CS 439 Task 2: Airplane Maneuvers

Description

This task addresses the simulation and visualization of the flight path of a basic airplane. Lecture 15 discusses how the provided model `Airplane` functions. Your job is to implement simulations for each test in the provided simulation `Simulation` and visualize the results. Limited analysis plays a role in two aspects: correcting for path errors in some simulations, and explaining the basis for these corrections in the report.

Graduate students have two more simulations to consider.

Specifications

Download the appropriate version of Gnuplot for your platform at https://sourceforge.net/projects/gnuplot/

The airplane model generates text output files in comma-delimited Excel (`.csv`) and Gnuplot (`.gnu`) formats. For each test, provide a different filename. The filename does not matter for the task because you will not submit these files, but choose something meaningful for yourself to keep from getting confused. The Gnuplot file contains both the configuration and the data to plot. From the Gnuplot command line, type `load '`*`filename`*`.gnu'`

You may change the view however you want to make it more readable. The `set zrange [0:1000]` command may be helpful for some of the tests with a vertical component. (This example says minimum 0 feet, maximum 1000.) Otherwise, Gnuplot will scale automatically and make small altitude fluctuations look huge or large ones appear small. Autoscaling on the horizontal plane is managed correctly because the axes are locked to the same scale.

The Excel file is not explicitly used, but you may find it helpful in understanding and explaining some simulations. It contains the same data and more. The header row is defined as follows:

`time`	simulation time in seconds
`x`	airplane *x* position in feet
`y`	airplane *y* position in feet
`altitude`	airplane altitude position (*z*) in feet
`yaw`	airplane yaw in compass degrees
`pitch`	airplane pitch in degrees (positive up)
`roll`	airplane roll in degrees (positive right wing down)
`speed`	airplane speed in miles per hour
`wind_dir`	wind direction *from* compass degrees
`wind_spd`	wind speed in miles per hour
`dx`	change in airplane *x* by own movement only
`dy`	change in airplane *y* by own movement only
`wind_dx`	change in airplane *x* by wind only
`wind_dy`	change in airplane *y* by wind only
`vmg_dx`	airplane velocity made good *x* as difference between `dx` and `wind_dx`
`vmg_dy`	airplane velocity made good *y* as difference between `dy` and `wind_dy`

You must submit this output for each test in support of your graphs. Make one Excel workbook with separate sheets for each.

Unless otherwise specified, start each test from position (0,0) at altitude 1,000 and speed 100 with a time step of 0.1 seconds. The tests have some freedom, so it is ok to be off by a reasonable amount; i.e., “that looks about right.” As always, ask if you are unsure.

In the report, show the combined results as at least two graphs: one two dimensional (usually top view) and one three dimensional. Hint: use commands `set view 0,0` and `set view 70,45`, respectively.

Part I

Tests 1 through 11 have no wind, which by default is equivalent to `airplane.setWind(0,0)`

Test 1: Straight and level

Fly straight north for 30 seconds while holding altitude.

Test 2: Straight climbing

Fly straight north for 30 seconds while climbing at 10 degrees of pitch.

Test 3: Turning clockwise

Fly a complete clockwise circle at bank 30 and pitch 0.

Test 4: Turning clockwise climbing

Fly a complete clockwise circle at bank angle 30 while climbing at 10 degrees of pitch.

Test 5: Turning clockwise level

Fly a complete clockwise circle at bank angle 30 while holding altitude with pitch. Adjust the vertical scale in Gnuplot (see below) so the deviation is apparent but not misleading. Your solution will not be perfect. Gnuplot by default will scale any error to fill the vertical graph limits, which makes minor (acceptable) errors look extreme against the scale of the horizontal plane.

Test 6: Turning with different bank angles

Fly a complete clockwise circle at bank 30, 45, and 60 and pitch 0.

Do the flights with three airplanes. Use a different filename for each. Manually combine the three Gnuplot datasets into one output file with a blank line between each. Keep the configuration the same at the top.

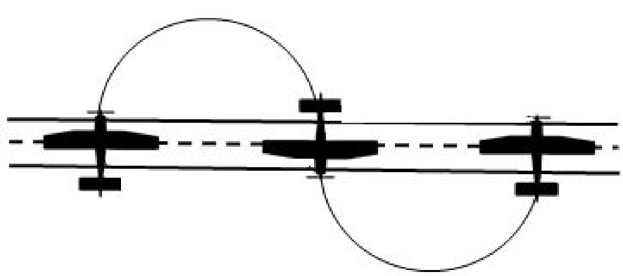
Test 7: Turning with different speeds

Fly a complete clockwise circle at bank 30 with speeds 50, 100, and 150 and pitch 0.

Do all three experiments simultaneously with three airplanes as in Test 6.

