

Analyse de lois de commande d'asservissement visuel

Analysis of Classical and New Image-Based Control Laws

Mohammed Marey*

François Chaumette

INRIA / IRISA Rennes
Campus de Beaulieu,
35 042 Rennes-cedex, France.

Prenom.Nom@irisa.fr

Résumé

Dans cet article, nous analysons et comparons cinq lois de commande d'asservissement visuel. Trois d'entre elles sont classiques tandis que les deux restantes sont originales. La première loi de commande que nous proposons est obtenue par une simple généralisation des schémas de contrôle classiques. Elle est basée sur un paramètre permettant de varier le comportement du système. La deuxième a été construite pour tenter d'obtenir la stabilité asymptotique globale du système. Cependant nous démontrons que, comme toutes les lois existantes, elle est potentiellement sujette à des minima locaux, rendus ici attractifs. Nous présentons ensuite une étude analytique de ces lois de commande dans le cas de mouvements de translation et de rotation selon l'axe optique. Cette analyse nous a permis de mettre en évidence de nouveaux cas de singularité. Enfin, plusieurs résultats montrent que la loi de commande avec un paramètre de comportement connaît une plus large gamme de succès que les quatre autres lois.

Mots Clef

Asservissement visuel, robotique.

Abstract

In this paper, we analyse and compare five image-based visual servoing control laws. Three of them are classical while two new ones are proposed. The first one is obtained by a simple generalisation of the classical control schemes. It is based on a parameter which allows tuning the system behavior. The second one is designed to try to obtain the global asymptotic stability of the system but we demonstrate that it may fail due to attractive local minima. An analytical study when translational motion along and rotational motion around the optical axis is also presented. Finally, numerous simulation results show that the new control law with a behavior controller has a wider range of success than the other control schemes.

*Mohammed Marey bénéficie d'un financement du gouvernement égyptien

Keywords

Visual servoing, robotics.

1 Introduction

L'asservissement visuel est une approche connue pour augmenter la précision, la polyvalence et la robustesse des systèmes de vision robotique [10, 4]. Deux aspects principaux ont un fort impact sur le comportement de tout schéma d'asservissement visuel : le choix des informations visuelles utilisées comme entrées de la loi de commande et le choix de cette loi de commande. Les informations visuelles peuvent être choisies dans l'espace de l'image (coordonnées de points, paramètres de droites ou d'ellipses, moments,... [7, 14, 5, 9, 3]), dans l'espace cartésien (pose, coordonnées de points 3D,...) [15, 13], ou être composées de combinaisons des deux sortes de primitives en essayant d'intégrer les avantages des méthodes basées sur l'image et basées sur la position [11, 6, 1, 8]. On obtient alors des schémas d'asservissement visuel dits 2D, 3D ou 2D 1/2 respectivement.

Quant au choix de la loi de commande [7, 12, 4], il influe sur le comportement des informations visuelles choisies (décroissance exponentielle locale ou globale, minimisation au second ordre) et peut, ou non, conduire à des minima locaux ou à des singularités [2]. Dans cet article, nous ne nous intéressons pas au choix des informations visuelles, mais à l'analyse des lois de commande. Pour cela, nous considérons les informations visuelles les plus simples et les plus usuelles, à savoir les coordonnées de points dans l'image. Nous nous plaçons donc dans le cadre du schéma d'asservissement visuel 2D le plus basique. Pour tenter d'être le plus exhaustif possible, nous présentons deux nouvelles lois de commande. La première est une généralisation des lois de commande classiques. Elle utilise un paramètre de comportement qui peut tout d'abord être utilisé comme un commutateur entre les trois schémas classiques selon que ce paramètre est mis à 0, 1/2 ou 1. Par ailleurs, le choix d'autres valeurs permet de modifier le comportement du système. Nous proposons également

une deuxième loi de commande. Celle-ci a été construite pour tenter d'obtenir la stabilité asymptotique globale du système. Malheureusement, si celle-ci peut être démontrée dans l'espace de la fonction de tâche, elle ne peut l'être dans l'espace des configurations SE_3 , du moins avec les informations visuelles choisies, en raison de minima locaux. Cet article inclut enfin une étude analytique complète des lois de commande pour des mouvements combinant translation le long de l'axe optique et rotation autour de cet axe. Cette analyse nous a permis de mettre en évidence, pour la première fois, une singularité de la commande proposée dans [12].

L'article est structuré de la manière suivante : dans la deuxième section, les trois lois de commande classiques d'asservissement visuel sont tout d'abord rappelées. On en déduit ensuite la loi de commande basée sur un paramètre de comportement. Dans la troisième section, une autre nouvelle loi de commande est proposée et sa stabilité est étudiée. Dans la quatrième section, une analyse des cinq lois de commande lors de rotations et translations le long de l'axe optique est proposée. Ses conclusions sont vérifiées dans la cinquième section par des résultats obtenus soit en simulation soit à l'aide d'un système expérimental de vision robotique.

2 Loi de commande avec un paramètre de comportement

Soit $\mathbf{s} \in \mathbb{R}^k$ le vecteur contenant les k informations visuelles choisies, $\mathbf{s}^* \in \mathbb{R}^k$ la valeurs désirée de \mathbf{s} , et $\mathbf{v} \in \mathbb{R}^6$ la vitesse instantanée de la caméra. La plupart des lois de commande classiques ont la forme suivante :

$$\mathbf{v} = -\lambda \widehat{\mathbf{L}}_{\mathbf{s}}^+ (\mathbf{s} - \mathbf{s}^*) \quad (1)$$

où $\lambda \in \mathbb{R}^+$ est un gain fixant l'amplitude la vitesse envoyée au contrôleur bas niveau du robot et où $\widehat{\mathbf{L}}_{\mathbf{s}}^+$ est la pseudo inverse d'une estimation ou d'une approximation de la matrice d'interaction associée à \mathbf{s} (définie telle que $\dot{\mathbf{s}} = \mathbf{L}_{\mathbf{s}} \mathbf{v}$). Plusieurs formes pour $\widehat{\mathbf{L}}_{\mathbf{s}}$ ont été proposées par le passé [7, 4]. Pour simplifier, considérons que toutes les valeurs peuvent être précisément calculées, nous amenant aux possibilités suivantes :

$$1) \quad \widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}^*} \quad (2)$$

$$2) \quad \widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\mathbf{s}(t)} \quad (3)$$

$$3) \quad \widehat{\mathbf{L}}_{\mathbf{s}} = (\mathbf{L}_{\mathbf{s}^*} + \mathbf{L}_{\mathbf{s}(t)})/2. \quad (4)$$

En choisissant (2), $\widehat{\mathbf{L}}_{\mathbf{s}}$ correspond à la valeur de la matrice d'interaction à la configuration désirée et est donc constante durant tout l'asservissement. Avec (3), $\widehat{\mathbf{L}}_{\mathbf{s}}$ est modifiée à chaque itération de la loi de commande en utilisant la valeur courante de la matrice d'interaction (ce qui nécessite en général une localisation 3D de l'objet par rapport à la caméra pour estimer les paramètres de profondeur 3D intervenant dans les composantes en translation

de la matrice d'interaction). Enfin, avec (4), la moyenne de ces deux valeurs est utilisée [12]. En utilisant l'une de ces trois possibilités dans la loi de commande (1), on obtient les lois de commande que nous appellerons par la suite D, C et M (pour désiré, courant et moyenne respectivement). A proximité de la configuration désirée, où l'erreur $(\mathbf{s} - \mathbf{s}^*)$ est faible, le même comportement est obtenu quel que soit le choix fait pour $\widehat{\mathbf{L}}_{\mathbf{s}}$ puisqu'on a alors $\mathbf{L}_{\mathbf{s}(t)} \approx \mathbf{L}_{\mathbf{s}^*}$. Par contre, quand l'erreur $(\mathbf{s} - \mathbf{s}^*)$ est grande, il est bien connu que le choix de $\widehat{\mathbf{L}}_{\mathbf{s}}$ influe fortement sur le comportement du système puisqu'on a alors $\mathbf{L}_{\mathbf{s}(t)} \neq \mathbf{L}_{\mathbf{s}^*}$ (sauf si l'on pouvait trouver des informations visuelles telles que la matrice d'interaction est constante dans tout l'espace de travail, ce qui est un problème toujours ouvert).

