



HAL
open science

Stabilité globale pour la navigation réactive d'un robot mobile en présence d'obstacles

Ahmed Benzerrouk, Lounis Adouane, Zaher Al Barakeh, Philippe Martinet

► To cite this version:

Ahmed Benzerrouk, Lounis Adouane, Zaher Al Barakeh, Philippe Martinet. Stabilité globale pour la navigation réactive d'un robot mobile en présence d'obstacles. CIFA 2010, Sixième Conference Internationale Francophone d'Automatique, Jun 2010, Nancy, France. hal-01714862

HAL Id: hal-01714862

<https://hal.science/hal-01714862v1>

Submitted on 23 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Stabilité globale pour la navigation réactive d'un robot mobile en présence d'obstacles

Ahmed BENZERROUK, Lounis ADOUANE, Zaher AL-BARAKEH, Philippe MARTINET

Laboratoire des Sciences et Matériaux pour l'Electronique et d'Automatique
UMR 6602 – Université Blaise Pascal, CNRS
24, Avenue des landais. 63177 Aubière, France.
prenom.nom@lasmea.univ-bpclermont.fr

Résumé— Ce papier traite la navigation d'un robot mobile dans un environnement en présence d'obstacles. Le robot doit alors atteindre une cible finale tout en évitant des obstacles. Il est proposé de briser la complexité de la tâche à réaliser en la décomposant en un ensemble de tâches élémentaires : Attraction vers une cible et évitement d'obstacles. Chacune de ces tâches est accomplie grâce à un contrôleur élémentaire dédié. L'activation d'un contrôleur à la faveur d'un autre se fera en fonction de la tâche prioritaire à réaliser. Pour assurer la stabilité globale du système notamment aux moments des transitions entre contrôleurs, les propriétés des systèmes hybrides permettant le passage d'un système continu à un autre en présence d'évènements discrets sont exploitées. Ici, il est proposé d'agir sur le gain de la loi de commande élaborée afin d'assurer la stabilité de l'architecture globale, et ce même aux instants des commutations. La stabilité est étudiée grâce à une fonction de Lyapunov commune à tous les contrôleurs. Des résultats de simulations et expérimentaux appuieront les résultats théoriques.

Mots-clés— Robot mobile, architecture de contrôle hybride, stabilité au sens de Lyapunov.

I. INTRODUCTION

Le contrôle d'un robot mobile navigant dans un environnement encombré est un problème fondamental et fait l'objet d'une grande attention de la communauté robotique. Il s'agit principalement de garantir au robot mobile un contrôle fiable lui permettant une navigation en toute sécurité (loin de tout risque de collision, respect des contraintes structurelles du robot : non holonomie, accélération maximale, etc.)

Une partie de la littérature considère que la navigation du robot se base entièrement sur les méthodes de planification de chemin qui impliquent alors la connaissance totale ou partielle de l'environnement dans lequel il évolue : les diagrammes de Voronoï et les graphes de visibilité [1] ou alors les fonctions de potentiel artificielles comprenant toutes les informations sur la cible à atteindre et l'environnement [2] sont parmi ces méthodes.

Une autre partie s'intéresse, contrairement à la précédente, plus particulièrement à la capacité du robot à répondre aux lois de commande en fonction de ses contraintes (contraintes structurelles, à-coups de commande, etc.). Même si on retrouve ici des méthodes cognitives de planification et de re-planification de chemin [3], [4], d'autres méthodes plus réactives (se basant sur les informations émanant des capteurs du robot plutôt que sur la connaissance préalable de l'environnement externe) sont plus courantes [5], [6], ou [7]. Les travaux proposés dans ce papier s'inscrivent plutôt dans cette dernière approche.

Pour garantir la capacité du robot à accomplir une tâche

de façon réactive, on propose de s'inspirer des architectures de contrôle comportementales proposées initialement par Brooks [8] afin de briser la complexité de la tâche globale à réaliser. Cette dernière est alors divisée en plusieurs tâches élémentaires avec des contrôleurs correspondant. La tâche globale est accomplie en coordonnant ces tâches/contrôleurs élémentaires. Il existe deux principes majeurs de coordination : la sélection d'action [8] et la fusion d'actions [9]. Dans le premier, un seul contrôleur sélectionné parmi les contrôleurs élémentaires est appliqué à chaque instant, tandis que dans le deuxième, la commande appliquée au robot résulte d'une fusion de tout ou une partie des contrôleurs disponibles dans l'architecture de contrôle.

On remarque que le cas de la sélection d'action est plus intéressant. En effet, un seul contrôleur est appliqué au robot mobile à un instant donné. Il est alors plus simple d'accéder et d'étudier individuellement la stabilité de chaque contrôleur. Cependant, passer d'un contrôleur à un autre en fonction de la tâche élémentaire à accomplir (éviter un obstacle, suivre une trajectoire, atteindre une cible, etc.) peut engendrer l'instabilité de la loi de commande globale si la commutation "switch" se fait de façon aléatoire, et ce même si chaque contrôleur est individuellement stable [10].

