AE353: Design Problem 03
(Control of Unpowered Glider)

T. Bretl

This paper describes the third design problem that you will complete in AE353 (Spring 2020). It asks you to design, implement, and test a controller for un unpowered glider, and to submit a report that describes both your method of approach and your results.

I. Nomenclature

x, y	=	horizontal and vertical position in meters
$\dot{x}, \dot{y}$	=	horizontal and vertical velocity in meters per second
θ	=	pitch angle in radians
$\dot{\theta}$	=	pitch angular rate in radians per second
ϕ	=	elevator angle in radians
$\dot{\phi}$	=	elevator angular rate in radians per second
v	=	airspeed in meters per second
α	=	angle of attack in radians
c_L, c_D	=	coefficients of lift and drag
$\dot{\phi}_{\max}$	=	maximum elevator angular rate in radians per second

II. Goal

The code provided in `DesignProblem03` simulates an unpowered glider that is similar to one used for experiments on fixed-wing perching [1, 2]. This glider has one control surface, an elevator. An actuator allows you to specify the angular rate of this elevator. Sensors allow you to measure both the pitch angle and the airspeed of the glider. The goal is to make the glider fly as long a distance as possible, as reliably as possible, if released at a height of about two meters with a forward speed of about six meters per second and with a pitch angle of your choice.

III. Model

The motion of the glider is governed by ordinary differential equations with the form

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{bmatrix} = f(\theta, \phi, \dot{x}, \dot{y}, \dot{\theta}, \dot{\phi}) \tag{1}$$

where θ is the pitch angle, ϕ is the elevator angle, $\dot{x}$ is the forward speed, $\dot{y}$ is the vertical speed, $\dot{\theta}$ is the pitch angular rate, and $\dot{\phi} \in [-\dot{\phi}_{\max}, \dot{\phi}_{\max}]$ is the elevator angular rate (which an actuator allows you to specify). You might be interested to know that these equations were derived by applying a flat-plate model of lift c_L and drag c_D as a function of angle of attack α, for both the wing and elevator:

$$c_L = 2 \sin\alpha \cos\alpha \qquad\qquad c_D = 2 \sin^2\alpha$$

Experimental results show that this flat-plate model is a good approximation for the glider and that it remains accurate at high angles of attack and even post-stall [2]. The function f in (1) depends on a number of parameters (e.g., mass, moment of inertia, surface area of the wing). You can save both a symbolic description and a numeric description of this function to a file for later analysis if you call the simulator with the optional parameter `'eomfile'`, for example as follows:

```
% Run the simulator to save the equations of motion
DesignProblem03('Controller', 'eomfile', 'eom.mat', 'display', 'false');
% Load the equations of motion
load('eom.mat');
% Parse the equations of motion
f = symEOM.f;
% Define symbolic variables that appear in the equations of motion
syms theta phi xdot ydot thetadot phidot
% Proceed to linearize or do whatever else you like...
%       (see: help sym/jacobian, help sym/diff, help sym/subs, etc.)
```

Assuming zero wind, the airspeed sensor measures the component of velocity that is aligned with the wing:

$$\text{airspeed} = \dot{x}\cos\theta + \dot{y}\sin\theta$$

This sensor cannot measure "negative airspeed," so if airspeed < 0 then the sensor will produce a zero measurement.

IV. Tasks

A. Analysis

The focus of your analysis this time will be on data collection. The initial conditions in the simulation are random. So, for a given control design, the flight distance will vary (perhaps significantly). It is not enough to look at the results of one flight—you will have to look at the results of many flights. At minimum, for each design and for each launch angle that you consider, you should do the following:

- Collect data from at least 1000 flights.
- Compute the minimum, maximum, median, mean, and standard deviation of the flight distance (e.g., `min`, `max`, `median`, `mean`, and `std` in MATLAB).
- Plot a histogram of the flight distance (e.g., `hist` in MATLAB).