Une forme bien plus générale de $\widehat{\mathbf{L}}_{\mathbf{s}}$ peut être obtenue facilement en introduisant un paramètre de comportement $\beta \in \mathbb{R}$:

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \mathbf{L}_{\beta} = \beta \mathbf{L}_{\mathbf{s}^*} + (1 - \beta) \mathbf{L}_{\mathbf{s}(t)}. \quad (5)$$

Nous appellerons par la suite loi de commande G celle obtenue en utilisant cette forme (5) dans (1). Bien sûr, si $\beta = 1$, nous retrouvons la loi de commande D basée sur la matrice d'interaction désirée. Si $\beta = 0$, nous obtenons la loi de commande C qui utilise la matrice d'interaction courante, et si $\beta = 1/2$, nous obtenons la loi de commande M. La propriété la plus intéressante de la loi de commande généralisée G est que, en faisant varier β de 0 à 1/2, le comportement du système change graduellement du comportement obtenu en utilisant la loi de commande C au comportement obtenu en utilisant la loi M. De même, le comportement varie graduellement du comportement obtenu en utilisant la loi M au comportement obtenu en utilisant la loi D quand β varie de 1/2 à 1. Ce nouveau schéma de commande nous permet donc d'adapter le comportement du système à partir de la valeur choisie pour β . Dans la dernière partie de cet article, nous verrons que certaines valeurs de β permettent effectivement au système d'avoir un meilleur comportement qu'en utilisant les lois de commande D, C ou M. On peut enfin signaler qu'en cas d'erreur dans les paramètres intervenant dans la matrice d'interaction (due à des erreurs de calibration, des erreurs de mesure ou des erreurs dans l'estimation de paramètres 3D intervenant dans les termes de translation), les matrices $\mathbf{L}_{\mathbf{s}^*}$ et $\mathbf{L}_{\mathbf{s}(t)}$ écrites ci-dessus doivent être remplacées par des approximations $\widehat{\mathbf{L}}_{\mathbf{s}^*}$ et $\widehat{\mathbf{L}}_{\mathbf{s}(t)}$, mais cela ne modifie pas fondamentalement leur comportement et propriétés générales respectifs.

3 Commande "Pseudo-GAS"

En utilisant des informations visuelles redondantes telles les coordonnées de points dans l'image, seule la stabilité asymptotique locale peut être démontrée pour les lois de commande D, C et M [4]. Ceci est également vrai pour la loi de commande G. Nous présentons à présent une loi de commande construite pour tenter d'obtenir la stabilité asymptotique globale.

3.1 Modélisation

Choisissons comme fonction de tâche $\mathbf{e} \in \mathbb{R}^6$ l'erreur suivante

$$\mathbf{e} = \mathbf{L}_{\mathbf{s}^*}^+(\mathbf{s} - \mathbf{s}^*) \quad (6)$$

où \mathbf{s}^* est choisie pour avoir $\mathbf{L}_{\mathbf{s}^*}$ de rang plein.

Comme $\mathbf{L}_{\mathbf{s}^*}$ est constante, la dérivée temporelle de \mathbf{e} est donnée par $\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{v}$ où $\mathbf{L}_e \in \mathbb{R}^{6 \times 6}$ s'écrit

$$\mathbf{L}_e = \mathbf{L}_{\mathbf{s}^*}^+ \mathbf{L}_s. \quad (7)$$

Nous pouvons remarquer que \mathbf{L}_e est de rang plein 6 lorsque \mathbf{L}_s est aussi de rang plein 6. Pour obtenir une décroissance exponentielle de \mathbf{e} (c'est à dire, $\dot{\mathbf{e}} = -\lambda \mathbf{e}$), nous obtenons immédiatement la commande

$$\mathbf{v} = -\lambda \mathbf{L}_e^{-1} \mathbf{e} \quad (8)$$

En introduisant (6) et (7) dans (8), on en déduit

$$\mathbf{v} = -\lambda (\mathbf{L}_{\mathbf{s}^*}^+ \mathbf{L}_s)^{-1} \mathbf{L}_{\mathbf{s}^*}^+ (\mathbf{s} - \mathbf{s}^*). \quad (9)$$

La dérivation de cette loi de commande suit un raisonnement extrêmement classique et simple. Il est donc étonnant qu'elle n'ait pas été proposée ni testée auparavant. Cela provient probablement du fait qu'elle utilise aussi bien \mathbf{L}_s que $\mathbf{L}_{\mathbf{s}^*}$, ce qui n'était pas d'usage avant la loi M proposée récemment dans [12]. Pour obtenir la loi de commande D à partir des dérivations ci-dessus, il suffit d'approximer \mathbf{L}_s par $\mathbf{L}_{\mathbf{s}^*}$ pour aboutir à $\mathbf{L}_e = \mathbf{L}_{\mathbf{s}^*}^+ \mathbf{L}_s = \mathbf{I}_6$. Pour obtenir la loi de commande C, il faut poser $\mathbf{e} = \mathbf{L}_{\mathbf{s}^*}^+(\mathbf{s} - \mathbf{s}^*)$ et négliger les variations de \mathbf{L}_s dans le calcul de $\dot{\mathbf{e}}$ pour aboutir à $\mathbf{L}_e = \mathbf{L}_{\mathbf{s}^*}^+ \mathbf{L}_s = \mathbf{I}_6$.

On peut noter qu'à proximité de la configuration désirée, où $\mathbf{L}_s \approx \mathbf{L}_{\mathbf{s}^*}$, cette loi de commande est équivalente à D puisque $\mathbf{L}_{\mathbf{s}^*}^+ \mathbf{L}_s \approx \mathbf{I}_6$, et donc également à toutes les autres. Par contre, lorsque \mathbf{s} est loin de \mathbf{s}^* , cette nouvelle loi de commande entraîne un comportement différent puisque (9) ne peut s'écrire sous la forme générale établie précédemment.