Ce champ d'études (preuve de la stabilité d'une architecture à commutations entre contrôleurs) a été peu exploré dans la littérature : on trouve une étude réalisée dans [5]. Par contre, dans ce papier, afin de faire face aux commutations indésirables entre contrôleurs, les auteurs introduisent une fusion d'actions permettant de passer en douceur d'un contrôleur à un autre. L'avantage d'étudier chaque contrôleur séparément a alors été perdu, car il faut en plus étudier la stabilité de la fusion d'actions des deux contrôleurs. Un contrôle d'un robot mobile suivant une trajectoire en présence d'obstacles, et se basant sur le théorème des fonctions multiples de Lyapunov [10] a été établi dans [11] : Un contrôleur secondaire a alors été introduit afin de satisfaire ce théorème. Cependant, cette architecture de contrôle n'est pas adaptée à n'importe quel environnement encombré. Nous proposons ici d'explorer le théorème de la Fonction de Lyapunov Commune FLC [12] utilisé dans le cadre des systèmes hybrides afin de soumettre les transitions entre les systèmes continus à des contraintes. Ces contraintes permettent de garantir la stabilité globale du système. Dans notre cas, le système global est l'architecture de contrôle proposée.

Trouver une fonction de Lyapunov commune à tous les

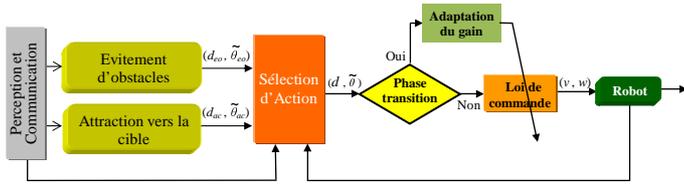


Fig. 1. Architecture de contrôle proposée avec un bloc d'ajustement du gain k .

systèmes élémentaires qui forment un système hybride est une tâche connue comme étant difficile [12]. Dans ce papier, on propose de résoudre ce problème dans le cadre de la robotique en garantissant la stabilité globale de l'architecture de contrôle proposée.

On s'intéresse ici à la tâche d'atteindre une cible tout en évitant les obstacles de l'environnement : cette tâche est alors divisée en deux tâches/contrôleurs élémentaires : attraction vers une cible et évitement d'obstacles.

La suite du papier est organisé comme suit : dans la section suivante, l'architecture de contrôle proposée est introduite. Sa stabilité basée sur le théorème de la fonction de Lyapunov commune est étudiée dans la section III. Les résultats de simulation sont donnés dans IV. L'architecture de contrôle proposée est implémentée par la suite sur un robot Khepera III. Les résultats se trouvent dans la section V. Enfin, la section VI comprend une conclusion et quelques perspectives.

II. ARCHITECTURE DE CONTRÔLE PROPOSÉE

L'architecture de contrôle proposée est donnée dans la figure (1).

Il est proposé dans ce qui suit de détailler les blocs de cette architecture en commençant par les contrôleurs élémentaires : Attraction vers la cible, et Evitement d'obstacles. Cependant, on rappelle d'abord que le robot utilisé est un robot unicycle dont le modèle cinématique s'exprime dans un repère absolu par le système d'équations d'états suivant :

$$\begin{aligned} \dot{x} &= v \cos(\theta) \\ \dot{y} &= v \sin(\theta) \\ \dot{\theta} &= \omega \end{aligned} \quad (1)$$

Où

- x, y sont les coordonnées du robot dans le repère absolu,
- θ est l'orientation du robot dans ce repère,
- v et ω sont les vitesses linéaire et angulaire du robot respectivement.

A. Contrôleur d'attraction vers la cible

Le robot doit atteindre une cible donnée de rayon R_c et de centre (x_c, y_c) . La figure (2) permet de définir les erreurs de position suivantes :

$$\begin{aligned} e_x &= (x_c - x) = d_{ac} \cos(\theta_r) \\ e_y &= (y_c - y) = d_{ac} \sin(\theta_r) \end{aligned} \quad (2)$$

Leurs dérivées s'écrivent

$$\begin{aligned} \dot{e}_x &= -\dot{x} = -v \cos(\theta) \\ \dot{e}_y &= -\dot{y} = -v \sin(\theta) \end{aligned} \quad (3)$$

où d_{ac} est la distance euclidienne séparant le robot au centre de la cible. Il s'agit du premier paramètre générée par ce contrôleur (cf. Figure 1) afin d'être utilisé par la loi de commande (cf. Section II-C). Cette distance est calculée par

$$d_{ac} = \sqrt{e_x^2 + e_y^2} \quad (4)$$

De la même façon, on définit $\tilde{\theta}_{ac}$ qui est l'erreur d'orientation du robot par rapport à la cible telle que : $\tilde{\theta}_{ac} \in]-\pi, \pi]$. Elle sera le deuxième paramètre. En effet, d'après la figure (2)

$$\begin{aligned} \tilde{\theta}_{ac} &= \theta_{ac} - \theta \\ \tilde{\theta}_{ac} &= \tan^{-1}(e_y/e_x) - \theta \end{aligned} \quad (5)$$

