaire du robot (surface (2))

Suivi d'une trajectoire cercle cas de surface (2) avec autre paramètre :

Gains du contrôleur cinématique : $k_x = 1.5, k_y = 1.5, k_\phi = 2$.

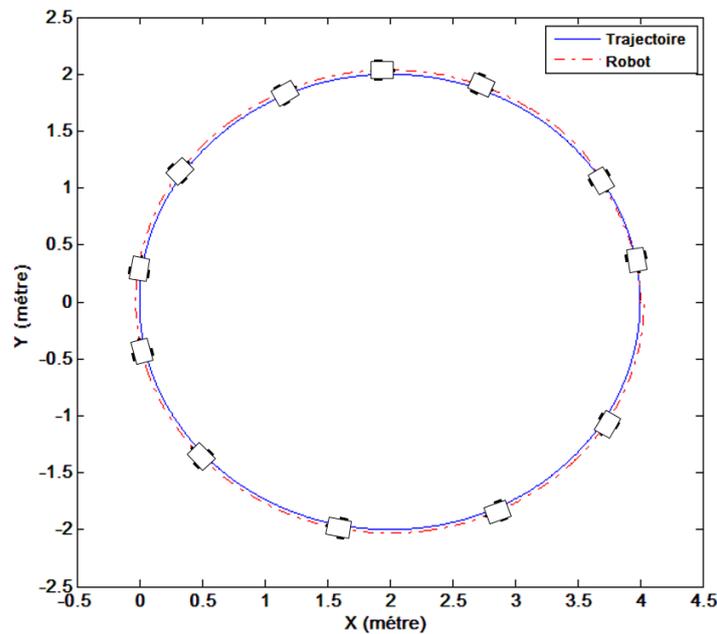


Figure IV.66 Suivi d'une trajectoire cercle (surface (2), $k_x=1.5, k_y=1.5, k_\phi=2$)

Vu de la figure Fig IV.66 que la poursuite utilisant les bons gains du contrôleur cinématique est stable. Si on compare la vitesse du robot dans les deux cas Fig IV.65 et Fig IV.68 on trouve que le contrôleur cinématique avec les bons gains a réussi de diminuer la vitesse dans la première seconde. La vitesse élevée qui vient de l'effort de correction des erreurs exercé par le contrôleur cinématique provoque le dérapage qui empêche la poursuite de la trajectoire. Cela peut être confirmé par la mise du point de départ loin de la courbe de trajectoire Fig IV.69, ce problème peut être résolu par un limiteur de vitesse placé entre le contrôleur cinématique et le contrôleur dynamique. La vitesse limite doit être plus grande que la vitesse de la trajectoire (la vitesse désirée), pour la Fig IV.69 la vitesse de la trajectoire est 0.8378 m/s et la limite 0.85 m/s .

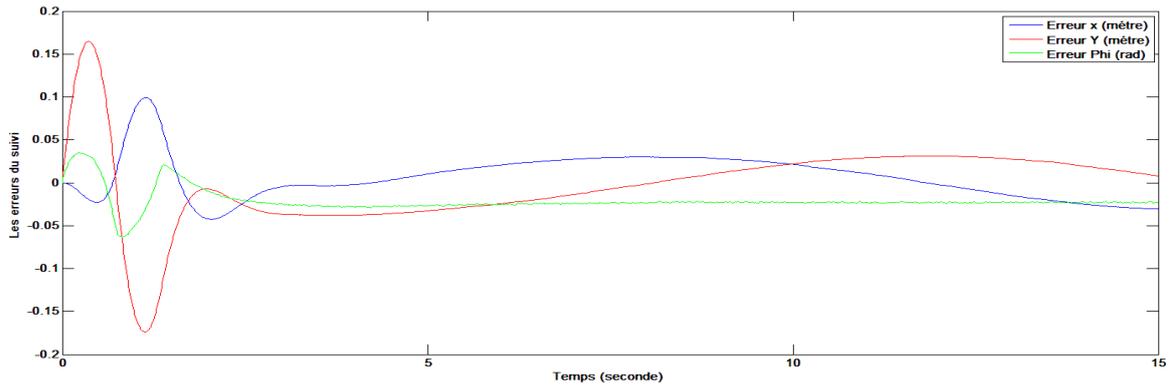


Figure IV.67 Les erreurs du suivi (surface (2), $k_x=1.5, k_y=1.5, k_\phi=2$)

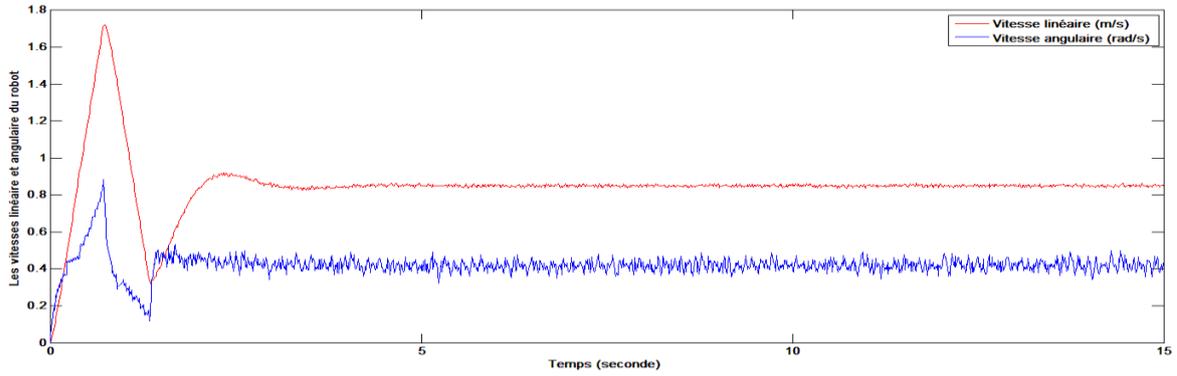


Figure IV.68 Les vitesses linéaire et angulaire du robot (surface (2), $k_x=1.5, k_y=1.5, k_\phi=2$).