Pour étudier la stabilité du schéma de commande (9), on considère comme fonction de Lyapunov candidate

$$\mathcal{L} = \frac{1}{2} \|\mathbf{e}(t)\|^2. \quad (10)$$

On a

$$\dot{\mathcal{L}} = \mathbf{e}^\top \dot{\mathbf{e}} = \mathbf{e}^\top \mathbf{L}_e \mathbf{v} \quad (11)$$

En utilisant (8), on obtient

$$\dot{\mathcal{L}} = -\lambda \mathbf{e}^\top \mathbf{L}_e \mathbf{L}_e^{-1} \mathbf{e} = -\lambda \mathbf{e}^\top \mathbf{e} < 0, \quad \forall \mathbf{e} \neq 0. \quad (12)$$

La loi de commande (9) semble donc être très prometteuse puisqu'elle est globalement asymptotiquement stable (GAS) dans l'espace de la fonction de tâche \mathbf{e} (car \mathcal{L} tend vers 0 quelle que soit la valeur initiale de \mathbf{e}). De plus, (9) assure le comportement spécifié $\dot{\mathbf{e}} = -\lambda \mathbf{e}$, à savoir une convergence exponentielle découplée de chaque composante de \mathbf{e} , à partir du moment où $\mathbf{L}_{\mathbf{s}^*}$ et \mathbf{L}_s ne sont pas entachées d'erreur.

Cela dit, pour démontrer la stabilité dans l'espace des configurations, à savoir SE_3 ou encore l'espace des informations visuelles \mathbf{s} , nous devons démontrer que $\mathbf{e} = 0$ si et seulement si $\mathbf{s} = \mathbf{s}^*$. Cette dernière condition devient fautive dès que

$$(\mathbf{s} - \mathbf{s}^*) \in \text{Ker } \mathbf{L}_{\mathbf{s}^*}^+ \quad (13)$$

Dans ce cas, on a alors $\mathbf{e} = 0$, ce qui implique que $\mathbf{v} = 0$, alors que $\mathbf{s} \neq \mathbf{s}^*$, ce qui correspond à un minimum local. En d'autres termes, la fonction de tâche \mathbf{e} choisie assure un difféomorphisme avec SE_3 , mais ce difféomorphisme n'est pas global (i.e. valide dans tout SE_3) mais seulement local dès qu'il existe une configuration telle que (13) est vérifiée.

Comme nous allons le voir à présent, nous avons pu mettre en évidence des configurations qui mènent à des minima locaux. Nous appelons donc par la suite la loi de commande 9 PG (pour "pseudo-GAS"). On peut en conclure que, comme pour toutes les lois de commande existant jusqu'à présent, seule la stabilité locale de (9) peut être assurée, du moins en utilisant comme informations visuelles des coordonnées de points dans l'image. Ce résultat, même s'il n'est pas positif, nous semble tout de même intéressant puisqu'il sera peut-être possible à l'avenir de trouver d'autres informations visuelles redondantes telles que $\text{Ker } \mathbf{L}_{\mathbf{s}^*}^+ = \mathbf{0}$, ce qui permettra alors d'obtenir la stabilité asymptotique globale en utilisant la loi de commande (9).

3.2 Résultats de simulation

Dans cette section, nous considérons deux configurations difficiles et comparons les résultats obtenus par les différents schémas de commande décrits précédemment. On rappelle que les informations visuelles \mathbf{s} ont été choisies comme les coordonnées x et y de points de l'image. Comme souvent, l'objet d'intérêt est un carré et nous utilisons les coordonnées de ses quatre sommets. Pour l'ensemble des résultats présentés, le gain λ a été fixé constant et égal à 0.1.

Premier cas. La pose désirée entre la caméra et le carré est $(0, 0, 0.5, 0, 0, 0)$, ce qui signifie que la caméra doit être à 0.5 m de ce carré, son plan image parallèle au plan du carré, et tel que le carré soit centré dans l'image (les trois premières composantes de la pose représente la translation entre la caméra et l'objet (en m), les trois dernières les rotations entre ces deux repères (en dg)). Dans le cas considéré, les configurations où $(\mathbf{s} - \mathbf{s}^*) \in \text{Ker } \mathbf{L}_{\mathbf{s}^*}^+$ correspondent à des cas très particuliers où les quatre points sont alignés dans l'image, c'est-à-dire lorsque le carré est observé sur sa tranche [2]. La pose initiale de la caméra a été fixée à $(0, 0, 0.4, 80, 20, 10)$ et a donc une orientation très différente de la pose désirée. Les résultats de simulation obtenus pour les lois de commande D, C, M et PG sont illustrés sur la Figure 1. Les lois de commande classiques D, C et M permettent à la caméra de converger vers la pose désirée alors que la nouvelle loi de commande PG converge sur une configuration où les quatre points sont alignés dans

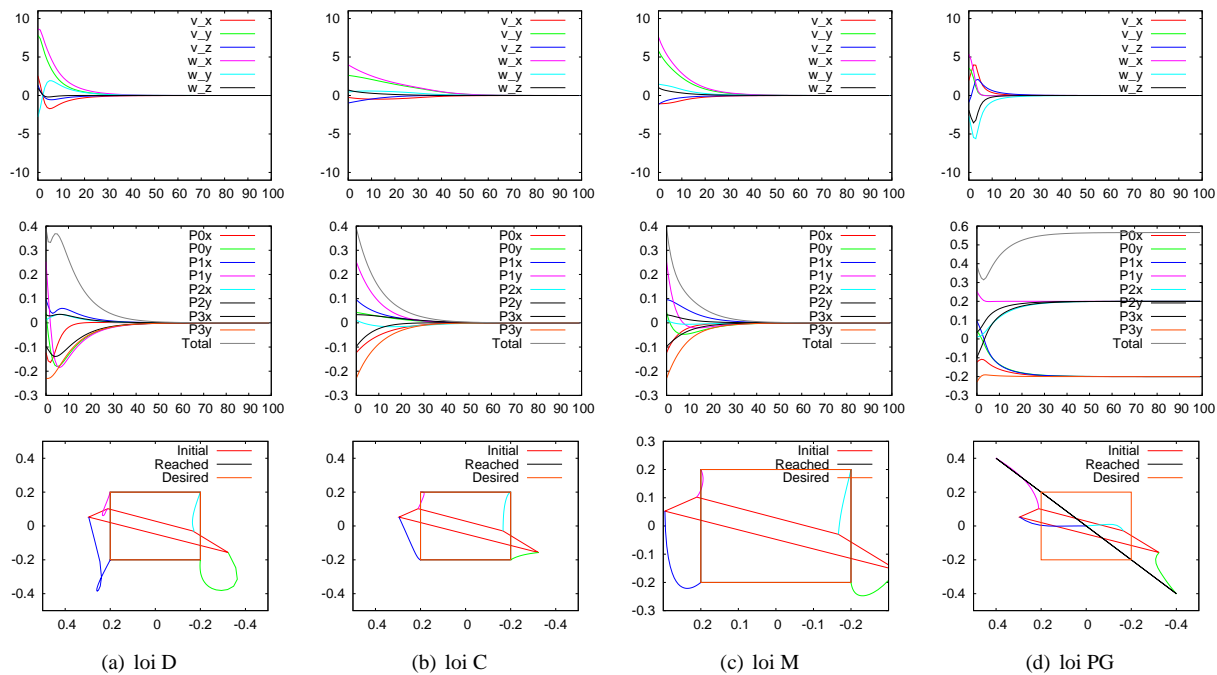


FIG. 1 – Résultats obtenus pour le cas 1. Première ligne : vitesse de la caméra (en cm/s et dg/s), deuxième ligne : erreurs des points de l’image (en m), troisième ligne : trajectoires des points dans l’image .