Sa dérivée $\dot{\tilde{\theta}}$ est exprimée alors comme suit

$$\dot{\tilde{\theta}}_{ac} = \frac{\dot{(e_y/e_x)}}{1+(e_y/e_x)^2} - \dot{\theta} \quad (6)$$

Ce qui donne tout calculs faits (utilisant (2) et (3))

$$\dot{\tilde{\theta}} = v \frac{\sin(\tilde{\theta}_{ac})}{d_{ac}} - \omega \quad (7a)$$

$$\dot{\tilde{\theta}} = \omega_r - \omega \quad (7b)$$

Cette dernière équation servira plus bas pour définir la loi de commande (cf. Section II-C).

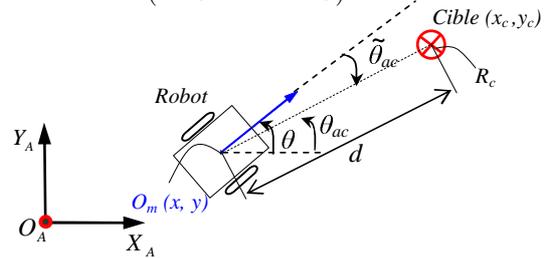


Fig. 2. Loi de commande proposée pour l'attraction vers la cible.

B. Contrôleur d'évitement d'obstacles

L'objectif de ce contrôleur est de rendre le robot capable d'éviter les obstacles qui gênent son attraction vers la cible. Pour se concentrer sur l'architecture de contrôle proposée, ce contrôleur n'est que brièvement décrit. Les détails théoriques sont disponible dans [13].

Ce contrôleur se base sur les méthodes des cycles limites [14], [15]. Les équations différentielles représentant la trajectoire désirée du robot sont données par le système suivant :

$$\begin{aligned} \dot{x}_r &= ay_r + x_r(R_c^2 - x_r^2 - y_r^2) \\ \dot{y}_r &= -ax_r + y_r(R_c^2 - x_r^2 - y_r^2) \end{aligned} \quad (8)$$

avec $a = \pm 1$ selon la direction optimale d'évitement (trigonométrique ou anti-trigonométrique). (x_r, y_r) représentent les coordonnées du robot par rapport au centre de l'obstacle. Ce dernier est entouré d'un cercle de rayon $R_{cl} = R_o + R_r + \epsilon$ où : R_o est le rayon de l'obstacle, R_r est le rayon du robot et ϵ est une marge de sécurité (cf. Figure 3).

Globalement, on résume l'algorithme d'évitement d'obstacles par les points suivants

- détecter l'obstacle gênant le plus proche,

- selon les informations provenant des capteurs, la direction d'évitement est choisie,
- le robot évite l'obstacle en suivant un cycle limite à rayon $R_c = R_{cl} - \xi$ (phase d'attraction),
- le robot entame la sortie de la zone d'influence de l'obstacle en suivant un cycle limite $R_c = R_{cl} + \xi$ (phase de répulsion) (cf. Figure 3). Avec ξ est une constante telle que $\xi \ll \epsilon$.

La consigne θ_{eo} permettant de calculer l'erreur d'orientation pour la loi de commande (cf. Figure 2) se calcule alors par la formule :

$$\theta_{eo} = \tan^{-1}\left(\frac{y_r}{x_r}\right) \quad (9)$$

Ainsi, l'erreur d'orientation (constituant le deuxième paramètre donné par le contrôleur d'évitement d'obstacles) est

$$\tilde{\theta}_{eo} = \theta_{eo} - \theta \quad (10)$$

Sa dérivée est

$$\dot{\tilde{\theta}} = \dot{\tan^{-1}}\left(\frac{y_r}{x_r}\right) - \omega \quad (11a)$$

$$\dot{\tilde{\theta}} = \omega_r - \omega \quad (11b)$$

Comme pour l'équation (7), (11) sera aussi utilisée dans la loi de commande (cf. Section II-C). Notons que l'on définit le 1^{er} paramètre $d_{eo} \rightarrow \infty$. Ainsi, la vitesse linéaire du robot à l'évitement d'obstacles (cf. Section II-C) dépendra uniquement de l'erreur d'orientation $\tilde{\theta}_{eo}$.

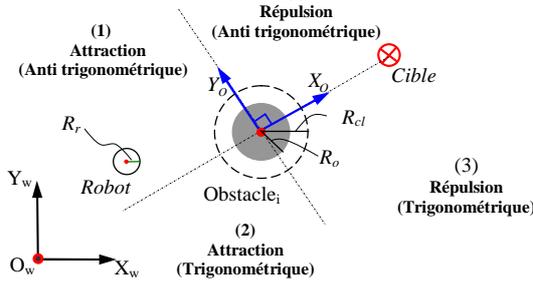


Fig. 3. Principe d'évitement d'obstacles utilisé.