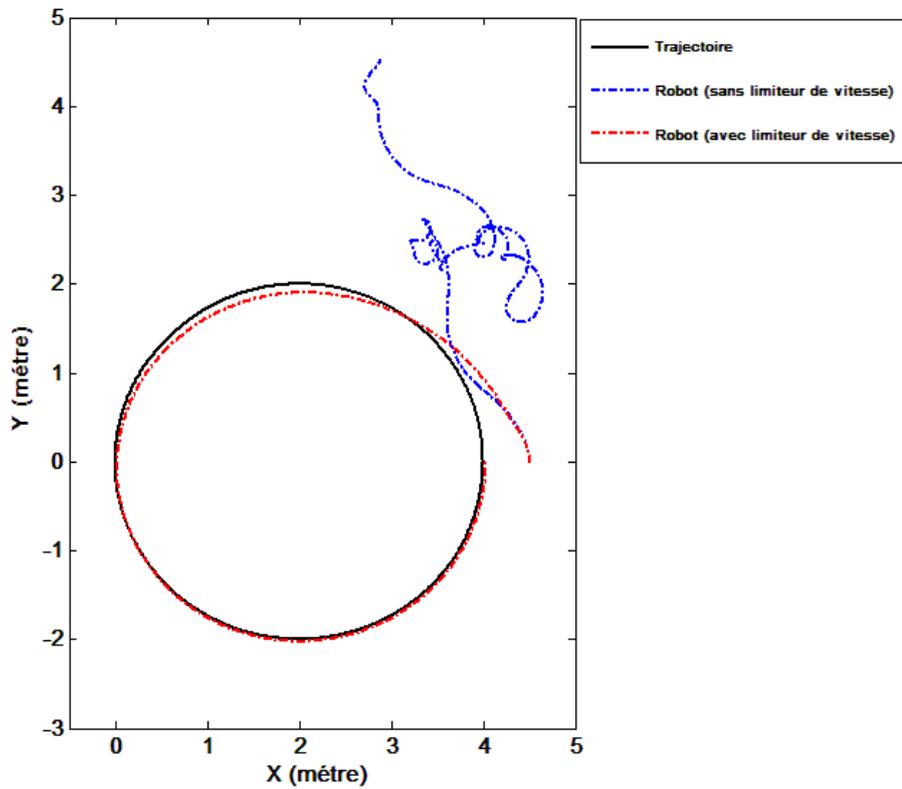


Figure IV.69 Comparaison de suivi d'une trajectoire cercle avec une posture de départ loin de la trajectoire
Condition initial : $x = 4.5, y = 0, \phi = \pi/2$. (avec et sans limiteur de vitesse)

IV-2-3. VIRAGE A 90° :

Suivi d'une trajectoire virage à 90° cas de surface (1) :

Condition initial : $x = 0$, $y = 0$, $\varphi = 0$.

Gains du contrôleur cinématique : $k_x = 1.5$, $k_y = 1.5$, $k_\varphi = 2$.

Gains d'apprentissage du réseau de neurones : $F = 10$, $G = 10$

Gains du terme adaptatif : $k = 0.01$. Gain de terme robuste : $k_z = 1$, $w_{max} = 5$.

Gains du PD : $k_v = 40$, $k_w = 40$. Temps de simulation : 10s .

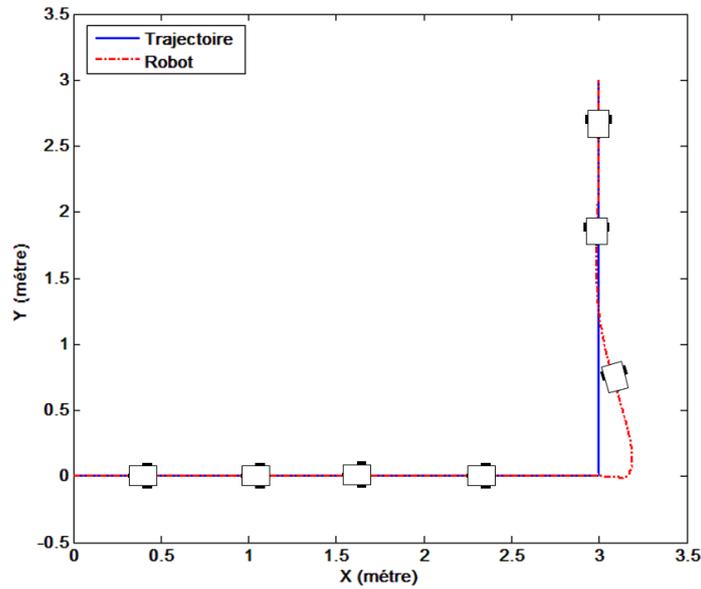


Figure IV.70 Suivi d'une trajectoire L (surface (1))

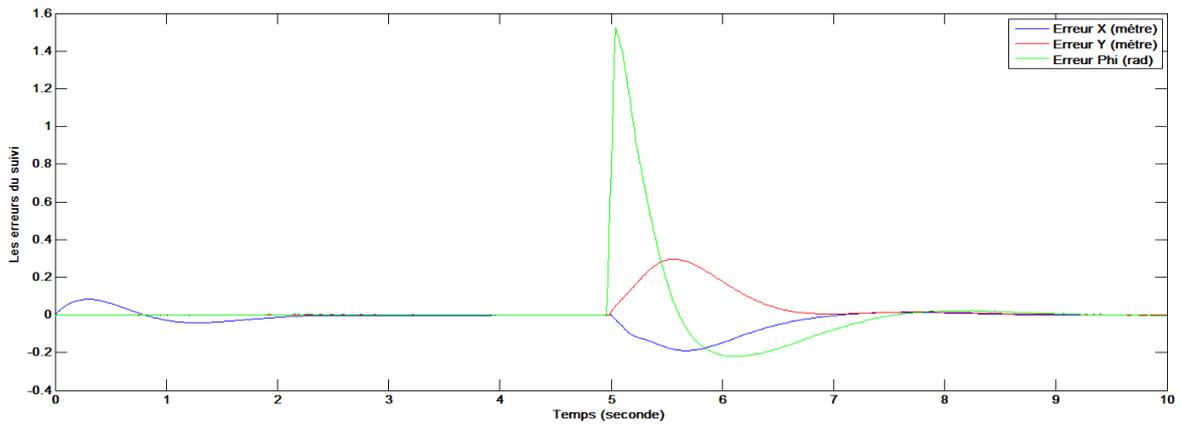


Figure IV.71 Les erreurs du suivi (surface (1))

