COCO Project

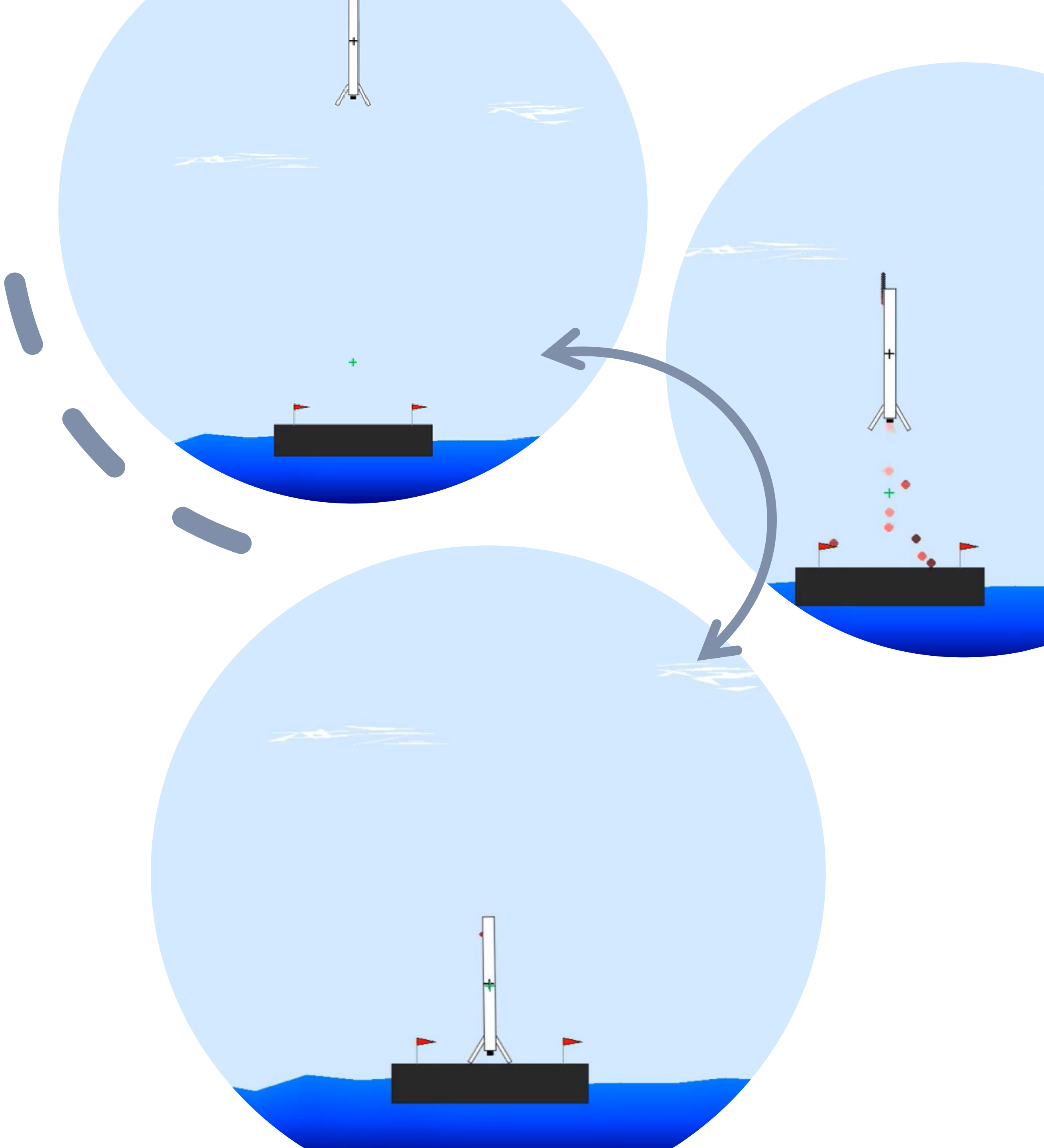
Vertical Takeoff, Vertical Landing Rocket