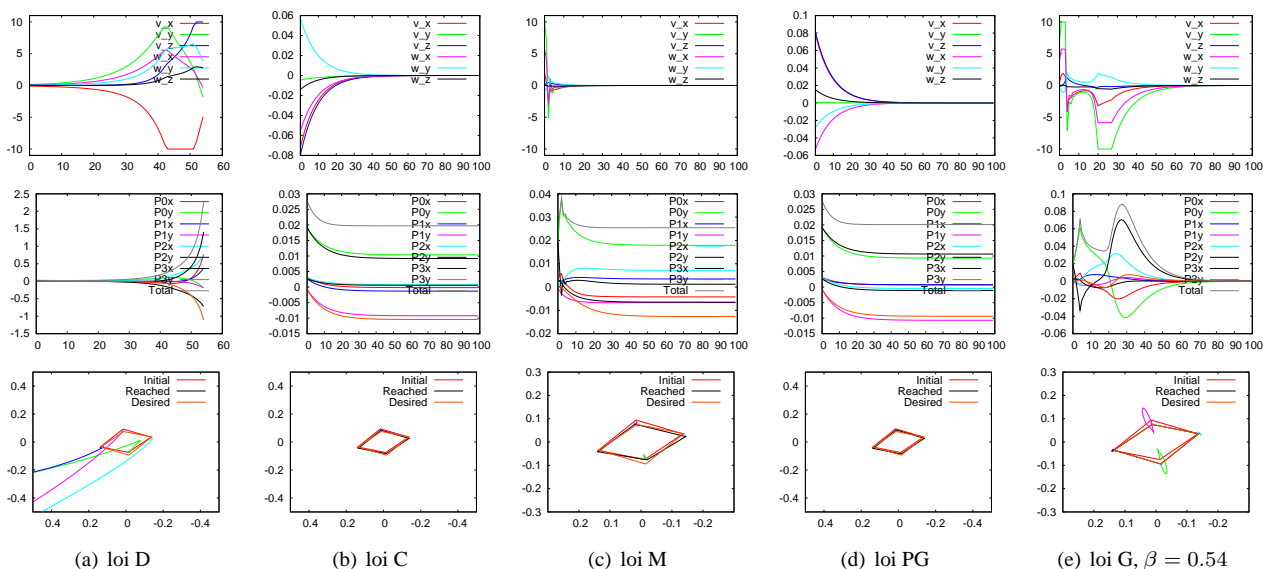


FIG. 2 – Résultats obtenus pour le cas 2

l'image. En fait, cette configuration correspond à un minimum local attractif pour la loi de commande PG alors qu'il en est autrement pour les autres lois. Comme attendu, toutes les composantes de la fonction de tâche e définie en (6) convergent exponentiellement vers zéro comme on peut le vérifier sur la Figure 3, mais, comme on l'a déjà dit, cela n'est pas suffisant pour obtenir un bon comportement du système.

Finalement, nous avons vérifié que la loi de commande G converge à la configuration désirée pour toutes les valeurs de β comprises dans l'intervalle $[-1.9; 1.04]$. On peut noter qu'il n'est donc pas forcément nécessaire que le paramètre β soit compris entre 0 et 1, et que le choix de β n'est pas crucial pour cette configuration.

Deuxième cas. La pose désirée entre la caméra et le carré a maintenant été fixée à $(0, 0, 1, 45, -30, 30)$. Dans ce cas, où la position désirée du plan image n'est pas parallèle à l'objet, la loi de commande C est connue pour être sujette à des minima locaux [2]. La pose initiale de la caméra est donnée par $(0, 0, 1, -46, 30, 30)$. Les résultats obtenus sont illustrés sur la Figure 2. Nous pouvons tout d'abord remarquer que la loi de commande D diverge totalement, le carré sortant entièrement et rapidement du champ de vue de la caméra. Les lois de commande C, M et PG convergent, elles, en un minimum local. Pour M, c'est la première fois, à notre connaissance, qu'un tel problème de minimum local est mis en évidence. Pour la nouvelle loi de commande PG, nous pouvons une fois de plus remarquer sur la Figure 3 que toutes les composantes de e convergent exponentiellement à zéro. Finalement, la nouvelle loi de commande G converge à la position désirée lorsque β est correctement choisi, à savoir $0.515 < \beta < 0.569$, ce qui est cohérent avec le fait que les lois de commande D, C, et M (où $\beta = 1, 0$ et $1/2$ respectivement) échouent. Comme on peut le voir sur la Figure 2, le comportement étonnant des vitesses de la caméra permet de sortir de la zone d'attraction du minimum local.

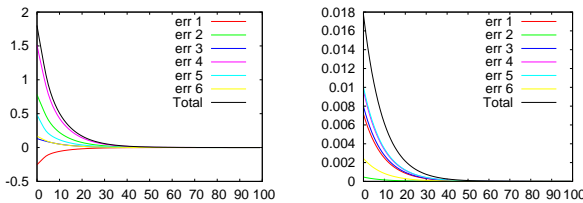


FIG. 3 – Evolution de la fonction de tâche e pour la loi de commande PG dans les cas 1 (à gauche) et 2 (à droite)

4 Etude analytique

Cette section présente une étude analytique des lois de commandes décrites précédemment lorsque le déplacement de la caméra est constitué d'une combinaison de translations et de rotations par rapport à l'axe optique de la caméra.

Comme précédemment, les informations visuelles utilisées sont les projections des sommets d'un carré dans le plan image. Le fait d'avoir choisi un objet symétrique ne modifie les comportements obtenus en simulation ou avec des expérimentations réelles. Il permet par contre d'aboutir à des formes analytiques relativement sympathiques et donc exploitables.

Soit (X, Y, Z) les coordonnées d'un point 3D dans le repère de la caméra. Les coordonnées de ce point dans le plan image sont données par $\mathbf{x} = (x, y)$ où $x = X/Z$ et $y = Y/Z$. Il est bien connu que la matrice d'interaction associée à \mathbf{x} est donnée par [7]

$$\mathbf{L}_x = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \quad (14)$$

En utilisant quatre points, le vecteur \mathbf{s} est défini par

$$\mathbf{s} = (x_0, x_1, x_3, x_3, y_0, y_1, y_2, y_3).$$

L'étude analytique inclut deux cas où le mouvement le long de l'axe z de la caméra varie de Z à Z^* et où $r_z = 90^\circ$ dans le premier cas et $r_z = 180^\circ$ dans le second. Dans les deux cas, le plan de l'objet est parallèle au plan image.