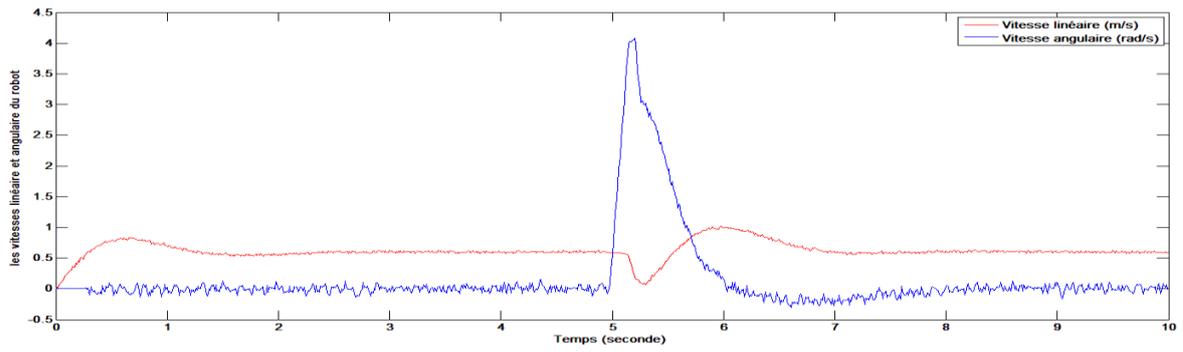


Figure IV.72 Les vitesses linéaire et angulaire du robot (surface 1))

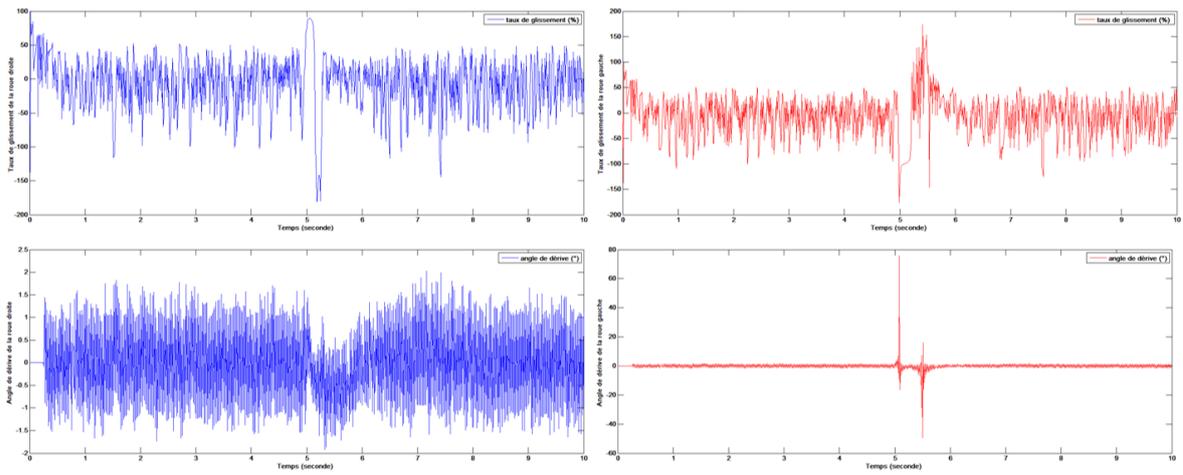


Figure IV.73 Les taux de glissement et les angles de dérive droite et gauche (surface (1))

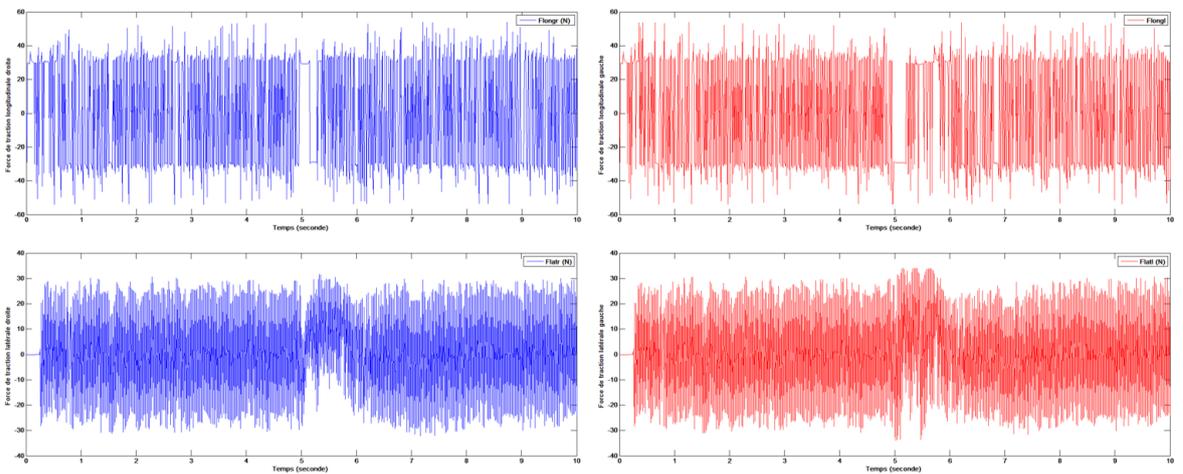


Figure IV.74 Les forces de traction longitudinales et latérales (surface (1))