Philippe Brigger – 18-927-822

ETH Zürich, July 2023

Current State

- Simulation settings
  - No wind disturbances
  - Landing barge does not move
  - Initial position above landing barge
- PID controller
  - Stabilizes and lands rocket
  - No state or input constraints taken into account
- Computational effort per calculation step
  - $\emptyset = 0.023\ s$
  - $\sigma = 0.006\ s$

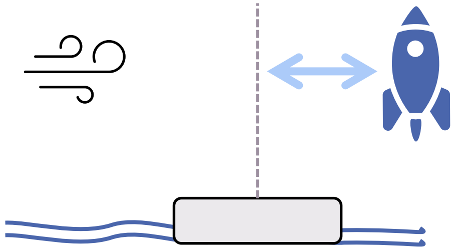


Failure Modes

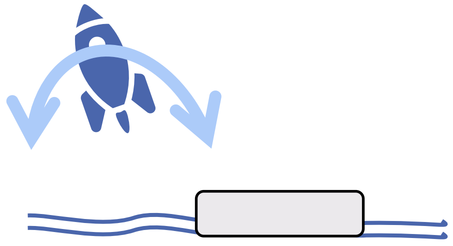
- PID controller failure modes:
  - Initial position not above landing barge (I, II, III)
  - Wind disturbances (I)
  - Initially tilted and non-zero initial velocities (II)
  - Moving landing barge (III)
- Failure modes are likely
  - Rocket usually not straight above landing barge
  - Wind disturbances on open-sea plausible, rocket thrust creates air turbulences when rocket is near ground
  - Water flow can make landing barge move

Conclusion: A more sophisticated controller is necessary!

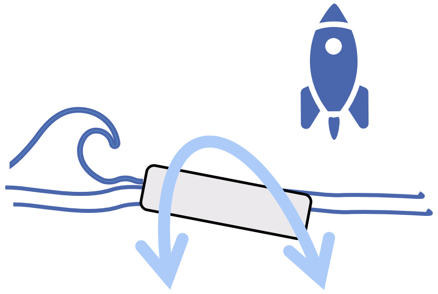
Failure Mode (I)



Failure Mode (II)



Failure Mode (III)



My Recommendation

- Use knowledge of:
  - Linearized rocket model (LTI system)
  - State constraints
  - Input constraints
- Model Predictive Control (MPC)
  - Minimize cost, respect constraints
  - Solve *N*-step optimization problem (Receding horizon control)
- Advantages of MPC over PID
  - Satisfies state and input constraints
  - Adapts to disturbances and setpoint changes
  - Can optimize fuel consumption via cost function

MPC Design

$$\min_{U,\,X}\left\{(x_N - x_{des})^T Q_N (x_N - x_{des}) + \sum_{i=0}^{N-1} (x_i - x_{des})^T Q (x_i - x_{des}) + u_i^T R u_i\right\}$$

