

# Rapport mi-parcours PFE - Dietrich - Dubost

## I - Description du sujet

L'ENSEM dispose de matériel très performant mais sous-utilisé. Nous souhaitons pallier à ce problème en travaillant avec les *Quadcopters* Parrot et le système de motion tracking par caméra infrarouge. Ce sujet de PFE a été motivé par une vidéo montrant les applications possibles de ce genre de systèmes. De plus nous souhaitons réaliser des stages liant l'aéronautique et l'automatique et ce sujet d'avoir un premier contact avec ce milieu.

Notre projet consiste à modéliser et stabiliser le drone, mesurer son état grâce au système de caméras. Une fois le modèle et la commande établis nous souhaitons diriger le drone à l'aide d'une baguette.

L'objectif à terme est de réaliser à l'instar des robots Khepera une bibliothèque Matlab permettant d'utiliser ce système plus simplement.

## II - Travail réalisé

Au cours de nos lectures préliminaires nous avons acquis des connaissances en mécanique du vol pour des aéronefs à voilure mobile. Nous avons également regardé les différentes méthodes de modélisation et de contrôle qui sont utilisées pour piloter ces systèmes, les commandes sont généralement réalisées en puissance. Nous savons qu'il est possible de caractériser l'état d'un drone en connaissant les variables suivantes :

$$X = [x \ y \ z \ \theta \ \psi \ \varphi \ \dot{x} \ \dot{y} \ \dot{z} \ \dot{\theta} \ \dot{\psi} \ \dot{\varphi}]^T$$

Nous devons donc être capables de les mesurer. Le contrôle, d'un engin à quatre rotors se fait grâce à la commande de chaque moteur de façon indépendante :

$$U = [u_1 \ u_2 \ u_3 \ u_4]^T$$

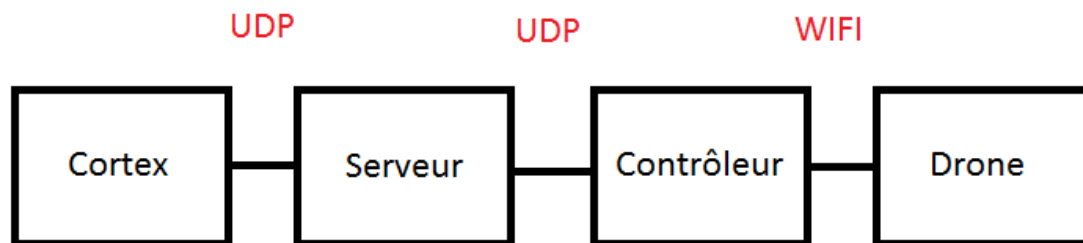
Le plus gros problème auquel nous faisons face actuellement est le contrôle du drone. Le SDK (environnement de développement) du drone nous permet de contrôler  $\{\varphi \ \theta \ \dot{z} \ \dot{\psi}\}$ . Cependant dans cette version du drone il n'est pas possible de contrôler chaque moteur de manière indépendante. Le logiciel embarqué est destiné principalement à des développeurs d'applications à but ludique. Bien que le système soit ouvert, nous ne pouvons pas accéder à des couches hardware assez basse pour faire de la régulation.

Nous avons trouvé un modèle Simulink décrivant le comportement du drone. Le système de contrôle/commande utilise les gyroscopes, la caméra ventrale et les accéléromètres pour estimer les différents paramètres du drone. Cependant la précision de ces paramètres est approximative, vis-à-

vis des résultats donnés par le système de tracking à notre disposition. C'est pourquoi nous avons choisi d'utiliser l'association des caméras de tracking et du logiciel cortex pour avoir plus de précision.

Durant nos premières séances nous sommes parvenus à récupérer les données issues des caméras directement sous Matlab, nous pouvons voir en quasi temps réel (environ 10ms de latence) la position, l'orientation et les différentes vitesses linéaires/rotations. Cependant lorsque nous avons tenté d'implémenter ces données dans le modèle que nous avons trouvé, cette opération s'est avérée impossible. Certains blocs de simulations ne sont pas compatibles entre eux.

Pour pallier à ce problème nous nous avons décidé d'utiliser une technique non optimale pour pouvoir utiliser ces données sous Simulink. Un ordinateur était chargé de récupérer les informations provenant de Cortex et de les renvoyer sur les réseaux. Un autre ordinateur lui contrôlait le drone, l'explication sera plus claire sur le schéma ci-dessous.



Cependant avec la multiplication des temps de transfert et le temps d'exécution de Matlab, les données ne sont pas rafraichies assez régulièrement. Cette solution n'est donc pas viable.

Actuellement nous sommes confrontés à un autre problème, nous pensons que toutes nos variables ne sont pas exprimées dans les mêmes repères. En effet Cortex utilise un repère qui lui est propre, tandis que le drone en possède également un. Nous devons réussir à identifier dans quel repère sont exprimées les variables que nous utilisons sous Simulink. Au vu de ces difficultés il paraît donc plus judicieux de traduire les schémas bloc du modèle Simulink en un programme Matlab. C'est ce qui nous occupe actuellement.

### III - Référence bibliographique

De Lellis Costa de Oliveira, Marcelo. *Modeling, identification and control design for a micro quadrotor*. Mémoire de Master. Control Engineering. Prague : Czech Technical University, 2011

Bresciani, Tommaso. *Identification and control of a quadrotor helicopter*. Mémoire de Master. Control Engineering. Lund : Lund University, 2008

Piskorski S., Brulez N., Eline P.. *AR Drone Developer Guide SDK 1.7*. [En ligne] Parrot, 2011. Disponible sur [[http ://projects.ardrone.org](http://projects.ardrone.org)]. Consulté le (15.10.2013)