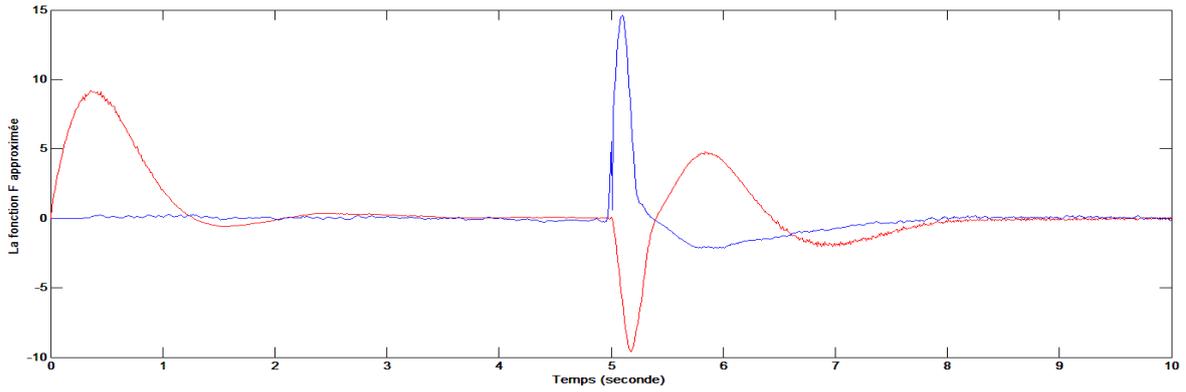


Figure IV.75 Fonction F approximée (surface (1))

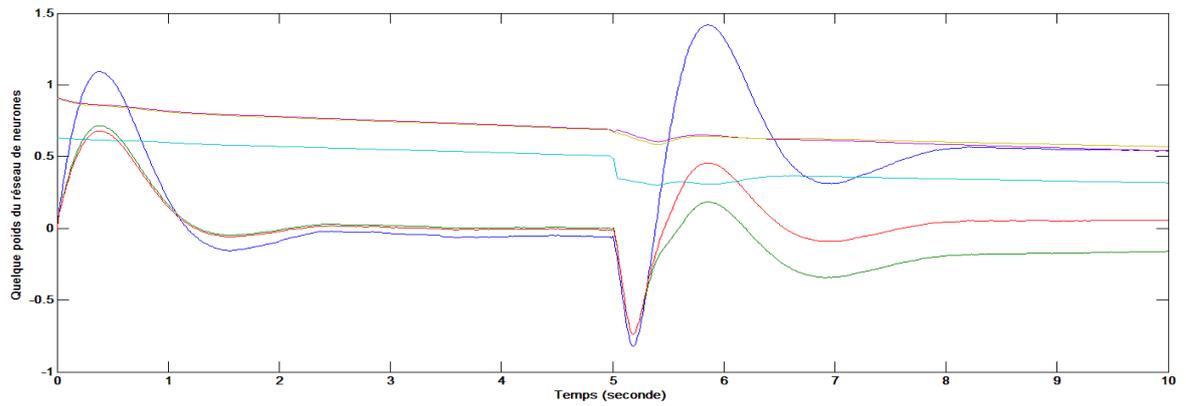


Figure IV.76 Quelques poids du réseau de neurones

La réponse de la commande neuronale pour la trajectoire L surface (1) a été mieux que celle de la commande backstepping. La rectification vers zéro des erreurs du suivi résultantes du virage a été rapide (moins de 2 sec), les figure Fig IV.75 et IV.76 montrent la stabilité du réseau de neurones.

Suivi d'une trajectoire virage à 90° cas de surface (2) :

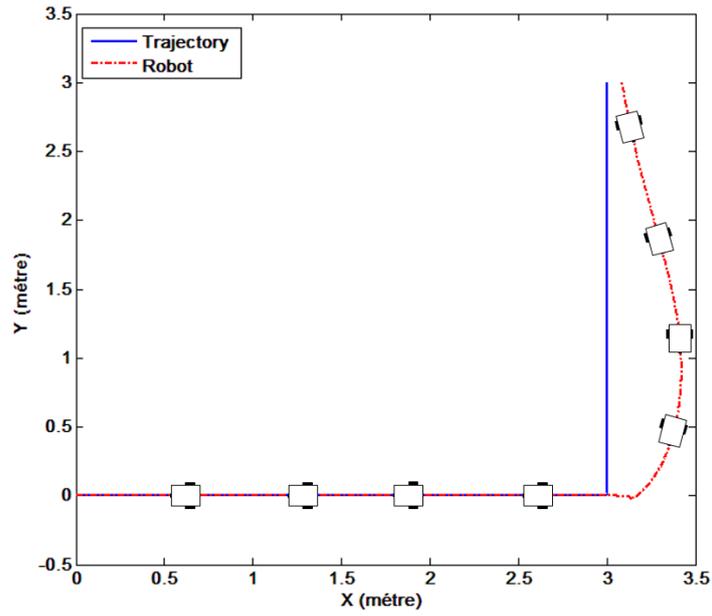


Figure IV.77 Suivi d'une trajectoire L surface (2)

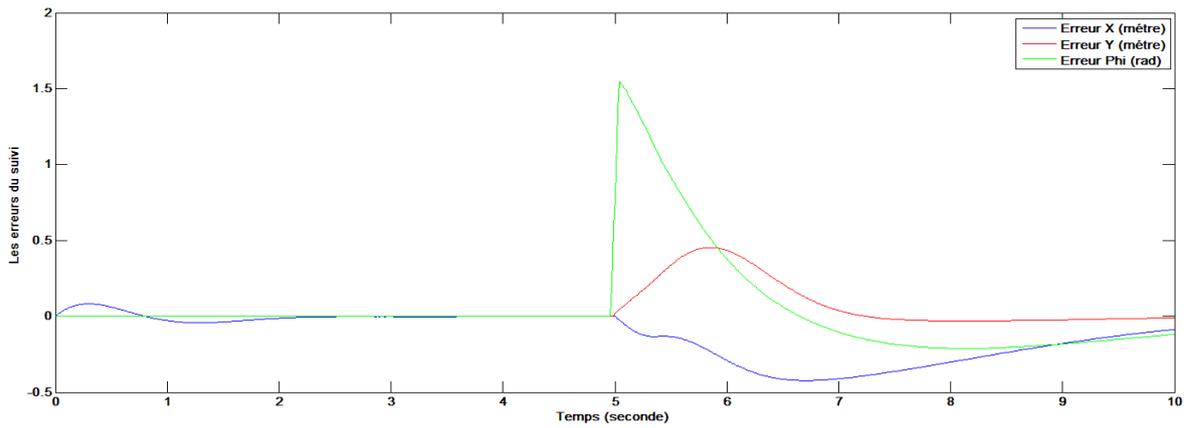


Figure IV.78 Les erreurs du suivi (surface (2))