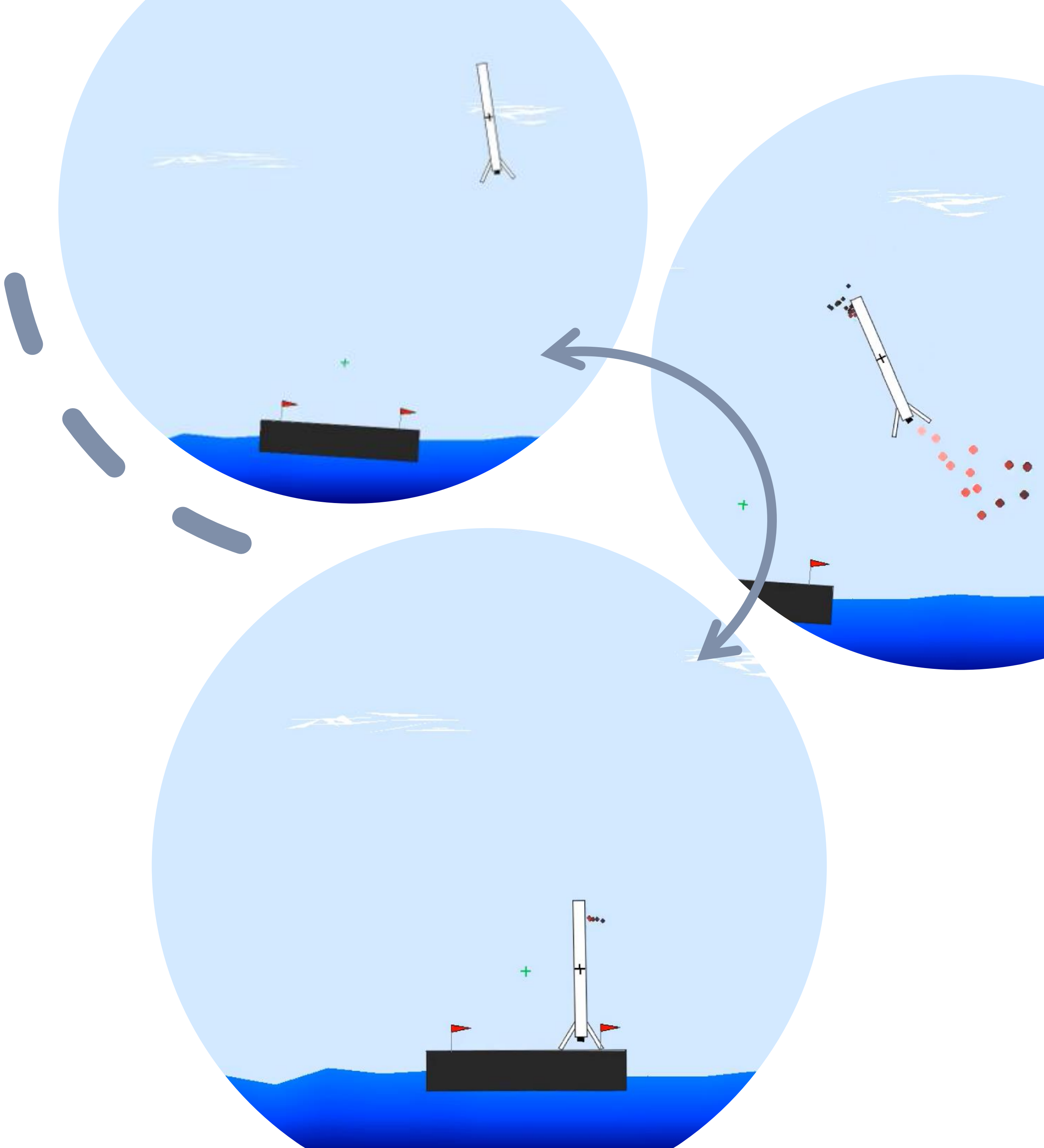
$$subj.to \quad x_{i+1} = Ax_i + Bu_i$$

$$x_0 = obs, \qquad x_i \in \mathcal{X}, \qquad u_i \in \mathcal{U}$$

N:	Prediction horizon	Q_N:	Final state cost matrix
x_i:	State (of X) in i^{th} step	Q:	State stage cost matrix
u_i:	Input (of U) in i^{th} step	R:	Input stage cost matrix
x_{des}:	Desired landing position	obs:	Simulation state
A:	Linearized system matrix	$\mathcal{X}$:	State constraints
B:	Linearized input matrix	$\mathcal{U}$:	Input constraints

