TTK4190 Guidance and Control of Vehicles

Assignment 1

Kinematics and Kinetics

Fall 2019

Deadline Monday the 16th of September at 23:59

Objective

You will learn how to implement and simulate the attitude kinematics of a rigid body in space, but the theory is also applicable for underwater vehicles, robotic systems and unmanned aerial vehicles. You will also simulate the horizontal motion of a ship, learn the Nomoto model for heading and solve a path-following problem.

Grading

This assignment must be passed to get access to the final exam. The overall impression of how well you have understood the problems will be the basis for the evaluation. You need at least 60 % of the assignment correct to pass. **You are encouraged/supposed to work in groups of 2-4 people, but are allowed to do the assignment individually if that is preferred**. Note that the grading will be equally severe if you do the work individually. The participants in the group will receive the same feedback.

Deadline and Delivery Details

The assignment must be handed in by 23:59 on Monday the 16th of September. You must deliver a report and produce your own Matlab files for each simulation problem. Simulink cannot be used. Matlab m-files should not be included in the delivery. Make sure all plots clearly show the required data (title, legend, label and tag the axes, set the grid on and so forth). An example of how to simulate the attitude dynamics of a rigid-body (using unit quaternions) is demonstrated in the file attitude.m, which is posted on Blackboard and displayed in Appendix A.1. It is recommended to use this file as a template and change the necessary parts of the code. You are encouraged to use and look into the MSS toolbox (wwww.marinecontrol.org), which includes several useful functions for this assignment and you need the toolbox to run the template. The report has to be handed in via Blackboard. You are strongly urged to write the report on your PC using your favorite editor (LaTeX, Word, Pages...). You can hand in a scanned version (not recommended), but in the end it has to be a PDF document. Paper versions are not accepted. A latex template for the report is published on Blackboard, but it is not necessary to use the template.

Problem 1 - Attitude Control of Satellite (65 %)

Consider a satellite with inertia matrix $\mathbf{I}_{CG} = mr^2\,\mathbf{I}_3$, $m = 180$ kg, $r = 2.0$ m. The equations of motion are:

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{T}_q(\mathbf{q})\omega \\ \mathbf{I}_{CG}\dot{\omega} - \mathbf{S}(\mathbf{I}_{CG}\omega)\omega &= \tau \end{aligned} \tag{1}$$

Problem 1.1 What is the equilibrium point $\mathbf{x}_0$ of the closed-loop system $\mathbf{x} = [\varepsilon^\top, \omega^\top]^\top$ corresponding to $\mathbf{q} = [\eta, \varepsilon_1, \varepsilon_2, \varepsilon_3]^\top = [1,0,0,0]^\top$ and $\tau = \mathbf{0}$? It is not necessary to include the state η in the analysis since it is a function of ε, that is $\eta = \sqrt{1-\varepsilon^\top\varepsilon}$. Linearize the spacecraft model about $\mathbf{x} = \mathbf{x}_0$ and write down the expressions for the $\mathbf{A}$ and $\mathbf{B}$ matrices.

Hint: use (2.69) in Fossen (2020)

Problem 1.2 Consider the attitude control law:

$$\tau = -\mathbf{K}_d\omega - k_p\varepsilon \tag{2}$$

where $k_p > 0$ is a scalar control gain, $\mathbf{K}_d = k_d\,\mathbf{I}_3$ is a controller gain matrix with $k_d > 0$ and ε is the imaginary part of the unit quaternion. Let $k_p = 2$ and $k_d = 40$ and verify that the linearized closed-loop system is stable. Would you prefer real or complex poles in this particular application? Explain why/why not?

Problem 1.3 Let $k_p = 2$ and $k_d = 40$. Simulate the attitude dynamics (1) of the closed-loop system with the control law given by (2) for initial conditions $\phi(0) = -5^o$, $\theta(0) = 10^o$ and $\psi(0) = -20^o$ by modifying attitude.m. The initial angular velocities are zero. Plot the results (convert the resulting $\mathbf{q}(t)$ to Euler angles for easier visualization). Does the behavior of the system match your expectations? Explain why/why not. How would you modify the control law to follow nonzero constant reference signals? Include figures of the Euler angles, angular velocities and the control inputs in the report.

