Yann PIERRE

Maëllys MASCART

Maxent BRUNEL

ODE : LAND A PLANE

This document will expose our models, plans and solutions for the glider project.

Introduction :

ODEs are used in a wide range of models to study and predict different systems. Some famous models include :

- The Lotka-Voltera predator-prey model,
- The Verhulst equation,
- The SIR model
- Glider model

```
function Glider_EE(T,h,teta,vini,mud,mul)
      % Glider model
      % Explicit-Euler
      % % % % % % % % % % % % % % %
      N=round(T/h); % stepsize
      t=linspace(0,T,N+1); % grid
      y=zeros(4,N+1); % numerical approximation
      % EE:
      y(:,1)=[0 0 teta vini]';
      for j=1:N
      y(:,j+1)=y(:,j)+h*Glyder(mud,mul,y(:,j));
   end
   % visualization:
      plot(y(1,:),y(2,:));
      xlabel('x')
   ylabel('y')
end
      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function f = Glyder(mud,mul,y)
      % SIR model
      f = zeros(4,1);
      f(1) = y(4)*cos(y(3));
      f(2) = y(4)*sin(y(3));
      f(3) = -9.8*cos(y(3))/y(4)+mul*y(4);
   f(4) = -9.8*sin(y(3))-mud*(y(4))^2;
end
```

Objectives:

1) The plane must reach/touch the ground horizontally, it must not crash.
2) The plane must land at a specific point.
   More especially, the goal is the plane must arrive at a specific place AND touch the floor horizontally.

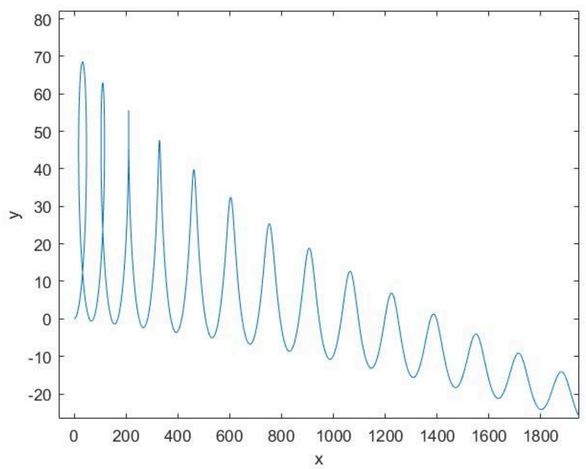
If we have time:

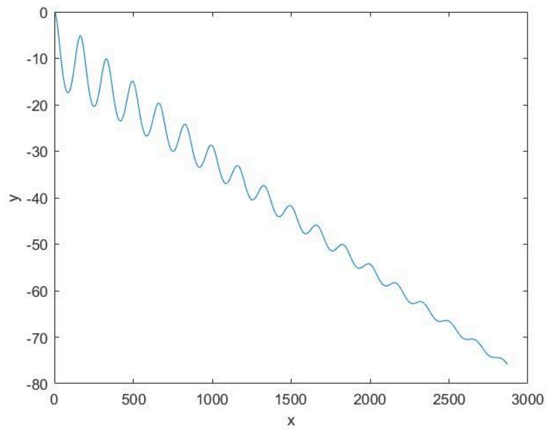
3) The plane must go as far as possible.

Methods / Solutions:

1) Landing :

   With the above code, we succeed to plot the trajectory of a glider, and we can modify the starting speed, angle and position, which will probably need to be randomized at every episode.





Now, we want to simulate a landing. The first thing would be to set up a threshold at y=0 which indicates a “crash”.

Second step would be to set the “action” for the glider to adjust its trajectory, which means adjust the angle teta. I’m seeing from the examples we were given, that the “action” is changing the state of the wing (which translates in real life to moving the wing’s flaps, which affects the lift and drag coefficients).

But landing also means reaching this y=0 slowly and steady. So a first idea would be to put a second line, close to the ground (y=5,10 ? we’ll test), and reward the agent for staying in this interval. And maybe a bigger reward if he reaches a speed of 0 in it.

We’ll need to check if we need to put a threshold on the angle teta. It could be funny to see it diving straight down, and adjust the trajectory at the last second.

2) Specific landing

We’ll need to randomize a “landing spot”, and adjust the threshold at every episode (If you’re above the landing interval and beyond the landing point you failed etc..)