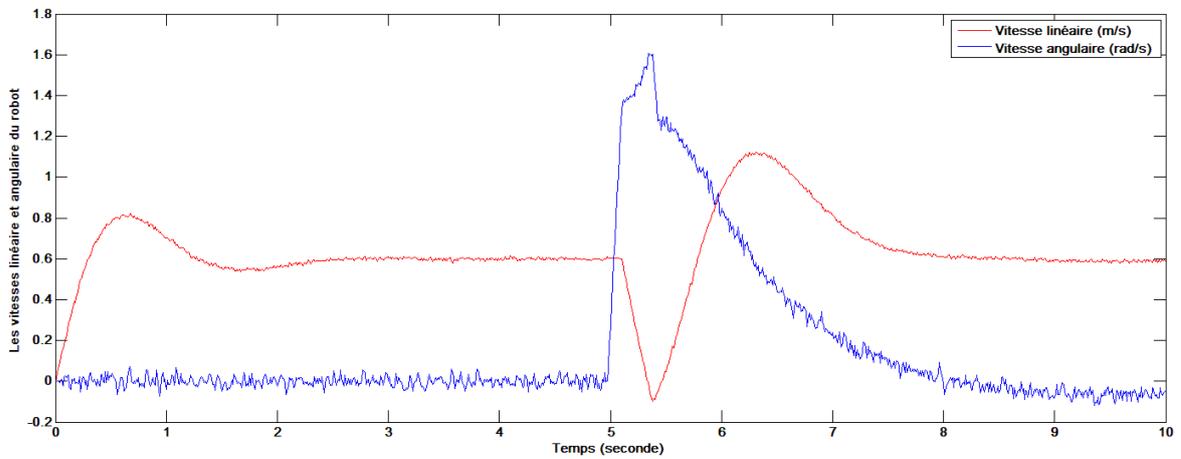


Figure IV.79 Les vitesses linéaire et angulaire du robot (surface (2))

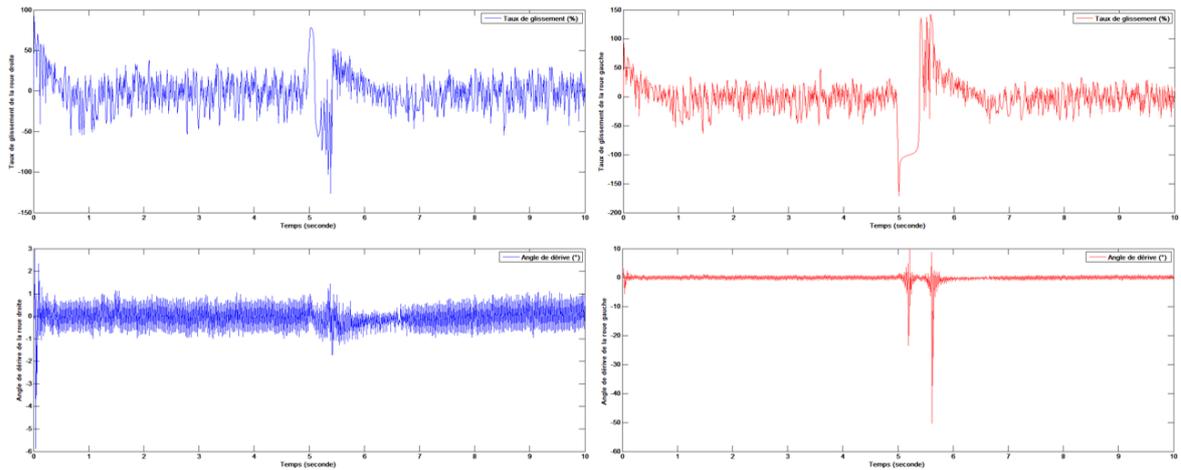


Figure IV.80 Les taux de glissement et les angles de dérive droite et gauche (surface (2))

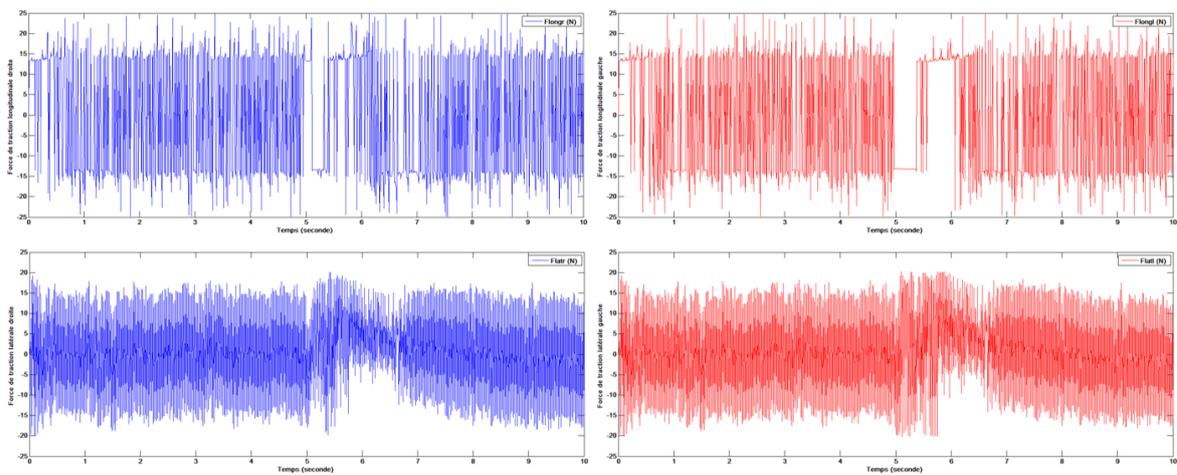


Figure IV.81 Les forces de traction longitudinales et latérales droite et gauche (surface (2))

Il est vu de la figure Fig IV.77 que la commande neuronale a réussi de stabiliser la poursuite même sur une surface glissante.