C. Loi de commande

La loi de commande proposée est appliquée au robot quel que soit le contrôleur actif. Seul les paramètres $(d, \tilde{\theta})$ changent en fonction du contrôleur appliqué (attraction vers la cible ou évitement d'obstacles) (cf. Figure 1).

Elle s'exprime comme suit :

$$v = v_{max} e^{-\frac{1}{d}} \cos(\tilde{\theta}) \quad (12a)$$

$$\omega = \omega_r + k\tilde{\theta} \quad (12b)$$

- v_{max} est une constante représentant la vitesse linéaire maximale autorisée.
 - k est une constante telle que : $k > 0$.
 - $d = d_{ac}$ (attraction vers la cible) (cf. Figure 2) ou $d \rightarrow \infty$ (évitement d'obstacles) (on aura alors dans ce cas $e^{-\frac{1}{d}} \rightarrow 1$) (cf. Figure 1).
 - Il est à noter aussi qu'une condition initiale est imposée. Il s'agit d'avoir $\tilde{\theta} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ pour avoir $v \geq 0$.
- Considérons la fonction de Lyapunov suivante

$$V = \frac{1}{2} \tilde{\theta}^2$$

Le système est asymptotiquement stable si $\dot{V} < 0$.

$$\dot{V} = k\tilde{\theta}\dot{\tilde{\theta}}$$

En remplaçant (7) ou (11) (selon le contrôleur actif) dans (12), on obtient :

$$\dot{\tilde{\theta}} = -k\tilde{\theta}$$

et \dot{V} devient :

$$\dot{V} = -k\tilde{\theta}^2 < 0 \quad (13)$$

pour tout $\tilde{\theta} \neq 0$. d'où la stabilité asymptotique du contrôleur.

Par ailleurs, comme la fonction de Lyapunov est commune aux deux contrôleurs, elle ne prend en compte que l'erreur d'orientation $\tilde{\theta}$. Pour prouver que la distance séparant le robot à la cible décroît continuellement, il suffit de prouver que $\dot{d}_{ac} < 0$ (cf. Equation 15).

Pour le contrôleur d'attraction vers la cible, la dérivée de la distance d_{ac} s'écrit (cf. Equation 4)

$$\dot{d}_{ac} = \frac{e_x \dot{e}_x + e_y \dot{e}_y}{d_{ac}} \quad (14)$$

Utilisant les équations (2) et (3)

On obtient tout calculs faits

$$\dot{d}_{ac} = -v \cos(\tilde{\theta}) \quad (15)$$

En effet, en remplaçant v par (12a), on obtient :

$$\dot{d}_{ac} = -v_{max} e^{-\frac{1}{d_{ac}}} \cos^2(\tilde{\theta}) < 0$$

si $\tilde{\theta} \in]-\frac{\pi}{2}, \frac{\pi}{2}[$. Cette dernière condition est nécessaire uniquement pour l'état initial du robot. En effet, comme $\dot{V} < 0$ et V tend vers 0, alors l'erreur d'orientation décroît aussi pour tendre vers 0.

D. Sélection d'action

Le bloc de la sélection d'action permet de commuter d'un contrôleur à un autre. Il prend la décision grâce aux informations fournies par le bloc "Perception et Communication". Ainsi, selon le contrôleur actif, les paramètres correspondant $(d, \tilde{\theta})$ sont envoyés à la loi de commande afin de calculer les vitesses du robot. Si le robot détecte un obstacle et la distance robot-obstacle $d_{robot-obstacle}$ devient critique, c'est à dire $d_{robot-obstacle} \leq R_{cl}$, alors l'évitement d'obstacles est activé. Sinon, c'est l'attraction vers la cible qui est actif.

Les deux blocs restant : *phase de transition* et *Adaptation du gain* seront détaillés dans la section suivante.

III. STABILITÉ GLOBALE DE L'ARCHITECTURE DE CONTRÔLE PROPOSÉE

Bien qu'il soit prouvé que les contrôleurs cités ci-dessus sont individuellement stables, la commutation entre ces contrôleurs peut engendrer l'instabilité du système global [10]. En effet, il peut y'avoir une discontinuité de commande en passant d'un contrôleur à un autre. Il est alors judicieux de contraindre ces commutations afin d'assurer un comportement satisfaisant du robot.

Parmi les théorèmes des systèmes hybrides permettant la commutation d'un système stable à un autre, on retrouve le théorème de la fonction de Lyapunov commune [12]. Il

s'agit de la généralisation du théorème de la fonction de Lyapunov aux systèmes hybrides. Ce théorème s'énonce comme suit :

Théorème- Un système hybride est dit stable au sens de Lyapunov s'il existe une fonction de Lyapunov V dans un voisinage de l'état d'équilibre satisfaisant les propriétés classiques suivantes :

- $V(x)$ doit être définie positive.
- si $\dot{V}(x)$ est définie négative sur le système global, alors la stabilité est asymptotique.
- si $\dot{V}(x)$ est semi-définie négative sur le système global, alors la stabilité est marginale.