Remember that `DesignProblem03` provides options to save data and to turn the graphics off, both of which are useful for data collection. Here is an example of how to use these options to collect and analyze data from many flights.

```
% Number of flights
nFlights = 1e3;
% Loop over each flight
for i=1:nFlights
    % Run simulation without graphics and save data
    DesignProblem03('Controller','datafile','data.mat','display',false);
    % Load data
    load('data.mat');
    % Get t and x
    t = processdata.t;
    x = processdata.x;
    % Do analysis...
    %     (your code here)
end
```

B. Presentation

The focus of your presentation this time will be on equations. For example, you should be sure to do the following:

- Number any equation to which you will refer, and do the reference with `\eqref`.
- Use standard notation for operations like multiplication (e.g., Ax for "A times x" and not $A \times x$ or $A * x$).
- Place square brackets around matrices. For example,

```
\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{bmatrix}
```

produces the standard notation

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{bmatrix},$$

while

```
\begin{matrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{matrix}
```

produces the non-standard and confusing notation

$$\begin{matrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{matrix}.$$

- Align equations properly. For example,

```
\begin{align*}
c_{L} &= 2\sin\alpha \cos\alpha \\
c_{D} &= 2\sin^{2}\alpha
\end{align*}
```

produces the nice-looking result

$$\begin{aligned} c_L &= 2 \sin\alpha \cos\alpha \\ c_D &= 2 \sin^2 \alpha \end{aligned}$$

while

```
\begin{equation*}
c_{L} = 2\sin\alpha \cos\alpha
\end{equation*}
\begin{equation*}
c_{D} = 2\sin^{2}\alpha
\end{equation*}
```

produces the messy-looking result

$$c_L = 2 \sin\alpha \cos\alpha$$
$$c_D = 2 \sin^2 \alpha.$$

- Use the correct font for functions like cosine (`$\cos\alpha$` produces a nice-looking $\cos\alpha$, while `$cos\alpha$` produces a messy-looking $cos\alpha$).

Please be responsive to feedback on your draft reports—this feedback will focus on helping you improve your presentation of equations.

C. Report

Your report must be a PDF document that was generated using LaTeX and that conforms to the guidelines for "Preparation of Papers for AIAA Technical Conferences" (`https://go.aerospace.illinois.edu/aiaa-latex-template`). The author must be listed as "Anonymous" (with no affiliations). The report must be exactly four pages. It must have the following sections:

- *Abstract*. Summarize your key results in one short paragraph.
- *Nomenclature*. List all symbols used in your report, with units.
- *Goal*. At minimum, this section will describe the system you have been asked to control, define one requirement and one verification (which must depend on rigorous data collection and analysis as described in Section IV.A), and prepare the reader to understand the rest of your report.
- *Model.* At minimum, this section will linearize (1) about some choice of equilibrium point, expressing the result in state-space form.
- *Control Design.* At minimum, this section will present the design of a controller and an observer (remembering to check, first, that the linearized model is both controllable and observable).
- *Results.* At minimum, this section will describe what you did to implement and test your controller and observer in simulation. It will show that you have followed the instructions that you wrote to verify that your requirement has (or has not) been satisfied and will, in particular, include at least one figure of aggregate results (e.g., a histogram). It will also include at least one figure that provides evidence the observer is working (e.g., a plot of the error in the state estimate as a function of time).

You are encouraged to go beyond these minimum requirements. For example, it is likely that you will want to iterate on your design and on your choice of initial launch angle, in order to maximize flight distance. In this case, it would be helpful to describe the process that you used to improve your design.

V. Deliverables

A. Submit a first draft by 11:59PM on Wednesday, April 1, 2020

This draft must be a PDF document that follows the guidelines provided in Section IV.C and that includes, at minimum, a **complete draft** of your "Goal" and "Model" sections. It must be submitted here:

`https://forms.illinois.edu/sec/3018672`

You may submit as many times as you want—only the last submission will be recorded.

B. Submit a second draft by 11:59PM on Monday, April 6, 2020

This draft must be a PDF document that follows the guidelines provided in Section IV.C and that includes, at minimum, a **complete draft** of your "Control Design" and "Results" sections. It must be submitted here:

`https://forms.illinois.edu/sec/7635612`

You may submit as many times as you want—only the last submission will be recorded.