Problem 1.4 Consider the modified attitude control law:

$$\tau = -\mathbf{K}_d\omega - k_p\tilde{\varepsilon} \tag{3}$$

where $\tilde{\varepsilon}$ is the error in the imaginary part of the quaternion (between the setpoint and true state). The quaternion error is defined as:

$$\tilde{\mathbf{q}} := \begin{bmatrix} \tilde{\eta} \\ \tilde{\varepsilon} \end{bmatrix} = \bar{\mathbf{q}}_d \otimes \mathbf{q} \tag{4}$$

where $\mathbf{q}_d$ is the desired quaternion, $\mathbf{q}$ is the current state and $\bar{\mathbf{q}} = [\eta, -\varepsilon^\top]^\top$ denotes the conjugate (sometimes called the inverse) of a quaternion $\mathbf{q}$. The quaternion product is defined as:

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = \begin{bmatrix} \eta_1\eta_2 - \varepsilon_1^\top\varepsilon_2 \\ \eta_1\varepsilon_2 + \eta_2\varepsilon_1 + \mathbf{S}(\varepsilon_1)\varepsilon_2 \end{bmatrix} \tag{5}$$

where $\mathbf{S}(\varepsilon_1)$ is the skew-symmetric matrix.

Write down the matrix expression for the quaternion error $\tilde{\mathbf{q}}$ on component form. What is $\tilde{\mathbf{q}}$ after convergence, that is $\mathbf{q} = \mathbf{q}_d$?

Problem 1.5 Let $k_p = 20$ and $k_d = 400$. Simulate the attitude dynamics of the closed-loop system with the control law given by (3). The desired attitude is given by the time-varying reference signal $\mathbf{q}_d(t)$ corresponding to $\phi(t) = 0, \theta(t) = 15\cos(0.1t)$ and $\psi(t) = 10\sin(0.05t)$ (all in degrees) and the initial values are equal to the ones in Problem 1.3. Does the behavior of the system match your expectations? Explain why/why not. Include the same set of figures as in Problem 1.3 and the tracking error in the report.

Problem 1.6 Consider the modified attitude control law:

$$\tau = -\mathbf{K}_d\tilde{\omega} - k_p\tilde{\varepsilon} \tag{6}$$

where $\tilde{\omega} = \omega - \omega_d$ is the difference between the desired and current angular velocity. Let the desired attitude be given by the reference signals from Problem 1.5 and calculate the desired angular velocity using (see equation (2.31) in Fossen (2020)).

$$\omega_d = \mathbf{T}_{\Theta_d}^{-1}(\Theta_d)\dot{\Theta}_d$$

where Θ_d is the desired Euler angles. Let $k_p = 20$ and $k_d = 400$ and simulate the controller (6) in Matlab with the same parameters, initial conditions and reference signals as in Problem 1.5. Deliver the same set of figures and compare the results. Does the behavior of the system match your expectations? Explain why/why not. Can you propose a way to further improve the control law?

Problem 1.7 It can be shown that:

$$\dot{\tilde{\eta}} = -\frac{1}{2}\tilde{\varepsilon}^\top\tilde{\omega} \tag{7}$$

Assume setpoint regulation, that is $\omega_d = \mathbf{0}$, $\varepsilon_d =$ constant and η_d = constant, and the control law given by (3). Consider the Lyapunov function candidate (Fjellstad and Fossen, 1994):

$$V = \frac{1}{2}\tilde{\omega}^\top\mathbf{I}_{CG}\tilde{\omega} + 2k_p(1-\tilde{\eta}) \tag{8}$$

Explain why V is positive and radially unbounded. Show that:

$$\dot{V} = -k_d\omega^\top\omega \tag{9}$$

Use a suitable Lyapunov theorem/method to prove that the equilibrium of the closed-loop system is asymptotically stable. You need to use the nonlinear system dynamics. Is the system globally or locally asymptotically stable?

