



**Mobile Robots**  
**Summer Semester 2013**  
**Assignment 4**

due date: 28.05.2013, presentation: 06.06.2013  
email submissions: markus.ryll@tuebingen.mpg.de

**Exercise 1 (4 Punkte)**

Ziel des ersten Teils des Assignments ist es die Simulationsumgebung Matlab-Simulink zu installieren und das Model zu starten.

- Installieren Sie MatLab.
- Laden Sie die Zip-Datei Assignment04.zip herunter und extrahieren Sie diese in einen beliebigen Ordner.
- Starten Sie Matlab und wechseln Sie das 'Current Directory' in den Ordner, in welchen Sie die Zip-Datei entpackt haben.
- Oeffnen Sie ueber die 'Current Directory'-Leiste das Simulink-Model 'systema.mdl'. Es oeffnet sich nun die Simulink-Umgebung.
- Doppelklicken Sie auf das Objekt 'optsiminit'. Damit werden Konstanten geladen. Sie koennen diese in der MatLab-Umgebung im Workspace sehen.
- Starten Sie das Model indem Sie in der Simulink-Umgebung auf den 'Play-Button' klicken. Die Simulation wird nun fuer 20 Sekunden ausgefuehrt.
- Machen Sie sich mit Simulink vertraut. Durch einen Doppelklick auf die beiden Scopes 'Position' und 'Orientation' koennen Sie das Verhalten des Quadcopters ueber die Zeit beobachten.

(a) Geben Sie an, welche Position und Orientierung der Quadcopter nach Ablauf der 20 sekunden Simulation hat.

**Exercise 2 (6 Punkte)**

Die Simulation ist in 4 verschiedene Bloecke gegliedert: Control, Motors+Modules, Aero, Dynamics

- Control: Enthaelte den Controller des Quadcopters.
- Motors+Modules: Enthaelte die Simulation der Motoren.
- Aero: Enthaelte die aerodynamische Simulation (wie in der Vorlesung besprochen).
- Dynamics: Enthaelte die dynamische Simulation des Quadcopter (wie in der Vorlesung besprochen).

Bei einem Doppelklick auf die einzelnen Bloecke koennen sie sehen, welcher in diesen aufgerufen wird. Der Quellcode fuer die dynamische Simulation befindet sich in der Datei 'dinamica.m', die Sie ueber das 'Current Directory' in der Matlabumgebung oeffnen koennen.

- (a) Ab der Zeile 96 (in 'dinamica.m') werden die Kraefte ( $Y_aA=Y$ -Actuator action,  $Y_hF=Y$ -Hub force) in Y-Richtung auf den Quadcopter berechnet. Diese sind im aktuellen Quellcode zu Null gesetzt. Reimplementieren Sie diese Kraefte analog der Kraefte in X-Richtung mit Hilfe der Unterlagen aus der Vorlesung. Geben Sie die neuen Gleichungen an.
- (b) Lassen Sie die Simulation erneut laufen und geben Sie die neue finale Position und Orientierung des Quadcopter nach Ablauf der 20 sekundigen Simulation an.
- (c) Der Block 'Aero' beschreibt die aerodynamischen Effekte. Geben Sie an was die Eingaenge und was die Ausgaenge des Blockes sind. Schaeuen Sie sich hierzu die Datei 'aero.m' an.

### **Exercise 3 (6 Punkte)**

Der Controller des Modells regelt die Fluggeschwindigkeit ( $\dot{x}_d$  = gewuenschte Geschwindigkeit in X-Richtung,  $\dot{y}_d$ = gewuenschte Geschwindigkeit in Y-Richtung) die Flughoehe ( $Z_d$ ) und Orientierung um die Z-Achse (Yawd). Sie sehen dies als vier konstante Eingangsblocke des 'Control'-Blocks (Eingaenge In2 - In5).

- (a) Entwerfen Sie eigenstaendig einen neuen Eingangsblock fuer die gewuenschte Flughoehe ( $Z_d$ ) der folgende Werte ueber die Zeit vorgibt:  $t(0)=2m$ ,  $t(5)=3m$ ,  $t(10)=2.5m$ ,  $t(15)=2m$ ,  $t(20)=2m$ . Die Werte des Blocks sollen ueber die Zeit stetig sein. Geben Sie die Struktur oder den Quellcode Ihres neuen Blockes an. Zeigen Sie einen Plot der Z-Position ueber die Zeit.

### **Exercise 4 (4 Punkte)**

- (a) In der Vorlesung wurde verschiedene aerodynamische Effekte besprochen. Beschreiben Sie in eigenen Worten die Ursache des Rolling-Moments und die groessten Einflussfaktoren auf dieses.
- (b) Beschreiben Sie, warum im Controller des Kontroll-Schemas oft nur die Actuator Action modelliert wird und die aerodynamischen Effekte auBen vor gelassen werden. Wie kommt es, dass ein Quadcopter trotz der unvollstaendigen Beschreibung im Controller steuerbar ist?

Viel Spass!