C. Submit a final report by 11:59PM on Friday, April 10, 2020

This report must be a PDF document that follows the guidelines provided in Section IV.C. You must submit three things in addition to the report:

- The LaTeX source files that you used to produce the document (in a ZIP folder with all figures, references, etc., in addition to the `.tex` file). Please do *not* include MATLAB code with your LaTeX source files.
- The MATLAB code that you used to produce all of the figures, tables, and other results that are included in the document (in a ZIP folder). Your code must have a MATLAB script called `GenerateResults.m`. It should be possible for any of your peers to reproduce *everything* in your report by executing this script. Please do *not* include LaTeX source files with your MATLAB code.
- The implementation of your "best" controller as a single file `Controller.m`. If one of your peers runs the simulator with this controller, they should see behavior that is consistent with claims made in the report.

Your report must be submitted here:

`https://forms.illinois.edu/sec/1071483`

You may submit as many times as you want—only the last submission will be recorded.

VI. Evaluation

Your work will be evaluated based on meeting intermediate deadlines (20%) and on staff reviews of your code (20%) and of your report (60%).

Reviews of your technical approach will place special emphasis on *data collection and analysis*. Reviews of your report will place special emphasis on your *presentation of equations*.

Staff review will be "double-blind." You won't know who reviewed your report, and your reviewers won't know whose report they reviewed. To enable a double-blind review process, it is **very important** that your final report be completely anonymous. To repeat, **DO NOT INCLUDE YOUR NAME** or anything else that would identify you in any of the materials you submit—not in your PDF, in your two ZIP folders, or in your MATLAB script.

VII. On-Time Submission

Draft reports and final reports must be submitted on time or they will receive zero credit. I am so serious about this that I will even give you extra credit if you submit your final report early:

> **If the last submission of your final report occurs by 11:59PM on Wednesday, April 8, 2020—so, 48 hours before the final deadline—then you will receive 5% extra credit.**

Please do not put yourself in the danger zone (Figure 1). An easy way to make sure you do not receive zero credit is to submit each of your first two drafts (Sections V.A-V.B), as soon as you write them, also as your final report.

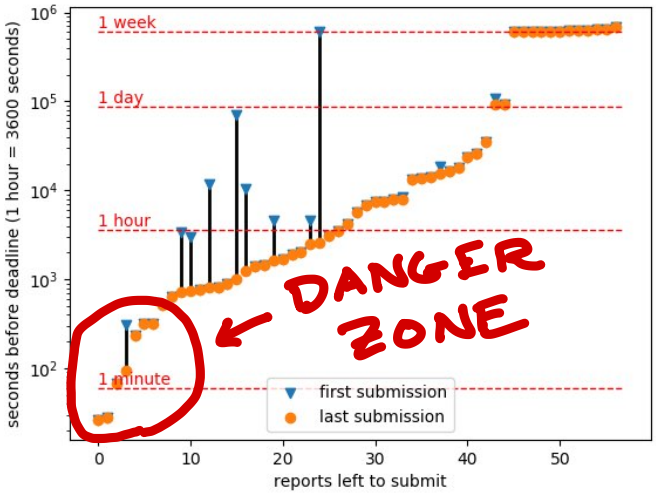

Fig. 1 Submission times for the second design report this semester. If you choose to submit your final report for the first time at the last minute, you are asking for trouble.

References

[1] Roberts, J. W., Cory, R., and Tedrake, R., "On the controllability of fixed-wing perching," *American Control Conference*, 2009, pp. 2018–2023. https://doi.org/10.1109/ACC.2009.5160526, URL http://dx.doi.org/10.1109/ACC.2009.5160526.

[2] Moore, J., Cory, R., and Tedrake, R., "Robust post-stall perching with a simple fixed-wing glider using LQR-Trees," *Bioinspiration & Biomimetics*, Vol. 9, No. 2, 2014, p. 025013. URL http://stacks.iop.org/1748-3190/9/i=2/a=025013.