Problem 2 - Straight-line path following in the horizontal plane (35 %)

Consider a surface vessel moving in the horizontal plane (3-DOF motion). The states of interest are the horizontal velocities $(\dot{x}, \dot{y})$ and the yaw angle (ψ). Assume that no current is present.

Problem 2.1 Use the kinematics in chapter 2 in the book by Fossen (2020) to show that the north and east velocities $(\dot{x}, \dot{y})$ can be expressed as

$$\begin{aligned}\dot{x} &= u\cos(\psi) - v\sin(\psi)\\ \dot{y} &= u\sin(\psi) + v\cos(\psi)\end{aligned} \tag{10}$$

in the 3-DOF case. Explain the assumptions for these relationships to be valid.

Show that the velocities in (10) can be rewritten as

$$\begin{aligned}\dot{x} &= U\cos(\chi) = U\cos(\psi+\beta)\\ \dot{y} &= U\sin(\chi) = U\sin(\psi+\beta)\end{aligned} \tag{11}$$

where U is the horizontal speed over ground (SOG) and χ is the course over ground (COG).

Hint 1: $\sin(\alpha+\beta) = \sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta)$
Hint 2: $\cos(\alpha+\beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta)$
Hint 3: Use the connection between the body velocities, total speed and sideslip angle.

Problem 2.2 Assume that β is small and acts as a disturbance. Assume that the yaw angle ψ is small and show/explain that (11) can be rewritten as

$$\begin{aligned} \dot{x} &= U \\ \dot{y} &= U\psi \end{aligned} \tag{12}$$

The vessel is going to follow a straight line in the horizontal plane and the yaw angle is kept low so that (12) is valid. Under which assumption is y in (12) the cross-track error in the path-following problem? The cross-track error is described more closely at page 257-258 in the book by Fossen (2011) and can be interpreted as the shortest distance between the vessel and the path. Can we follow a straight-line in any direction by using y as the cross-track error?

Problem 2.3 From now on, assume that y is the cross-track error. Since we want to follow a straight line, the goal is to push y towards zero by controlling the yaw angle ψ. If this goal is reached, the vessel will follow the path. The Nomoto model (page 175 in Fossen (2020)) is a common way to express the yaw dynamics in 3-DOF motion:

$$\begin{aligned} T\dot{r} + r &= K\delta + b \\ \dot{\psi} &= r \end{aligned} \tag{13}$$

where r is the yaw rate and b is a disturbance/bias. The bias can be caused by environmental forces and/or a potential rudder offset. By using the Laplace-domain, the cross-track error y can be stated as

$$y(s) = h_1(s)\delta(s) + h_2(s)b(s) \tag{14}$$

where $b(s)$ is the bias in the Nomoto model . Find the transfer functions $h_1(s)$ and $h_2(s)$. A PID-controller is necessary if you want to control the cross-track error. Why do you need the D- and the I terms in the controller? Note that the Nomoto model is used for control design since the true nonlinear dynamics usually are to complicated.

Problem 2.4 Simulate the 3-DOF horizontal vessel motion and the path-following problem. In other words, use the Nomoto model (with a PID-controller for δ) and the equations in (11) to simulate the positions. Assume that sideslip is zero so that $\chi = \psi$. Moreover, the Nomoto gain and time constant are $T = 20$ and $k = 0.1$. The goal is to follow a path going straight to the North. The bias is a constant with value 0.001. The initial conditions should be:

$$\begin{aligned} x(0) &= 0m \\ y(0) &= 100m \\ \psi(0) &= 0^o \\ r(0) &= 0 \text{ deg/s} \end{aligned}$$

The PID-controller can be formulated as

$$\delta = -k_p y - k_d \dot{y} - k_i \int y \tag{15}$$

Find appropriate controller gains and attach simulation results which show that the vessel is able to follow the path. What happens without integral effect in the controller? Show simulation results in this case as well. Figures of the north-east positions, yaw angle, yaw rate and the control input should be included in the report. Assume that a speed-controller is able to keep the total speed U at $5m/s$. Comment on the results.