De ces simulation on voit que la commande neuronale est plus performante, capable de gérer la poursuite de n'importe quelle trajectoire avec glissement mais sa réussite dépend de la trajectoire elle-même. La poursuite de trajectoire est impossible dans le cas d'une trajectoire qui nécessite une grande vitesse et des lois de commande non lisses (non-smooth), l'approche appropriée dans ce cas est le suivi de chemin avec la limitation de vitesse du robot.

CONCLUSION :

Dans ce travail, nous avons étudié la modélisation et la commande d'un robot mobile à roues non-holonome « Pioneer 3dx ». Un modèle complet qui prend en considération l'effet de glissement des roues est présenté au chapitre 2, En suivant les étapes de modélisation mathématique communes dans la robotique : la modélisation géométrique, cinématique et dynamique. Des explications suffisamment explicites sur les contraintes non-holonomes, le glissement et les efforts de traction sont également présentés.

Par l'utilisation du modèle étudié, On essaye d'imiter le comportement d'un robot non-holonome de configuration à conduite différentielle dans des conditions presque réelles. Pour cela la modélisation d'interaction roues-chaussée a été une nécessité. Suivant le travail de Pacejka sur la pneumatique, nous avons réussi à simuler cette interaction en utilisant la Formule Magique. Nous avons présenté un modèle d'interaction capable de simuler le comportement d'un robot mobile à roues sur des surfaces glissantes et non-glissantes avec succès. Seulement à cause de la nature empirique de cette technique la détermination de ses paramètres qui convient notre robot a été impossible, en manquant des expérimentations sur les roues de « Pioneer 3dx ». Malheureusement c'est le drawback de ce travail.

La question de comment asservir un robot mobile pour accomplir acceptablement ses tâches sur des surfaces glissantes, nous a menée à chercher des techniques de contrôle capables d'adapter, approximer et gérer ce problème de glissement, qui est supposé inconnu et non structuré.

En basant sur leurs capacités d'approximation universelles et d'apprentissages, les réseaux de neurones nous donnent accès à un ensemble des techniques de commande adaptatives, capables d'asservir et identifier des systèmes incertains contenant des dynamiques inconnus. C'est la raison pour laquelle nous avons appliqué une commande neuronale afin de commander notre robot.

La commande neuronale étudiée est essentiellement basée sur une commande qui appartient à la famille des commandes Backstepping. Cette commande backstepping étudiée au chapitre trois est composée de deux contrôleurs : un contrôleur cinématique responsable de rectifier les erreurs cinématiques et de la posture. Et un contrôleur dynamique responsable de passer ces rectifications cinématique au robot sous forme de couples calculés. Seulement, la justesse du calcul des couples nécessite une connaissance complète de la dynamique du robot, cette connaissance est présumément indisponible. C'est ici où le réseau de neurones intervient et remplace le contrôleur dynamique incompetent. Le réseau de neurones estime

la dynamique du robot et calcul les couples nécessaires pour que le robot atteint la position désirée.

Cette technique qui apprend et adapte lorsque le mouvement du robot (on-line), nous a permis d'avoir un contrôle satisfaisant du robot sur des surfaces glissantes sans la moindre connaissance de la dynamique du robot ou des forces de traction qui perturbe le système. Cela est supporté par des simulations pour des variétés de trajectoires et les résultats sont présentés au chapitre quatre. L'inconvénient remarquable de cette technique est sa complexité et sa demande de grande capacité computationnelle.

Il est observé du travail de simulation que la tâche de suivi de trajectoire sur des surfaces glissantes est difficile à accomplir. Cette difficulté vient de la loi horaire associée à ce genre de tâches. Une fois que le robot perd sa position de poursuite à cause de dérapage, il est impossible pour la commande de rectifier les erreurs du suivi. Car le seul moyen est d'augmenter la vitesse pour récompenser le retard de la poursuite. Ce qui provoque une augmentation des taux de glissement par conséquent un autre dérapage est planifié.

Dans des cas pareils, la solution proposée pour remédier ce problème est l'utilisation d'un limiteur de vitesse qui limite les vitesses sortantes du contrôleur cinématique. Cette solution a été utilisée pour stabiliser la poursuite d'une trajectoire cercle dans le chapitre quatre. Une autre approche qui peut être redressé le problème est l'adaptation des gains du contrôleur cinématiques, on parle d'un contrôleur cinématique adaptatif, cela reste un perspective de ce travail. Il est conclu que pour une commande d'un robot mobile sur des surfaces glissantes, l'approche de suivi de chemin est la plus appropriée car elle n'est pas associée à une loi horaire ce qui laisse suffisamment de temps pour la limitation des vitesses.

La conception des trajectoires avec des vitesses modérées et la limitation des vitesses du robot restent les seuls moyens pour surmonter le problème de glissement pour les robots mobiles à roues. Mais la commande neuronale présentée dans ce travail à montrer une très grande performance et a pu stabiliser des suivis de trajectoires qui ne peuvent pas être stabilisées par d'autres commandes (Backstepping).