On a prouvé dans la section (II-C) que la loi de commande utilisée est asymptotiquement stable (la fonction de Lyapunov correspondante est strictement décroissante). Comme la même loi sera utilisée pour les deux contrôleurs, la fonction de Lyapunov sera décroissante au moment de l'exécution de chaque contrôleur. Cependant, le problème se pose (comme pour tous les systèmes hybrides) au moment des commutations où il y'aura une discontinuité de l'erreur $\tilde{\theta}$, avec le risque certain d'augmentation de la fonction de Lyapunov.

Pour pallier à cet inconvénient, on propose d'utiliser la notion de stabilité faible des systèmes hybrides proposée dans [16] et reprise dans [17] pour l'étude de systèmes mécaniques non réguliers. Ce théorème permet à la fonction de Lyapunov de croître au moment des transitions à condition que sa valeur reste bornée. On notera t_{s_i} l'instant de la i^{eme} commutation.

Plus précisément, l'énoncé de la proposition de la stabilité faible est comme suit :

Théorème- Supposons qu'un système évolue sur plusieurs phases (intervalles de temps associés à plusieurs systèmes continus). Si :

- $\dot{V}(x) \leq 0$ sur le système global en dehors des instants de commutations.
- Pour tout intervalle (phase) fini $[t_{s_i}, t_{s_{i+1}}]$:
 $V(t_{s_i}) \leq V(t_{s_{i+1}})$.

Alors le système global est faiblement stable.

Ce théorème relâche sous conditions la contrainte de décroissance de la fonction de Lyapunov aux moments des commutations dans la limite où cette fonction reste bornée. Cependant, elle impose toujours qu'elle soit décroissante en dehors de ces moments. Il est très facile de borner la fonction de Lyapunov proposée. En effet, comme elle ne dépend que de l'erreur d'orientation dont la valeur maximale est $\theta = \pi$, alors :

$$V_{max} = \pi^2/2$$

Dans ce qui suit, on se propose d'ajuster le gain k de la loi de commande (cf. Equation 12b) afin de passer les commutations tout en assurant la stabilité globale du contrôle. L'idée est que la fonction de Lyapunov converge le plus rapidement possible afin que sa valeur vérifie

$$V(t) \leq V(t_{avs})$$

$V(t_{avs})$ étant sa valeur à l'instant précédant celui de la commutation. L'objectif est alors de minimiser le temps où le robot navigue sous un contrôle faiblement stable, c'est à dire où $V(t) \geq V(t_{avs})$. D'après ce qui précède, on peut

aborder les deux blocs restant de l'architecture de contrôle (cf. Figure 1).

A. Adaptation du gain de la loi de commande

Le déclenchement de l'ajustement du gain de la loi de commande survient suite à l'un des événements suivants :

- le contrôle du robot commute d'un contrôleur à un autre,
- le contrôleur d'évitement d'obstacles commute d'un obstacle à un autre obstacle,
- le contrôleur d'évitement d'obstacles passe de la phase d'attraction à la phase de répulsion (cf. Figure 3) [13].

Pour assurer que V décroît en un temps fini T_{max} que l'on peut imposer selon certaines contraintes (éviter la collision avec les obstacles, contraintes structurelles du robot, etc.), il faut avoir

$$V(t_s + T_{max}) \leq V(t_{avs}) \quad (16)$$

Pour remonter à l'évolution de l'erreur, il suffit de résoudre l'équation (12b). En effet, utilisant (7b) ou (11b) (selon le contrôleur actif), elle devient

$$\dot{\tilde{\theta}} = -k_1 \tilde{\theta} \quad (17)$$

La résolution de cette équation différentielle donne

$$\tilde{\theta}(t) = \tilde{\theta}(t_s) e^{-k_1(t-t_s)} \quad (18)$$

L'évolution de la fonction de Lyapunov en est facilement déduite :

$$\begin{aligned} V(t) &= (\tilde{\theta}^2(t_s)/2) e^{-2k_1(t-t_s)} & (b) \\ V(t) &= V(t_s) e^{-2k_1(t-t_s)} & (c) \end{aligned} \quad (19)$$

Le gain de la loi de commande k est alors

$$k = \frac{\ln(V(t)/V(t_s))}{-2(t-t_s)} \quad (20)$$

On note que k ainsi calculé reste toujours positif. En effet, $V(t) \leq V(t_s)$ (cf. Equation 19) et donc $\ln(V(t)/V(t_s)) \leq 0$.

Ainsi, la valeur du gain k_s permettant à la fonction de Lyapunov d'atteindre la valeur $V(t_{avs})$ en un temps T_{max} est

$$k_s = \frac{\ln(V(t_{avs})/V(t_s))}{-2T_{max}} \quad (21)$$

B. Phase transition

S'il y'a un événement discret (commutation d'un contrôleur à un autre, passage de la phase d'attraction à la phase de répulsion, etc), le bloc *Phase de transition* active le bloc *Adaptation du gain* qui recalcule le nouveau gain k à utiliser dans la loi de commande.

