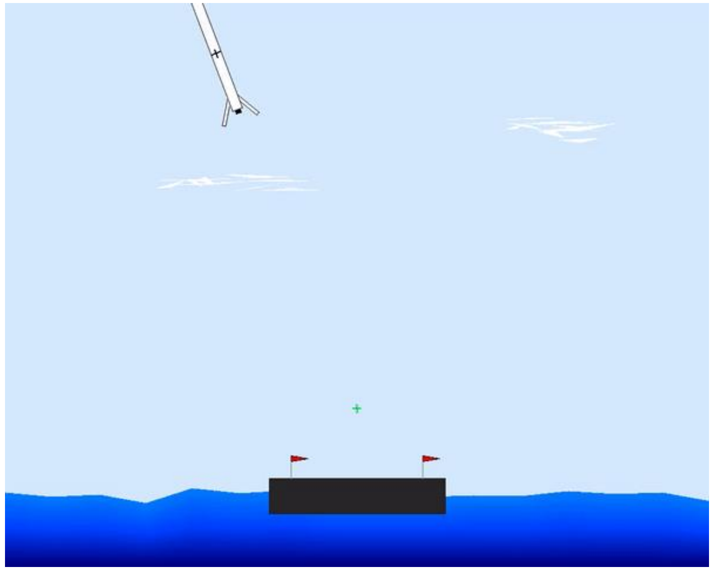
Current state

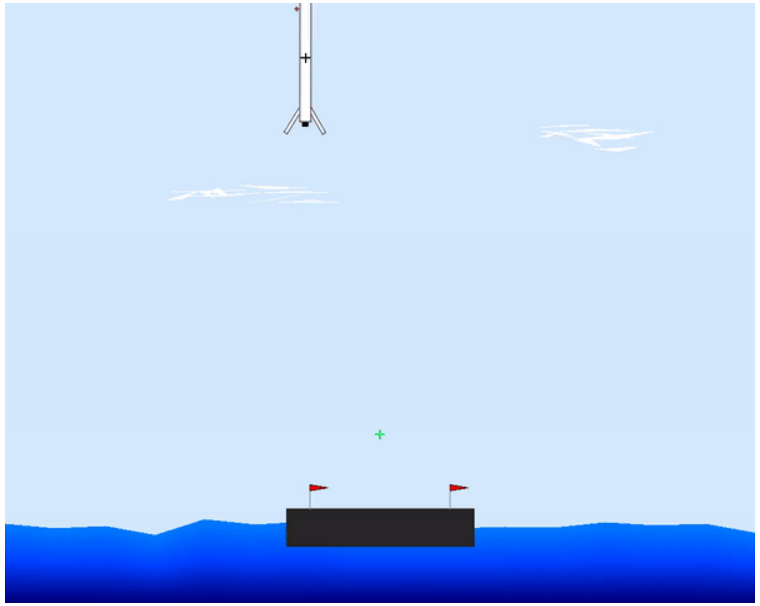
- PID controller with the following characteristics:
  - For F_E: $k_p = 10, k_i = 0, k_d = 10$
  - For F_S : $k_p = 0.085, k_i = 001, k_d = 10.55$
  - For Φ: $k_p = 5, k_i = 0, k_d = 6$



- The controller performs well under „easy“ initial states. The landing is smooth and pretty accurate.

Failure mode

- The PID controller is not always efficient for the landing of the rocket. As the PID is not explicitly designed to handle the constraints of the system, and if this system is under too high nonlinearities, the rocket can become unstable and diverges from the platform, as shown in the video.



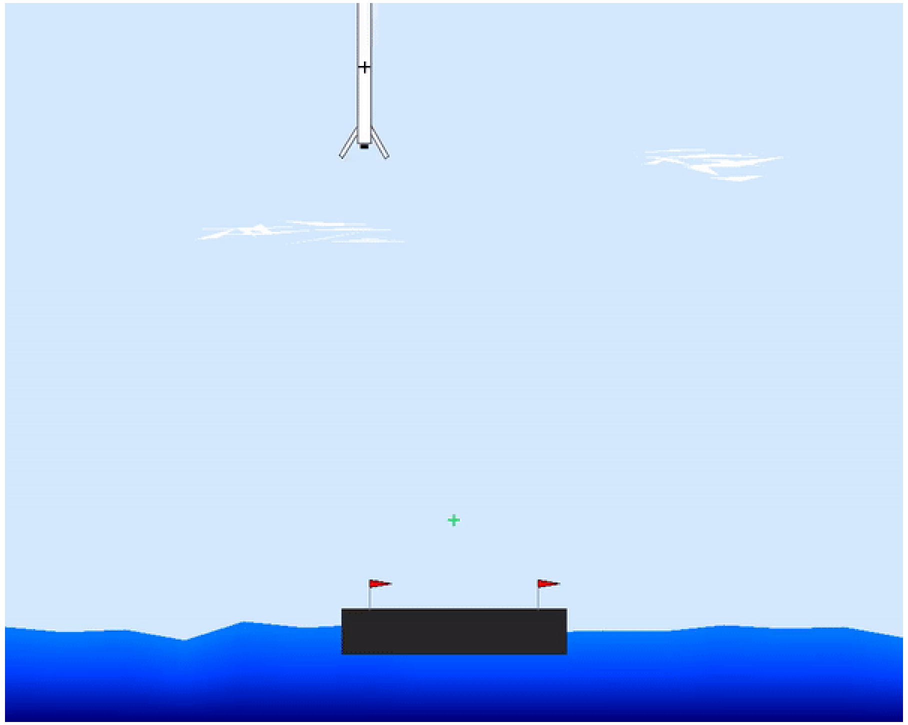
- In this scenario, the rocket comes vertically but not precisely from the center of the platform, as it can happen in real life. The PID controller will try to land the rocket at the exact goal position. However, this is too complicated for the controller to handle. Therefore, the control inputs are beyond the constraints, and the rocket do not manage to land on the platform.

My recommendation

- In order to resolve the problems of the PID controller, I would use a tracking MPC. Indeed, it is very efficient for landing a rocket because of the following reasons:
  - **Constraints handling**: the incorporation of the constraints during the design of the MPC provides a safe operation of the rocket within its boundaries
  - **Trajectory optimization**: the fact that the MPC optimizes the trajectory over a finite time horizon at each time step results in smoother and more accurate tracking
  - **Nonlinear System Handling**: even though the tracking MPC uses a linear model of the rocket, it is still more robust to nonlinearities than the PID controller

Demonstration

- With the video on the right, we can see that the rocket lands well on the platform, but not precisely at the exact goal position. This is still acceptable, and it is because of the design of the state and input costs. The controller favors to land the rocket by avoiding a too wide angle rather than to land at the middle of the platform, as the PID controller would do. This sacrifice of accuracy allows a safety landing rather than a fail.



A plan for deployment

- In order to deploy the tracking MPC for the rocket, a procedure must be followed. The most important steps are the following:
  - **Data**: For the identification and validation of a model, accurate and representative data are crucial. Data from flight tests, wind tunnel tests, and realistic simulations fall under this category. The information must be accurate and comprehensive, and it must show the dynamics of the rocket during landing
  - **System modelling**: the model of the rocket dynamics must be as accurate and reliable as possible. Otherwise, the controller's performance would be limited and could prevent the rocket from landing
  - **Real-time implementation**: the tracking MPC involves a real-time optimization calculations within the control loop. This requires sufficient computational resources such as high computational speed, high processing power and big memory
  - **Measurements**: the state is essential for feedback the controller. The onboards sensors must be really accurate and should return datas about the position, the velocity, the angle and the angular velocity