BIBLIOGRAPHIE:

- [1] Kolmanovsky, I., and McClamroch, N.H. “Developments in Nonholonomic Control Problems.” *Control Systems*, IEEE 15, no. 6 (December 1995): 20–36.
doi:10.1109/37.476384.
- [2] Brockett, R.W., Millman, R.S. and Sussmann, H.J. eds., 1983. *Differential geometric control theory: proceedings of the conference held at Michigan Technological University, June 28-July 2, 1982* (Vol. 27). Birkhauser.
- [3] Wang ZP, Ge SS, Lee TH, Lai XC. “Adaptive smart neural network tracking control of wheeled mobile robots”. In *Control, Automation, Robotics and Vision*, 2006. ICARCV'06. 9th International Conference on 2006 Dec 5 (pp. 1-6). IEEE.
- [4] Mohareri, O., 2009. *MOBILE ROBOT TRAJECTORY TRACKING USING NEURAL NETWORKS A Thesis In mechatronics* (Doctoral dissertation, American University of Sharjah).
- [5] Motte, I., & Campion, G. (2000). A slow manifold approach for the control of mobile robots not satisfying the kinematic constraints. *IEEE Transactions on Robotics and Automation*, 16(6), 875-880.
- [6] Lin, W. S., Chang, L. H., & Yang, P. C. (2007). Adaptive critic anti-slip control of wheeled autonomous robot. *IET control theory & applications*, 1(1), 51-57.
- [7] Tarokh, M., & McDermott, G. J. (2005). Kinematics modeling and analyses of articulated rovers. *Robotics, IEEE Transactions on*, 21(4), 539-553.
- [8] Dixon, W. E., Dawson, D. M., & Zergeroglu, E. (2000). Robust control of a mobile robot system with kinematic disturbances. In *Control Applications, 2000. Proceedings of the 2000 IEEE International Conference on* (pp. 437-442). IEEE.
- [9] Balakrishna, R., & Ghosal, A. (1995). Modeling of slip for wheeled mobile robots. *Robotics and Automation, IEEE Transactions on*, 11(1), 126-132.
- [10] Jung, S., & Hsia, T. C. (2005, August). Explicit lateral force control of an autonomous mobile robot with slip. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on* (pp. 388-393). IEEE.
- [11] Ploeg, J., Schouten, H. E., & Nijmeijer, H. (2008, June). Control design for a mobile robot including tire behavior. In *Intelligent Vehicles Symposium, 2008 IEEE* (pp. 240-245). IEEE.
- [12] Tian, Y., Sidek, N., & Sarkar, N. (2009, March). Modeling and control of a nonholonomic wheeled mobile robot with wheel slip dynamics. In *Computational Intelligence in Control and Automation, 2009. CICA 2009. IEEE Symposium on* (pp. 7-14). IEEE.

- [13] Sidek, N., & Sarkar, N. (2008, April). Dynamic modeling and control of nonholonomic mobile robot with lateral slip. In *Systems, 2008. ICONS 08. Third International Conference on* (pp. 35-40). IEEE.
- [14] Tian, Y. (2010). *Effect of wheel slip in modeling and control of wheeled mobile robots: applications to formation control and pursuit-evasion problems*(Doctoral dissertation).
- [15] Tian, Y., & Sarkar, N. (2014). Control of a mobile robot subject to wheel slip. *Journal of Intelligent & Robotic Systems*, 74(3-4), 915-929.
- [16] Adept mobile robots company site online:
http://www.mobilerobots.com/Mobile_Robots.aspx
- [17] Carole portillo & Frédéric klopcic. Cours : De la modélisation géométrique à l'implantation virtuelle de robots (Cours : De la modélisation géométrique à l'implantation virtuelle de robots)
<http://campus-douai.gemtech.fr/course/view.php?id=121>
- [18] Dhaouadi, R., & Hatab, A. A. (2013). Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework. *Advance in Robotics & Automation*, 2(2).
- [19] Nandy, S., Shome, S. N., Somani, R., Tanmay, T., Chakraborty, G., & Kumar, C. S. (2011, August). Detailed slip dynamics for nonholonomic mobile robotic system. In *Mechatronics and Automation (ICMA), 2011 International Conference on* (pp. 519-524). IEEE.
- [20] Tzafestas, S. G.. Introduction to Mobile Robot Control. Elsevier, 2014.
- [21] Sidek, S. N. (2008). *Dynamic modeling and control of nonholonomic wheeled mobile robot subjected to wheel slip* (Doctoral dissertation, Vanderbilt University).
- [22] Fu, T. (2000). *Modeling and performance analysis of ABS systems with nonlinear control* (Doctoral dissertation, Concordia University).
- [23] Thuilot, B. (1995). *Contribution à la modélisation et à la commande de robots mobiles à roues*(Doctoral dissertation, École Nationale Supérieure des Mines de Paris).
- [24] Li, L., & Wang, F. Y. (2007). *Advanced motion control and sensing for intelligent vehicles*. Springer Science & Business Media.
- [25] Wellstead, P. E., & Pettit, N. B. O. L. (1997, September). Analysis and redesign of an antilock brake system controller. In *Control Theory and Applications, IEE Proceedings-* (Vol. 144, No. 5, pp. 413-426). IET.
- [26] Schinkel, M., & Hunt, K. (2002). Anti-lock braking control using a sliding mode like approach. In *American Control Conference, 2002. Proceedings of the 2002* (Vol. 3, pp. 2386-2391). IEEE.
- [27] Burckhardt, M. (1993). *Fahrwerktechnik: Radschlupf-Regelsysteme*. Vogel-Verlag, Germany, 16.