Hint 1: Use the simplified linear relationship from (12) to implement the derivative term in the PID-controller.
Hint 2: It is clever to add saturation on the rudder input and keep it from going above e.g. 20^o as this is unrealistic and can cause problems.
Hint 3: The controller gains need to be very small. This is because the error in the controller is the cross-track error y, which is measured in meters, and you want to map that error to an angle in radians.

1 Appendix

1.1 attitude.m

```
% M-script for numerical integration of the attitude dynamics of a rigid
% body represented by unit quaternions. The MSS m-files must be on your
% Matlab path in order to run the script.
%
% System:                           .
%                                   q = T(q)w
%                                   .
%                                 I w - S(Iw)w = tau
% Control law:
%                                 tau = constant
%
% Definitions:
%                                 I = inertia matrix (3x3)
%                                 S(w) = skew-symmetric matrix (3x3)
%                                 T(q) = transformation matrix (4x3)
%                                 tau = control input (3x1)
%                                 w = angular velocity vector (3x1)
%                                 q = unit quaternion vector (4x1)
%
% Author:                        2018-08-15 Thor I. Fossen and Hakon H. Helgesen

%% USER INPUTS
h = 0.1;                          % sample time (s)
N  = 400;                         % number of samples. Should be adjusted

% model parameters
m = 180;
r = 2;
I = m*r^2*eye(3)                  % inertia matrix
I_inv = inv(I);

% constants
deg2rad = pi/180;
rad2deg = 180/pi;

phi = -10*deg2rad;                 % initial Euler angles
theta = 10*deg2rad;
psi = 5*deg2rad;

q = euler2q(phi,theta,psi);        % transform initial Euler angles to q

w = [0 0 0]';                      % initial angular rates

table = zeros(N+1,14);             % memory allocation

%% FOR-END LOOP
for i = 1:N+1,
t = (i-1)*h;                       % time
tau = [0.5 1 -1]';                    % control law

[phi,theta,psi] = q2euler(q); % transform q to Euler angles
[J,J1,J2] = quatern(q);            % kinematic transformation matrices

q_dot = J2*w;                              % quaternion kinematics
w_dot = I_inv*(Smtrx(I*w)*w + tau);  % rigid-body kinetics

table(i,:) = [t q' phi theta psi w' tau'];  % store data in table

q = q + h*q_dot;                      % Euler integration
w = w + h*w_dot;

q  = q/norm(q);                    % unit quaternion normalization
end
```

```
%% PLOT FIGURES
t        = table(:,1);
q        = table(:,2:5);
phi      = rad2deg*table(:,6);
theta    = rad2deg*table(:,7);
psi      = rad2deg*table(:,8);
w        = rad2deg*table(:,9:11);
tau      = table(:,12:14);


figure (1); clf;
hold on;
plot(t, phi, 'b');
plot(t, theta, 'r');
plot(t, psi, 'g');
hold off;
grid on;
legend('\phi', '\theta', '\psi');
title('Euler angles');
xlabel('time [s]');
ylabel('angle [deg]');

figure (2); clf;
hold on;
plot(t, w(:,1), 'b');
plot(t, w(:,2), 'r');
plot(t, w(:,3), 'g');
hold off;
grid on;
legend('x', 'y', 'z');
title('Angular velocities');
xlabel('time [s]');
ylabel('angular rate [deg/s]');

figure (3); clf;
hold on;
plot(t, tau(:,1), 'b');
plot(t, tau(:,2), 'r');
plot(t, tau(:,3), 'g');
hold off;
grid on;
legend('x', 'y', 'z');
title('Control input');
xlabel('time [s]');
ylabel('input [Nm]');
```

References

R. W. Beard and T. W. McLain. Small Unmanned Aircraft. Theory and Practice. Princeton University Press, 2012.

T. Fossen, Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley & Sons, 2011.

O.-E. Fjellstad and T. I. Fossen, “Quaternion feedback regulation of underwater vehicles,” vol. 2, 1994, pp. 857–862.

N. A. Chaturvedi, A. K. Sanyal, and N. H. McClamroch, “Rigid-body attitude control,” IEEE Control Systems, vol. 31, no. 3, pp. 30–51, June 2011.