4.1 Cas 1 : $r_z = 90^\circ$ & $t_z = (Z \rightarrow Z^*)$

Les coordonnées des quatre points du carré exprimées dans le repère de la caméra pour la pose initiale et pour la pose désirée sont respectivement données par $p_{i0} = (-L, -L, Z)$, $p_{i1} = (-L, L, Z)$, $p_{i2} = (L, L, Z)$, $p_{i3} = (L, -L, Z)$, $p_{d0} = (-L, L, Z^*)$, $p_{d1} = (L, L, Z^*)$, $p_{d2} = (L, -L, Z^*)$ et $p_{d3} = (-L, -L, Z^*)$ (voir Figure 4). Soit $l = L/Z$ et $l^* = L/Z^*$. La valeur initiale de \mathbf{s} est donc

$$\mathbf{s}_i = (-l, -l, l, l, -l, l, l, -l),$$

sa valeur finale

$$\mathbf{s}^* = (-l^*, l^*, l^*, -l^*, l^*, l^*, -l^*, -l^*)$$

et on en déduit immédiatement

$$\mathbf{s}_i - \mathbf{s}^* = (-l+l^*, -l-l^*, l-l^*, l+l^*, -l-l^*, l-l^*, l+l^*, -l+l^*).$$

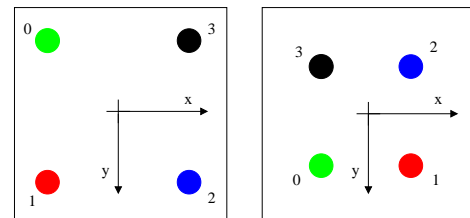


FIG. 4 – Configuration des quatre points dans le cas 1 : image initiale à gauche et image désirée à droite)

En utilisant la forme analytique de \mathbf{L}_x donnée par (14), il est possible de déterminer la valeur analytique de \mathbf{L}_β^+ à partir de la forme générale de \mathbf{L}_β donnée en (5). En utilisant

$Z = lZ^*/l^*$, on obtient ensuite après quelques calculs et simplifications :

$$\mathbf{L}_\beta^+ = \begin{bmatrix} -c_0 & -c_0 & -c_0 & -c_0 & -c_1 & c_1 & -c_1 & c_1 \\ -c_1 & c_1 & -c_1 & c_1 & -c_0 & -c_0 & -c_0 & -c_0 \\ -c_3 & c_4 & c_3 & -c_4 & c_4 & c_3 & -c_4 & -c_3 \\ -c_5 & c_5 & -c_5 & c_5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & -c_5 & c_5 & -c_5 \\ c_7 & c_6 & -c_6 & -c_7 & c_6 & -c_7 & -c_6 & c_7 \end{bmatrix}$$

où, quand $\beta \in [0; 1]$,

$$\begin{aligned} c_0 &= \frac{l^* Z^*}{4(\beta l^* + (1-\beta)l)} \\ c_1 &= \begin{cases} 0 & \text{si } \beta l^{*2} = (1-\beta)l^2 \\ c_0 \frac{\beta(1+l^{*2})+(1-\beta)(1+l^2)}{(\beta l^{*2} - (1-\beta)l^2)} & \text{sinon.} \end{cases} \\ c_3 &= \frac{l^* Z^* (\beta l^* + (1-\beta)l)}{8((1-\beta)^2 l^3 + \beta^2 l^{*3})}, \quad c_4 = \frac{l^* Z^* (\beta l^* - (1-\beta)l)}{8((1-\beta)^2 l^3 + \beta^2 l^{*3})} \\ c_5 &= \begin{cases} 0 & \text{si } \beta l^{*2} = (1-\beta)l^2 \\ \frac{-1}{4(\beta l^{*2} - (1-\beta)l^2)} & \text{sinon.} \end{cases} \\ c_6 &= \frac{\beta l^{*2} + (1-\beta)l^2}{8((1-\beta)^2 l^3 + \beta^2 l^{*3})}, \quad c_7 = \frac{\beta l^{*2} - (1-\beta)l^2}{8((1-\beta)^2 l^3 + \beta^2 l^{*3})} \end{aligned}$$

En utilisant la valeur de $\mathbf{s}_i - \mathbf{s}^*$ donnée ci-dessus, on peut facilement en déduire la vitesse initiale \mathbf{v}_i de la caméra :

$$\mathbf{v}_i = (0, 0, v_z, 0, 0, \omega_z) \quad (15)$$

où

$$\begin{cases} v_z &= \frac{\lambda Z^* l^* (\beta l^{*2} - (1-\beta)l^2)}{\beta^2 l^{*3} + (1-\beta)^2 l^3} \\ \omega_z &= \frac{\lambda l l^* (\beta l^* + (1-\beta)l)}{\beta^2 l^{*3} + (1-\beta)^2 l^3} \end{cases}$$

Comme attendu, le mouvement initial de la caméra consiste seulement à effectuer une translation le long de l'axe optique et une rotation autour de cet axe.

Nous pouvons tout d'abord remarquer que \mathbf{L}_β est singulière lorsque $\beta l^{*2} = (1-\beta)l^2$ puisqu'on a alors $c_5 = 0$, ce qui entraîne que les quatrième et cinquième lignes de \mathbf{L}_β^+ sont entièrement composées de 0. Une telle singularité se produit par exemple quand $l = l^*$ (c'est-à-dire $Z = Z^*$) et $\beta = 1/2$, ce qui est très surprenant. Ainsi la loi de commande proposée en [12] est singulière pour une rotation pure de 90° . En fait, la seule façon d'éviter cette singularité quelle que soit la valeur de l et l^* est de choisir $\beta = 0$ ou $\beta = 1$. A partir de (15), nous pouvons remarquer que cette singularité n'a pas d'effet sur les vitesses calculées dans des conditions parfaites (puisque $\omega_x = \omega_y = 0$), mais un comportement instable est à craindre en présence de bruit dans l'image ou pour des configurations proches de cette singularité (par exemple, quand le plan de l'objet est presque parallèle au plan image). Ceci sera mis en évidence sur les résultats expérimentaux présentés par la suite.

Pour le cas particulier d'une rotation pure, c'est-à-dire quand $Z = Z^*$ et $l = l^*$, la vitesse initiale \mathbf{v}_i devient

$$\mathbf{v}_i = \left(0, 0, \frac{\lambda Z^* (2\beta - 1)}{2\beta^2 - 2\beta + 1}, 0, 0, \frac{\lambda}{2\beta^2 - 2\beta + 1}\right) \quad (16)$$

Dans ce cas classique, on remarque que la vitesse \mathbf{v}_i contient des translations inattendues dont la direction dépend de la valeur de β : $v_z < 0$ si $\beta < 1/2$ et $v_z > 0$ si

$\beta > 1/2$. La seule façon d'éviter cette translation est de choisir $\beta = 1/2$ comme cela a déjà été montré dans [12] (mais où la singularité n'avait pas été établie).

Dans le cas général où $Z \neq Z^*$, et en sélectionnant $\beta = 1$ dans $\widehat{\mathbf{L}}_s^+$, la vitesse initiale \mathbf{v}_i obtenue pour la loi de commande D est donnée par

$$\mathbf{v}_i = (0, 0, \lambda Z^*, 0, 0, \frac{\lambda l^*}{l^*}) \quad (17)$$

Quelle que soit la valeur de Z et même si $Z < Z^*$, c'est-à-dire quand la caméra doit se déplacer vers l'arrière, le mouvement initial de la caméra contient un terme de translation vers l'avant. Ce résultat étonnant généralise la même propriété obtenue par le passé, mais pour le cas simple d'une rotation pure (où $Z = Z^*$) [4].

En sélectionnant à présent $\beta = 0$, la vitesse initiale \mathbf{v}_i de la loi de commande C est donnée par

$$\mathbf{v}_i = (0, 0, \frac{-\lambda l^* Z^*}{l}, 0, 0, \frac{\lambda l^*}{l}) \quad (18)$$

Dans ce cas, le mouvement initial de la caméra contient toujours un terme de translation vers l'arrière et ce quelle que soit la valeur initiale de Z , c'est-à-dire même quand $Z \geq Z^*$. On peut même noter que plus l est petit, et donc Z grand, plus la vitesse initiale vers l'arrière est importante, ce qui est encore plus surprenant que le résultat obtenu avec $\beta = 1$. Ces résultats étendent donc largement la propriété de mouvement de recul montrée dans [5] dans le cas d'une rotation pure. Finalement, en comparant (17) et (18), nous pouvons aussi remarquer que les amplitudes du mouvement de rotation obtenues par D et C ne sont étonnamment pas les mêmes lorsque $l \neq l^*$, c'est-à-dire lorsque $Z \neq Z^*$.