- [28] G. Rill, Simulation von Kraftfahrzeugen. Wiesbaden, Germany: Vieweg, 1994.
- [29] Bakker, E., Nyborg, L., & Pacejka, H. B. (1987). *Tyre modelling for use in vehicle dynamics studies*(No. 870421). SAE Technical Paper.
- [30] Bakker, E., Pacejka, H. B., & Lidner, L. (1989). *A new tire model with an application in vehicle dynamics studies* (No. 890087). SAE Technical Paper.
- [31] Pacejka, H. B., & Sharp, R. S. (1991). Shear force development by pneumatic tyres in steady state conditions: a review of modelling aspects. *Vehicle system dynamics*, 20(3-4), 121-175.
- [32] Dahl, P. R. (1968). *A solid friction model* (No. TOR-0158 (3107-18)-1). AEROSPACE CORP EL SEGUNDO CA.
- [33] Dahl, P. R. (1977). *Measurement of solid friction parameters of ball bearings* (No. TR-0077 (2901-03)-3). AEROSPACE CORP EL SEGUNDO CA ENGINEERING SCIENCE OPERATIONS.
- [34] Bliman, P. A., & Sorine, M. (1993). Friction modeling by hysteresis operators. Application to Dahl, sticktion and Stribeck effects. *Pitman Research Notes in Mathematics Series*, 10-10.
- [35] Bliman, P. A., & Sorine, M. (1993, September). A system-theoretic approach of systems with hysteresis. application to friction modelling and compensation. In *Proceedings of the 2nd European control conference* (pp. 1844-1849).
- [36] De Wit, C. C., Olsson, H., Astrom, K. J., & Lischinsky, P. (1995). A new model for control of systems with friction. *Automatic Control, IEEE Transactions on*, 40(3), 419-425.
- [37] De Wit, C. C., Tsiotras, P., Velenis, E., Basset, M., & Gissingner, G. (2003). Dynamic Tire Friction Models for Vehicle Traction/Braking Control. *Vehicle System Dynamics*, 39(3), 189-226.
- [38] Zegelaar, P. W. A., & Pacejka, H. B. (1996). The in-plane dynamics of tyres on uneven roads. *Vehicle System Dynamics*, 25(S1), 714-730.
- [39] Denti, E., & Fanteria, D. (2000). Models of wheel contact dynamics: an analytical study on the in-plane transient responses of a brush model. *Vehicle System Dynamics*, 34(3), 199-225.
- [40] Breuer, B., Eichhorn, U., & Roth, J. (1992, September). Measurement of tyre/road-friction ahead of the car and inside the tyre. In *International Symposium on Advanced Vehicle Control, 1992, Yokohama, Japan*.
- [41] Qu, Q., & Liu, Y. (2000). On lateral dynamics of vehicles based on nonlinear characteristics of tires. *Vehicle system dynamics*, 34(2), 131-141.
- [42] Hahn, J. O., Rajamani, R., & Alexander, L. (2002). GPS-based real-time identification of tire-road friction coefficient. *Control Systems Technology, IEEE Transactions on*, 10(3), 331-343.

- [43] Breuer, B., Eichhorn, U., & Roth, J. (1992, September). Measurement of tyre/road-friction ahead of the car and inside the tyre. In *International Symposium on Advanced Vehicle Control, 1992, Yokohama, Japan*.
- [44] Ono, E., Hosoe, S., & Tuan, H. D. (1998). Bifurcation in vehicle dynamics and robust front wheel steering control. *Control Systems Technology, IEEE Transactions on*, 6(3), 412-420.
- [45] Li, L., Wang, F. Y., & Zhou, Q. (2006). Integrated longitudinal and lateral tire/road friction modeling and monitoring for vehicle motion control. *Intelligent Transportation Systems, IEEE Transactions on*, 7(1), 1-19.
- [46] Pacejka, H. B., & Bakker, E. (1991). The Magic Formula Tyre Model. Tyre models for vehicle dynamic analysis: Proceedings of the 1st International Colloquium on Tyre Models for Vehicle Dynamics Analysis. *Delft, the Netherlands*.
- [47] Pacejka, H. B., & Besselink, I. J. M. (1997). Magic formula tyre model with transient properties. *Vehicle system dynamics*, 27(S1), 234-249.
- [48] Hans Pacejka . Tyre and Vehicle Dynamics .seconde edition, Elsevier, 2006.
- [49] De Luca, A., Oriolo, G., & Vendittelli, M. (2001). Control of wheeled mobile robots: An experimental overview. In *Ramsete* (pp. 181-226). Springer Berlin Heidelberg.
- [50] Kanayama, Y., Kimura, Y., Miyazaki, F., & Noguchi, T. (1991, November). A stable tracking control method for a non-holonomic mobile robot. In *Intelligent Robots and Systems' 91. Intelligence for Mechanical Systems, Proceedings IROS'91. IEEE/RSJ International Workshop on* (pp. 1236-1241). IEEE.
- [51] Gupta, M., Jin, L., & Homma, N. (2004). *Static and dynamic neural networks: from fundamentals to advanced theory*. John Wiley & Sons.
- [52] Haykin, S., & Lippmann, R. (1994). Neural Networks, A Comprehensive Foundation. *International Journal of Neural Systems*, 5(4), 363-364.
- [53] Omatu, S., Khalid, M. B., & Yusof, R. (2012). *Neuro-control and its applications*. Springer Science & Business Media.
- [54] Omatu, S. (2009). Neuro-control and its applications to electric vehicle control. In *Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living* (pp. 1-12). Springer Berlin Heidelberg.
- [55] Psaltis, D., Sideris, A., & Yamamura, A. (1988). A multilayered neural network controller. *IEEE control systems magazine*, 8(2), 17-21.
- [56] Hunt, K. J., & Sbarbaro, D. (1991, September). Neural networks for nonlinear internal model control. In *Control Theory and Applications, IEE Proceedings D* (Vol. 138, No. 5, pp. 431-438). IET.

- [57] Fierro, R., & Lewis, F. L. (1995, December). Control of a nonholonomic mobile robot: backstepping kinematics into dynamics. In *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on* (Vol. 4, pp. 3805-3810). IEEE.
- [58] Fierro, R., & Lewis, F. L. (1998). Control of a nonholonomic mobile robot using neural networks. *Neural Networks, IEEE Transactions on*, 9(4), 589-600.
- [59] Lewis, F. L., Campos, J., & Selmic, R. (2002). *Neuro-fuzzy control of industrial systems with actuator nonlinearities* (Vol. 24). Siam.
- [60] K. S. Narendra, "Adaptive control using neural networks," in *Neural Networks for Control*, W. T. Miller, R. S. Sutton, and P. W. Werbos, Eds. Cambridge, MA: MIT Press, 1991, pp. 115–142.