C. Retour à la valeur initiale du gain k

On a proposé un bloc permettant d'ajuster la loi de commande afin de pouvoir passer les commutations en garantissant la stabilité (cf. Subsection III-A). Cependant, il serait préférable que le gain k retrouve sa valeur initiale (correspondant à sa valeur optimale avant la commutation) qu'on notera dans la suite k_{opt} , une fois la condition (16) atteinte. Ainsi, l'idée est de lui donner une dynamique lui permettant de retrouver sa valeur initiale. Le choix que l'on garde est

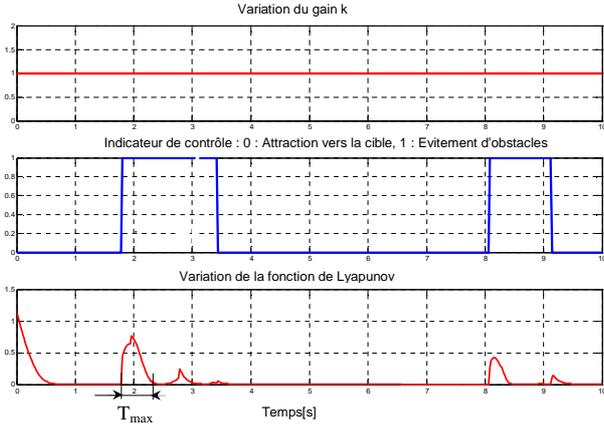


Fig. 4. Variation de la fonction de Lyapunov avec un gain k constant.

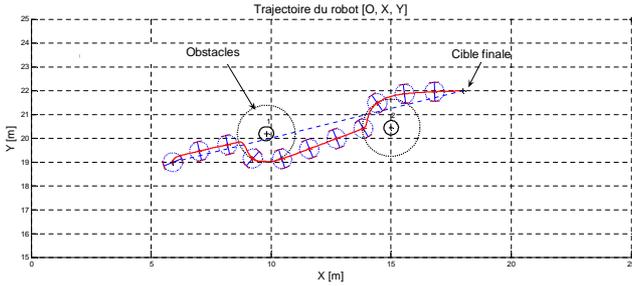


Fig. 5. Trajectoire du robot mobile simulé.

$$k(t) = (k_s - k_{opt})e^{-\lambda(t-t_s)} + k_{opt} \quad (22)$$

Où λ est une constante déterminant la vitesse de convergence de $k(t)$ vers sa valeur optimale. D'après l'équation (13), le système reste toujours asymptotiquement stable car $k(t) > 0$.

IV. RÉSULTATS DE SIMULATION

Afin de montrer la pertinence de l'architecture de contrôle proposée, on propose d'abord de simuler le mouvement d'un robot mobile en gardant la valeur k constante avec ($k = k_{opt} = 1$) tout au long de l'évolution du robot (il n'y a pas d'ajustement de gain aux moments des commutations). Les moments des transitions entre contrôleurs ainsi que l'évolution de la fonction de Lyapunov sont représentés sur la figure (4). Même si le robot semble accomplir sa tâche convenablement, on voit l'évolution de la fonction de Lyapunov au moment des transitions (cf. Figure 4). Le système est faiblement stable pendant un temps T_{max} représenté sur la figure. On remarque qu'il est relativement grand et imposé par la valeur constante du gain k .

Le mouvement d'un robot mobile en présence d'obstacles est maintenant simulé avec l'architecture de contrôle proposée (cf. Figure 5). On note que la valeur du gain k optimal choisie est la même que précédemment, c'est à dire $k_{opt} = 1$. Les moments des transitions entre les contrôleurs sont représentés sur la figure (6).

La même figure montre l'évolution de la fonction de Lyapunov en fonction du temps. On voit bien qu'avec l'architecture de contrôle proposée et précisément grâce au bloc d'ajustement du gain k , la fonction de Lyapunov converge beaucoup plus rapidement. En effet, T_{max} est nettement plus petit qu'avec un gain constant. Notons enfin que durant l'évitement d'obstacles (indicateur de contrôle = 1)

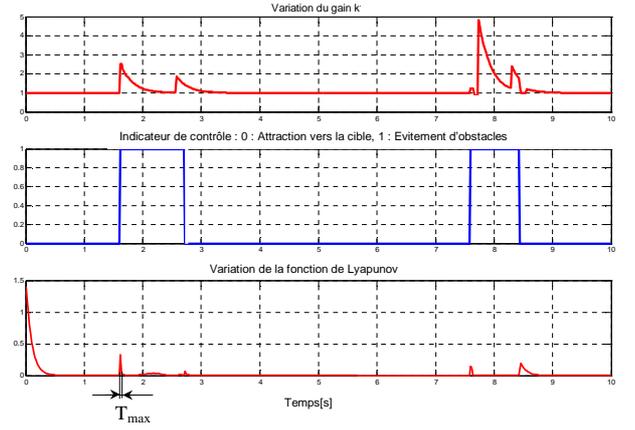


Fig. 6. Variation de la fonction de Lyapunov et du gain k .