Enfin, avec $\beta = 1/2$, la vitesse \mathbf{v}_i obtenue pour la loi de commande M est donnée par

$$\mathbf{v}_i = \left(0, 0, \frac{2\lambda Z^* l^* (l^{*2} - l^2)}{l^3 + l^{*3}}, 0, 0, \frac{2\lambda l l^* (l + l^*)}{l^3 + l^{*3}}\right) \quad (19)$$

Dans ce cas, un bon comportement est obtenu puisque le mouvement de translation est toujours dans la direction attendue ($v_z < 0$ quand $l^* < l$, c'est-à-dire quand $Z < Z^*$, $v_z > 0$ quand $l^* > l$ ($Z > Z^*$), et, comme on l'a déjà dit, $v_z = 0$ quand $l = l^*$ (où $Z = Z^*$)).

Finalement, la vitesse \mathbf{v}_i de la nouvelle loi de commande PG est

$$\mathbf{v}_i = (0, 0, \frac{-\lambda l^* Z^*}{l}, 0, 0, \frac{\lambda l^*}{l}) \quad (20)$$

soit exactement la même vitesse que celle donnée par (18) avec la loi de commande C.

4.2 Cas 2 : $r_z = 180^\circ$ & $t_z = (Z \rightarrow Z^*)$

Nous considérons à présent le cas encore plus problématique d'un mouvement de la caméra composé d'une rotation de 180° autour de son axe optique et d'un mouvement de translation le long de cet axe. Dans ce cas, on a (voir Figure 5)

$$\mathbf{s}_i - \mathbf{s}^* = (l+l^*, -l-l^*, -l-l^*, l+l^*, -l-l^*, -l-l^*, l+l^*, l+l^*)$$

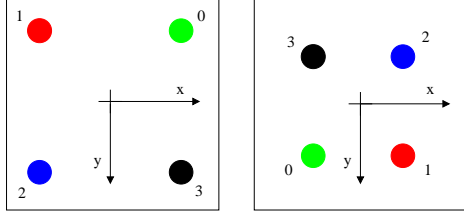


FIG. 5 – Configuration des quatre points dans le cas 2 : image initiale à gauche et image désirée à droite)

La matrice \mathbf{L}_β^+ est alors donnée par

$$\widehat{\mathbf{L}}_s^+ = \begin{bmatrix} -c_0 & -c_0 & -c_0 & -c_0 & -c_1 & c_1 & -c_1 & c_1 \\ -c_1 & c_1 & -c_1 & c_1 & -c_0 & -c_0 & -c_0 & -c_0 \\ -c_3 & c_3 & c_3 & -c_3 & c_3 & c_3 & -c_3 & -c_3 \\ -c_4 & c_4 & -c_4 & c_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & -c_4 & c_4 & -c_4 \\ c_5 & c_5 & -c_5 & -c_5 & c_5 & -c_5 & -c_5 & c_5 \end{bmatrix}$$

où, quand $\beta \in [0; 1]$,

$$c_0 = \frac{l^* Z^*}{4(\beta l^* + (1-\beta)l)}, \quad c_1 = c_0 \frac{\beta(1+l^{*2}) + (1-\beta)(1+l^2)}{\beta l^{*2} + (1-\beta)l^2}$$

$$c_3 = \begin{cases} 0 & \text{si } \beta l^{*2} = (1-\beta)l^2 \\ \frac{l^* Z^*}{8(\beta l^{*2} - (1-\beta)l^2)} & \text{sinon} \end{cases}$$

$$c_4 = \frac{1}{4(\beta l^{*2} + (1-\beta)l^2)}$$

$$c_5 = \begin{cases} 0 & \text{si } \beta l^* = (1-\beta)l \\ \frac{1}{8(\beta l^* - (1-\beta)l)} & \text{sinon} \end{cases}$$

En procédant comme précédemment et en utilisant la valeur de $\mathbf{s}_i - \mathbf{s}^*$, on obtient

$$\mathbf{v}_i = (0, 0, v_z, 0, 0, 0) \quad (21)$$

où

$$v_z = \begin{cases} 0 & \text{si } \beta l^{*2} = (1-\beta)l^2 \\ \frac{\lambda Z^* l^* (l+l^*)}{\beta l^{*2} - (1-\beta)l^2} & \text{sinon.} \end{cases}$$

Dans tous les cas, la commande ne produit aucun mouvement de rotation. Par contre, elle produit toujours un mouvement de translation, sauf quand $\beta l^{*2} = (1-\beta)l^2$ auquel cas \mathbf{L}_β est singulière, amenant à un minimum local répulsif où $v_z = 0$. Une telle situation se produit par exemple quand $Z = Z^*$ (c'est-à-dire $l = l^*$) et $\beta = 1/2$, ce qui correspond à la loi de commande proposée dans [12].

Une autre configuration de singularité se produit également quand $\beta l^* = (1-\beta)l$, soit notamment lorsque $l = l^*$ et $\beta = 1/2$.

Bien sûr, pour une rotation pure ($Z = Z^*$), nous retrouvons les résultats de [2] : un mouvement de translation vers l'avant est obtenu pour $\beta = 1$ et un mouvement de translation vers l'arrière est obtenu pour $\beta = 0$. Plus généralement, pour $\beta = 1$ et $\beta = 0$, la direction du mouvement est la même (c'est-à-dire respectivement vers l'avant et vers l'arrière) quelle que soit la valeur de l et l^* , soit quelle que soit la valeur de Z par rapport à Z^* . Pour les autres valeurs

de β , la direction du mouvement dépend de la valeur relative de Z par rapport à Z^* , mais malheureusement, aucune valeur de β n'amène à un bon comportement dans cette situation parce qu'aucun mouvement de rotation n'est produit par la loi de commande.

Finalement, les résultats obtenus par la loi de commande PG ne sont pas meilleurs que ceux obtenus avec les autres lois puisque nous avons dans ce cas

$$\mathbf{v}_i = \left(0, 0, \frac{-\lambda l^* Z^* (l+l^*)}{l^2}, 0, 0, 0\right) \quad (22)$$

ce qui est la même vitesse que ce qui a été obtenu quand $\beta = 0$.

5 Résultats

Dans cette section, nous analysons le comportement de la caméra via des résultats de simulation et des résultats expérimentaux obtenus à l'aide d'un système constitué d'une caméra embarquée sur un robot à six degrés de liberté. La pose initiale de la caméra a été choisie égale à $(0, 0, 1, 0, 0, r_z)$ et sa pose désirée à $(0, 0, 0.5, 0, 0, 0)$. La caméra doit donc avoir un mouvement de translation de 0.5 mètre vers l'avant et effectuer simultanément un mouvement de rotation autour de son axe optique de r_z degrés où r_z varie de 0 à 180. Les résultats obtenus pour trois valeurs particulières de r_z (120° , 170° et 180°) sont donnés respectivement sur les figures 6, 7 et 8. Comme dans la section précédente, nous avons considéré les quatre sommets d'un carré dont la demi-arête mesure $L = 0.1$ m, ce qui implique que $l^* = 0.2$ et $l = 0.1$. Nous avons aussi posé $\lambda = 0.1$ et des termes de saturation ont été introduits dans les composantes de la vitesse pour interdire l'application d'une vitesse trop importante. Une telle situation peut se produire quand \mathbf{L}_β est mal conditionnée, c'est-à-dire à proximité des configurations singulières. Plus précisément, toutes les composantes de la vitesse sont normalisées si besoin afin que la vitesse maximale ne soit pas supérieure à 0.1 m/s ou 10 deg/s.