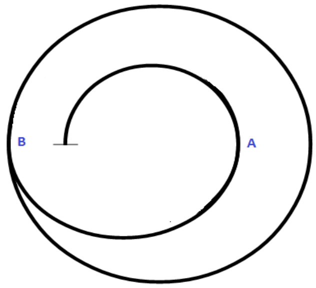
Test 8: S-turn

Fly an S-turn to the east at bank 30 right and left while holding altitude. The flight path should resemble this:



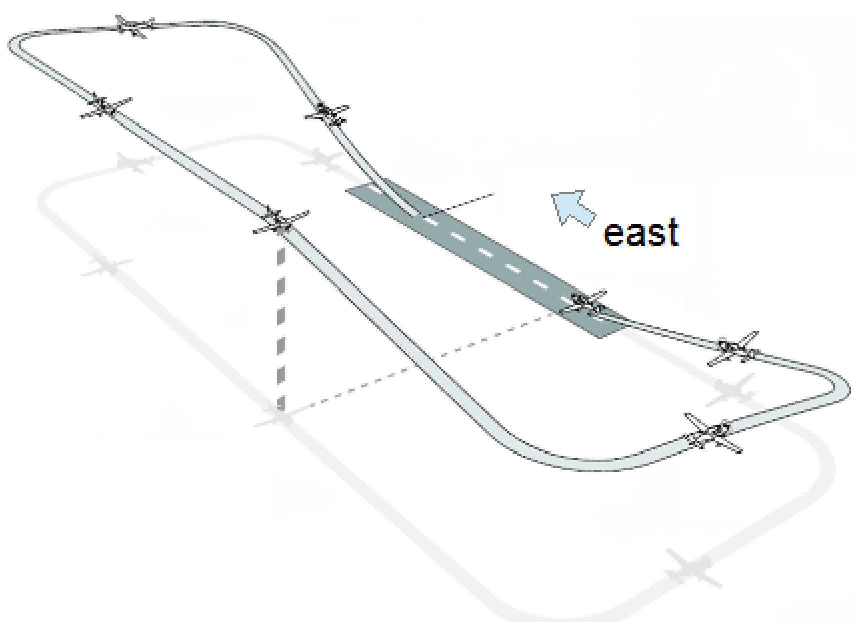
Test 9: Turning with variable speed

Fly two clockwise turns at bank 30 starting at speed 100. Halfway through the first turn (A), increase speed to 150. At one complete turn (B), increase to 200 and complete the second turn. The flight path should resemble something like this:



Test 10: Rectangular traffic pattern

Assume a runway 5,000 feet long facing east at 0 feet altitude. Start at the left (0,0) at 60 mph down the runway. At the three-quarter point, increase speed to 80 and climb to 500 feet. Turn left 90 degrees at 30 degrees of bank and continue climbing to 1,000 feet. Turn left (parallel to the runway in the opposite direction of takeoff), fly level until abeam the start of the runway (x=0), descend to 750 feet, turn left, descend to 500, turn left one last time to align with the runway, reduce speed to 60, land, and stop halfway down the runway. The flight path should resemble this.



Test 11: Loop

Attempt to loop the airplane. It is not possible, but try anyway to see what happens. Every three simulation seconds, increase the pitch by 10 degrees until it reaches 360. In addition to the graphs, use the Excel output to try to explain why this does not work. What should happen? In particular, look at the yaw, pitch, and roll data and compare against what an inside loop is supposed to look like.

Part II

Tests 12 through 18 have wind *from* the northeast at 10 miles per hour. Add `airplane.setWind(45,10)` before your simulation loop.

Test 12: Turning clockwise level with wind

Repeat Test 5.

Test 13: Turning counterclockwise level with wind

Repeat Test 5 counterclockwise.

Test 14: S-turn with wind

Repeat Test 8.

Test 15: Rectangular traffic pattern with wind

Repeat Test 10.

Test 16: Straight and level with wind

Repeat Test 1.

Test 17: Straight and level with wind, compensated

Repeat Test 1. Try to stay reasonably on course. Report the cross-track error as the average error in position between the actual and expected x positions at each time step.

Test 18: Turning clockwise level with wind, compensated

Repeat Test 5. Try to stay reasonably on course. Report the cross-track error as the average error in position between the actual and expected x and y positions at each time step. Use the Pythagorean Theory.

Part III

Only graduate students do Tests 19 and 20. Use the same wind from Part II.

Test 19: S-turn with wind, compensated

Repeat Test 8. Try to stay reasonably on course.

Test 20: Rectangular traffic pattern with wind, compensated

Repeat Test 10. Try to stay reasonably on course.

Deliverables

Zip all the files:

- Your Java code for `Simulation` only. No other code should need changing.
- A nicely formatted PDF document that captures the Gnuplot graphs for the tests.
- Your spreadsheet in Excel or OpenOffice Calc format. Make sure it is clear which tabs refers to which tests.

Do not submit the `.gnu` files. (They can be derived from the Excel data, if necessary.)