(cf. Figure 6), le gain k peut-être réajusté deux fois. Il s'agit de l'événement discret correspondant au passage de la phase d'attraction à la phase de répulsion durant l'évitement.

V. RÉSULTATS EXPÉRIMENTAUX

L'architecture de contrôle proposée est implémentée sur un robot mobile Khepera III. Il s'agit d'un robot de type unicycle. Une caméra placée en haut de la plateforme donne les positions des obstacles, de la cible ainsi que la position réelle du robot. Les positions de l'obstacle et de la cible sont exprimées dans le repère du robot.

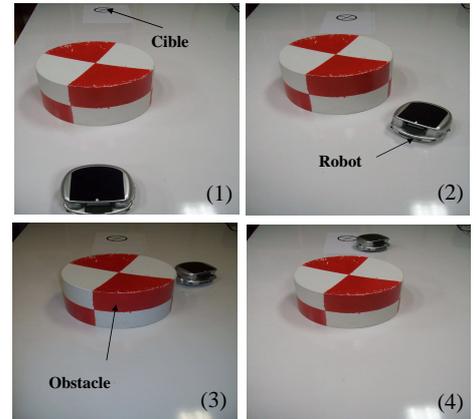


Fig. 7. La tâche d'attraction vers la cible avec évitement d'obstacle accomplie par le robot Khepera III.

Le robot doit alors atteindre une cible située à 1.4 mètres de sa position. Un obstacle gênant se trouve entre la position initiale du robot et la cible finale. L'expérience est réalisée deux fois en respectant la même condition initiale : une fois avec un gain k constant au niveau de la loi de commande (cf. Equation 12b), et une fois avec l'architecture de contrôle proposée. Dans les deux cas, le robot atteint bien la cible finale en évitant l'obstacle gênant (cf. Figure 7). L'évitement se fait dans le sens trigonométrique. En effet, comme le robot, le centre de l'obstacle et la cible se retrouvent sur la même droite, ce choix est pris par défaut par le robot. Toutefois, si cela n'était pas le cas, le contrôleur d'évitement d'obstacles lui permet de choisir le sens d'évitement optimal (cf. Section II-B). La trajectoire réelle parcourue par le robot avec l'architecture de contrôle proposée est représentée dans la figure (8).

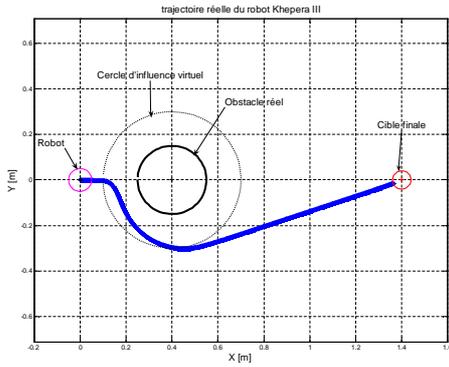


Fig. 8. Evolution du robot vers sa cible en présence d'un obstacle gênant.

Afin de valider les conclusions théorique et de simulation, les variations de la fonction de Lyapunov et du gain k de la loi de commande en fonction du temps sont représentées dans les figures (9) et (10). De même, on remarque que l'architecture de contrôle proposée permet à la fonction de Lyapunov de converger plus rapidement qu'avec l'utilisation d'un simple gain constant. Enfin, le gain k rejoint exponentiellement sa valeur nominale $k_{opt} = 1$, une fois la convergence atteinte.

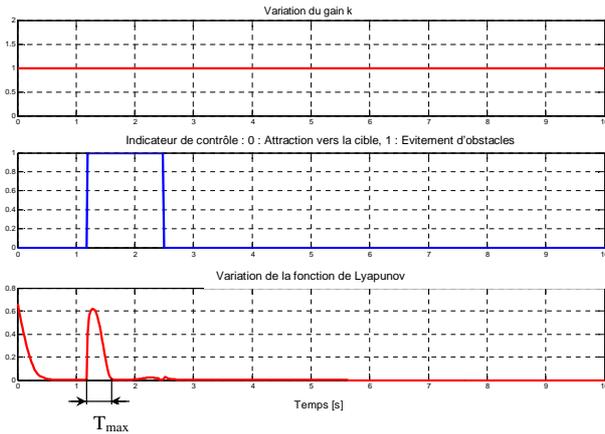


Fig. 9. Variation de la fonction de Lyapunov avec un gain k constant.