En appliquant la loi de commande D, la caméra se déplace correctement vers la pose désirée, c'est-à-dire sans dépassement en translation, tant que $r_z \leq 78^\circ$. Quand $r_z > 78^\circ$, la caméra continue son mouvement en translation après avoir atteint $Z = Z^*$ puis converge à la pose désirée après un mouvement vers l'arrière. Ce phénomène est visible sur la Figure 6.a. Il s'amplifie quand r_z augmente et, en simulation, la loi de commande échoue quand $r_z \geq 155^\circ$ car la caméra atteint alors le plan de l'objet où $Z = 0$. En pratique, la loi de commande échoue pour une valeur plus faible de r_z lorsque les points sortent du champ de vue de la caméra, ce qui se produit avant que la caméra n'atteigne le plan de l'objet (!) Finalement, comme on peut le voir sur la Figure 6.a, v_z est saturée dès la première itération de la loi de commande, et ω_z n'est saturée qu'après plusieurs itérations.

En appliquant la loi de commande C, la caméra a un mouvement correct tant que $r_z \leq 61^\circ$. Quand $r_z > 61^\circ$, la

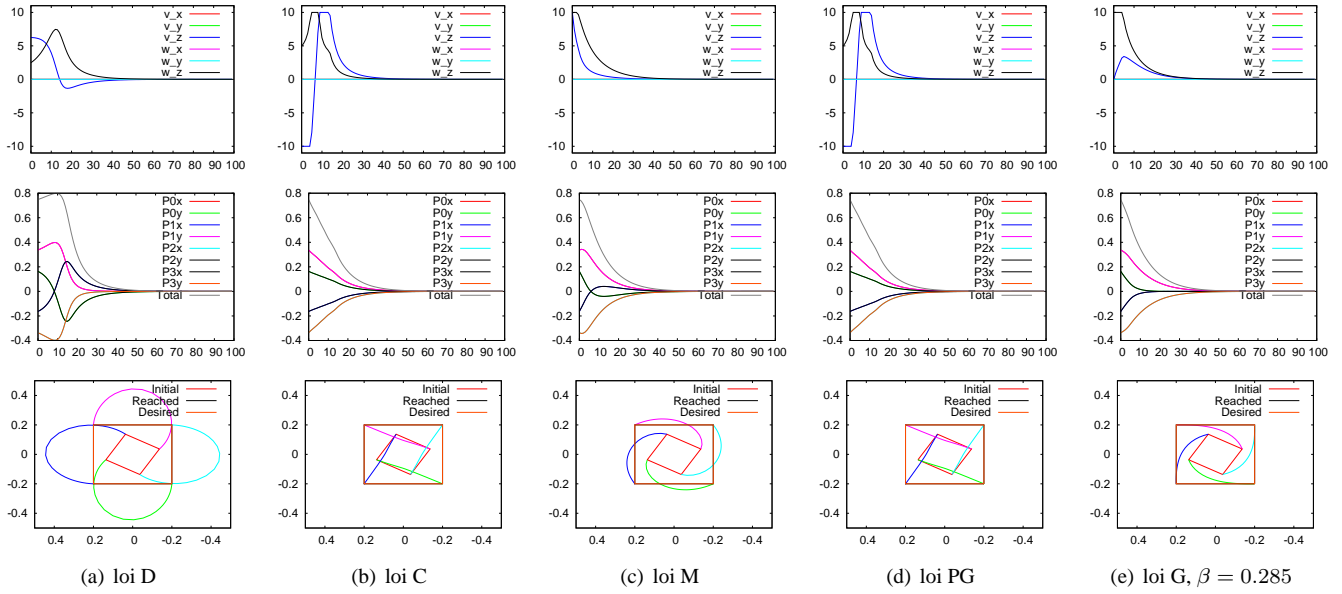


FIG. 6 – Résultats de simulation obtenus dans le cas où $t_z = 0.5$ m et $r_z = 120^\circ$. Première ligne : vitesse de la caméra (en cm/s et dg/s), deuxième ligne : erreurs des points de l’image (en m), troisième ligne : trajectoires des points dans l’image

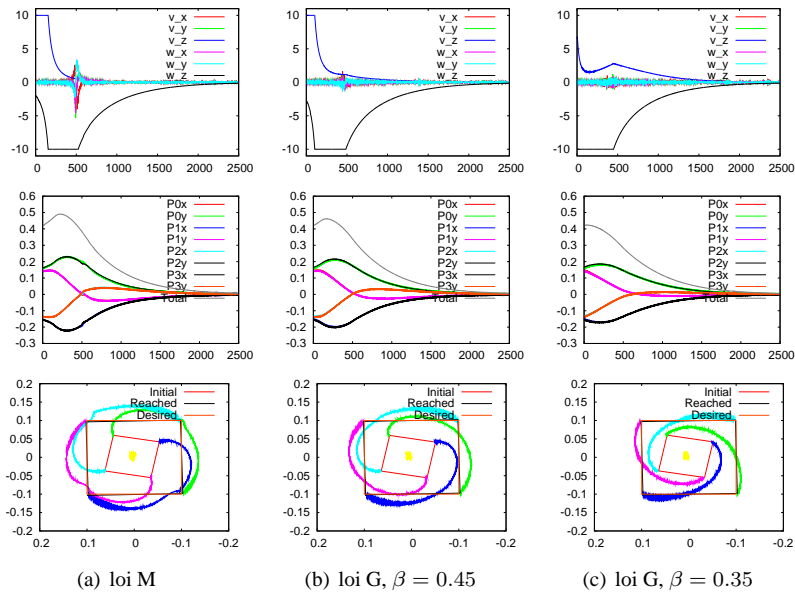


FIG. 7 – Résultats expérimentaux obtenus dans le cas où $t_z = 0.5$ m et $r_z = 170^\circ$.

caméra commence à se déplacer vers l’arrière et revient ensuite vers l’avant. La translation vers l’arrière est d’autant plus forte que r_z augmente. Nous pouvons remarquer sur la Figure 6.b que la vitesse en rotation est saturée lorsque le mouvement de translation change de sens. On peut par ailleurs remarquer sur la Figure 6.d que la loi de commande PG donne exactement le même comportement. En simulation, le nombre d’itérations nécessaires pour atteindre la pose désirée augmente rapidement quand $r_z > 150$. Finalement, quand $r_z \geq 178.6^\circ$, la translation en arrière est trop importante pour que le système puisse

converger. En pratique, la loi de commande échoue pour une valeur plus faible de r_z , le robot atteignant les limites de son espace de travail en raison du mouvement de recul.

Comme cela peut être vérifié sur la Figure 6.c, la loi de commande M converge avec un comportement parfait (c’est-à-dire sans aucune translation supplémentaire) pour tout $r_z < 180^\circ$. Comme cela a déjà été dit, la loi de commande M est en singularité quand $r_z = 180^\circ$, c’est pourquoi les composantes de la vitesse sont saturées au début de l’asservissement pour de grandes valeurs de r_z .