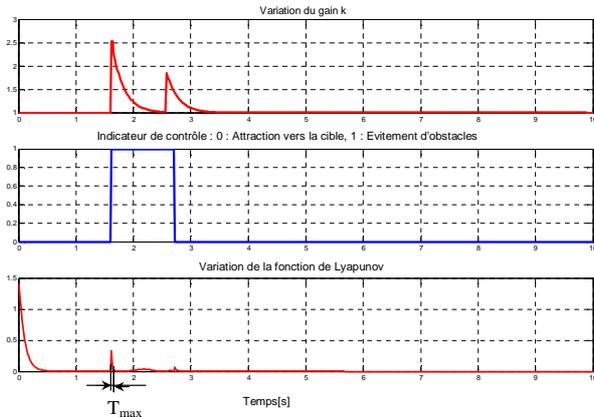


Fig. 10. Variation de la fonction de Lyapunov et du gain k tout au long de la navigation du robot.

VI. CONCLUSION ET PERSPECTIVES

Une architecture de contrôle se basant sur les systèmes hybrides a été proposée. Grâce à ces systèmes, il est possible de diviser l'architecture de contrôle en un ensemble de

contrôleurs élémentaires afin d'accéder et d'étudier chaque contrôleur séparément. Même si chaque contrôleur est individuellement stable, la stabilité globale n'est pas forcément garantie. Dans ce papier, la difficulté de trouver une seule fonction de Lyapunov pour tous les systèmes élémentaires formant le système hybride global a été résolue. La stabilité globale a alors été prouvée. L'idée proposée ici est d'ajouter à l'architecture de contrôle un bloc d'ajustement du gain de la loi de commande. Ceci pallie à l'inconvénient des sauts de la fonction de Lyapunov aux moments des commutations. Le système est alors faiblement stable pendant un temps nettement plus réduit qu'avec l'utilisation d'un gain constant classique. Les résultats de simulation et expérimentaux ont confirmé l'étude théorique.

L'application de cette architecture à un système multi robot fera l'objet de nos futurs travaux. Il s'agit de rajouter un contrôleur de maintien de formation et de prouver la stabilité globale du système grâce à une seule fonction de Lyapunov. L'introduction des contraintes structurales du robot pour contraindre le temps de convergence est également envisagée.

RÉFÉRENCES

- [1] Latombe J-C. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.
- [2] Rimon E. et Koditschek D. E. Exact robot navigation using artificial potential fields. *IEEE Transactions on Robotics and Automation*, 8(5) :501–518, 1992.
- [3] Belta C. et Isler V. et Pappas G.J. Discrete abstractions for robot motion planning and control in polygonal environments. *IEEE Transactions on Robotics*, 21(5) :864–874, 2005.
- [4] Conner D.C. et Choset H. et Rizzi A. Integrated planning and control for convex-bodied nonholonomic systems using local feedback control policies. In *Proceedings of Robotics : Science and Systems*, pages 57–64, Philadelphia, USA, 2006.
- [5] S. Egerstedt M. et Johansson K. H. et Lygeros J. et Sastry. Behavior based robotics using regularized hybrid automata. *Computer Science ISSN 0302-9743*, 1790 :103–116, 2000.
- [6] Toibero J.M. et Carelli R. et Kuchen B. Switching control of mobile robot for autonomous navigation in unknown environments. *IEEE International Conference on Robotics and Automation*, pages 1974–1979, 2007.
- [7] Adouane L. Hybrid and safe control architecture for mobile robot navigation. In *9th Conference on Autonomous Robot Systems and Competitions*, Portugal, May 2009.
- [8] Brooks R. A. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2 :14–23, 1986.
- [9] Arkin R. C. Motor schema-based mobile robot navigation. *International journal of robotics research*, 8(4) :92–112, 1989.
- [10] M. S. Branicky. Stability of switched and hybrid systems. In *33rd IEEE Conference on Decision Control*, pages 3498–3503, 1993.
- [11] Benzerrouk A. et Adouane L. et Martinet P. et Andreff N. Multi lyapunov function theorem applied to a mobile robot tracking a trajectory in presence of obstacles. In *European Conference on Mobile Robots*, 2009.
- [12] Liberzon D. *Switching in systems and control*. Birkhäuser, 2003.
- [13] Adouane L. Orbital obstacle avoidance algorithm for reliable and on-line mobile robot navigation. In *9th Conference on Autonomous Robot Systems and Competitions*, May 2009.
- [14] J. Kim, D. et Kim. A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer. *Robotics and Autonomous Systems*, 42 :17–30, 2003.
- [15] K. Lee Jie M.S. et Baek J.H. et Hong Y.S. et Woong. Real time obstacle avoidance for mobile robot using limit-cycle and vector field method. *Knowledge-Based Intelligent Information and Engineering Systems*, pages 866–873, October 2006.
- [16] Niculescu S. et Orhant Brogliato B. On the control of finite dimensional mechanical systems with unilateral constraints. *IEEE Transactions on Automatic Control*, 42(2) :200–215, 1997.
- [17] Bourgeot J.M. *Contribution à la commande de systèmes mécaniques non-réguliers*. PhD thesis, Institut National Polytechnique de Grenoble- France, 2004.