En appliquant la nouvelle la loi de commande G, des com-

portements différents sont obtenus selon la valeur choisie pour β . Quand la valeur de β est proche de 0, 1 et 1/2, le comportement de la loi de commande est similaire à celui des lois de commande D, C et M. Un bon choix de β conduit à une amélioration du comportement de la loi de commande pour un déplacement donné. Par exemple, dans le cas où $r_z = 120^\circ$, la loi de commande G permet à la caméra d'atteindre la pose désirée quand $\beta \in [-0.08, 1.19]$ avec le meilleur comportement obtenu quand $\beta = 0.285$ (voir Figure 6.e). Dans ce cas, la vitesse en rotation ω_z atteint sa valeur maximale à la première itération. L'erreur des coordonnées de chaque point commence aussi à décroître dès la première itération.

Les résultats expérimentaux présentés sur les Figures 7 et 8 confirment ceux obtenus en simulation. Pour une rotation de 170° et une translation de 0.5 m à effectuer, les commandes D, C et PG pour des problèmes de mouvement d'avancée ou de recul trop importants pour être réalisés. La loi de commande M converge, elle. On peut toutefois noter sur la Figure 7.a l'effet de la singularité en 90° qui entraîne des oscillations assez importantes sur les composantes v_x, v_y, ω_x et ω_y de la vitesse de la caméra. Pour la commande G, la caméra atteint sa pose désirée lorsque $\beta \in [0.33, 0.85]$. On peut noter la diminution de l'effet de la singularité lorsque $\beta = 0.45$ (voir Figure 7.b), cet effet disparaissant quasiment lorsque $\beta = 0.35$ (voir Figure 7.c). Enfin, pour le cas d'une rotation de 180° , seule la loi de commande M (soit encore G avec $\beta = 0.5$) permet d'atteindre la convergence. Comme on l'a vu lors de l'analyse analytique, aucun mouvement de rotation ne devrait être appliqué, ce que l'on peut observer sur la Figure 8 pour les premières itérations de la loi de commande. Cependant, en pratique, le moindre bruit de mesure permet de sortir de ce minimum local répulsif et d'atteindre ensuite la convergence, après avoir approché sans trop d'encombres la singularité à 90° .

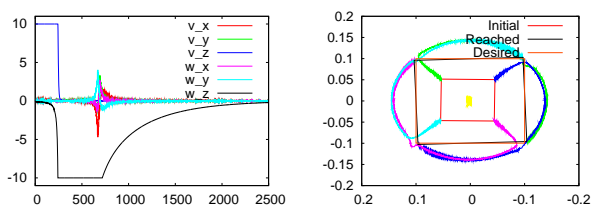


FIG. 8 – Résultats expérimentaux obtenus dans le cas où $t_z = 0.5$ m et $r_z = 180^\circ$: composantes de la vitesse de la caméra en cm/s et dg/s (à gauche) et trajectoires des points dans l'image (à droite)

6 Conclusion et perspectives

Les diverses lois de commande utilisées en asservissement visuel ont chacune leurs inconvénients et avantages respectifs. Dans cet article, nous avons mis en évidence, pour la première fois à notre connaissance, un minimum local pour la loi de commande proposée dans [12]. Celui-

ci a été obtenu en étudiant une nouvelle loi de commande construite pour tenter d'obtenir la stabilité asymptotique globale. Nous avons aussi accompli une étude analytique dans le cas d'un mouvement composé d'une translation le long de l'axe optique de la caméra et d'une rotation autour de cet axe. L'ensemble des résultats théoriques présentés ont été validés par des simulations et des expérimentations sur le robot de l'Irisa.

La nouvelle loi de commande basée sur un paramètre de comportement β permet, en sélectionnant β à 0, 1 ou 1/2, de basculer entre les schémas classiques d'asservissement visuel. D'autres valeurs de β peuvent être choisies et permettent alors de modifier le comportement de la loi de commande. Ce paramètre offre donc une plus large gamme d'applicabilité. La valeur idéale pour le paramètre de comportement dépend du mouvement que la caméra doit réaliser. Le problème restant est donc de pouvoir déterminer automatiquement la valeur de ce paramètre pour obtenir un bon comportement dans tous les cas. Cette perspective va faire l'objet de nos travaux futurs. On peut également penser à modifier la valeur du paramètre pendant l'exécution de la tâche pour s'adapter en ligne au comportement réalisé.

Références

- [1] E. Cervera, A. Pobil, F. Berry and P. Martinet, "Improving image-based visual servoing with three-dimensional features", *Int. Journal of Robotics Research*, 22 :821-839, Oct. 2003.
- [2] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing", *The Conference of Vision and Control. LNCIS 237*, pp 66-78, Springer Verlag, 1998.
- [3] F. Chaumette, "Image moments : a general and useful set of features for visual servoing", *IEEE Trans. on Robotics and Automation*, 20(4) :713-723, Aug. 2004.
- [4] F. Chaumette and S. Hutchinson, "Visual servo control Part I : basic approaches", *IEEE Robotics and Automation Magazine*, 13(4) :82-90, Dec. 2006.
- [5] P. Corke and S. Hutchinson, "A new partitioned approach to image-based visual servo control", *IEEE Trans. on Robotics and Automation*, 17(4) :507-515, Aug. 2001.
- [6] L. Deng, F. Janabi-Sharifi and W. Wilson, "Hybrid strategies for image constraints avoidance in visual servoing", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'02*, Lausanne, pp 348-353, Oct. 2002.
- [7] B. Espiau, F. Chaumette and P. Rives, "A new approach to visual servoing in robotics", *IEEE Trans. on Robotics and Automation*, 8(3) :313-326, June 1992.
- [8] N. Gans and S. Hutchinson, "An experimental study of hybrid switched system approach to visual ser-

voing", *IEEE Int. Conf. on Robotics and Automation, ICRA'03*, Taiwan, pp 3061-3068, Sep. 2003.

- [9] N. Gans, S. Hutchinson and P. Corke, "Performance tests for visual servo control systems, with application to partitioned approaches to visual servo control", *Int. Journal of Robotics Research*, 22 :955-981, Oct. 2003.
- [10] S. Hutchinson, G. Hager and P. Corke, "A tutorial on visual servo control", *IEEE Trans. on Robotics and Automation*, 12(5) :651-670, Oct. 1996.
- [11] E. Malis, F. Chaumette and S. Boudet, "2 1/2 D visual servoing", *IEEE Trans. on Robotics and Automation*, 15(2) :238-250, April 1999.
- [12] E. Malis, "Improving vision-based control using efficient second-order minimization techniques", *IEEE Int. Conf. on Robotics and Automation (ICRA'04)*, New Orleans, pp 1843-1848, Apr. 2004.
- [13] P. Martinet, J. Gallice and D. Khadraoui, "Vision based control law using 3d visual features", *Proc. WAC 96*, Vol. 3, pp. 497-502, Montpellier, May 1996.
- [14] F. Janabi-Sharifi and W. Wilson, "Automatic selection of image features for visual servoing", *IEEE Trans. on Robotics and Automation* 13(6) :890-903, Dec. 1997.
- [15] W. Wilson, C. Hulls and G. Bell, "Relative End-Effector Control Using Cartesian Position Based Visual Servoing", *IEEE Trans. on Robotics and Automation*, 12(5) :684-696, Oct. 1996.