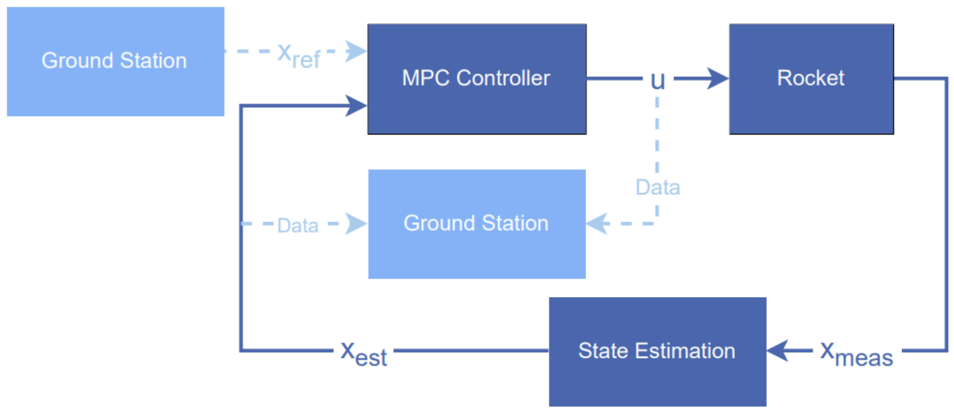
Demonstration

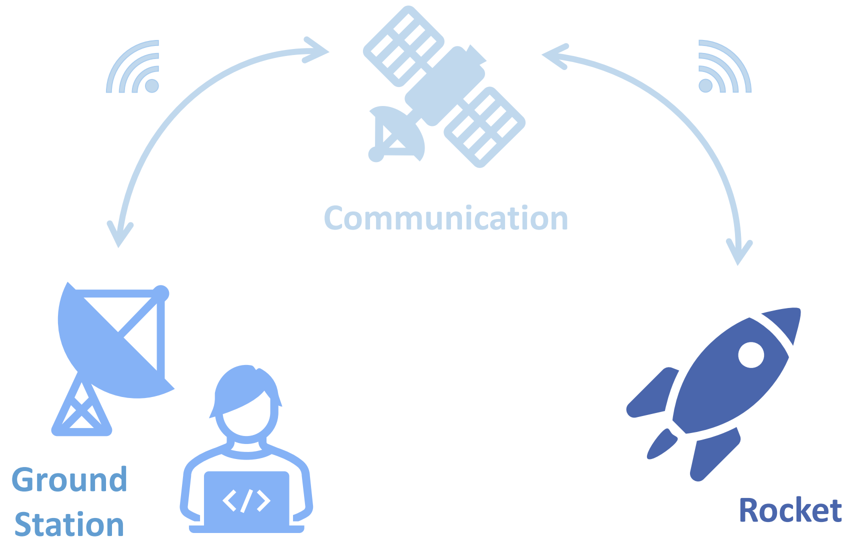
- Simulation settings
  - Wind disturbances (from failure mode I)
  - Tilt, non-zero velocities (from failure mode II)
  - Moving landing barge (from failure mode III)
  - Combination of all three failure modes
- MPC controller
  - Stabilizes and lands rocket (adapts to disturbances)
  - Satisfies state and input constraints
  - Uses less fuel
- Computational effort per calculation step
  - $\emptyset = 0.240\ s$
  - $\sigma = 0.133\ s$



Deployment Plan - Requirements

- **On-board** state estimation
  - IMU, GPS, barometer, magnetometer and camera sensors
  - Sensor fusion with Extended Kalman Filter approach
- **Communication** between **rocket** and **ground station**
  - Transmitter, receiver and antenna for radio frequency telemetry system
  - Enables data collection for offline model improvement
  - Mission control from ground station (E.g. landing position)
- **On-board** real-time computing
  - Many sensors for redundancy (robustness, safety)
  - Explicit MPC for real-time implementation (look-up tables)
  - High-performance on-board computer (run control loop at a high frequency)





Bonus – Data-Enabled Predictive Control (DeePC)

- No model information:
  - Apply random inputs to system u_{data}
  - Collect corresponding states x_{data}
  - Setup Hankel matrices H_L with collected data
- Data-Enabled Predictive Control (DeePC):
  - Under assumption of controllable, LTI system => Any trajectory is a linear combination of the collected data
  - Satisfies state and input constraints
  - Solve *N*-step optimization problem (Receding horizon control)
- Computational effort per calculation step:
  - $\emptyset = 17.904\ s,\ \sigma = 5.367\ s$

DeePC Design

$$\min_{U,X,g} \left\{ \sum_{i=0}^{N-1} (x_i - x_{des})^T Q (x_i - x_{des}) + u_i^T R u_i \right\}$$

$$subj.to \quad \begin{bmatrix} H_L(u_{data}) \\ H_L(x_{data}) \end{bmatrix} g = \begin{bmatrix} u_{past} \\ U \\ x_{past} \\ X \end{bmatrix}, \quad x_i \in \mathcal{X}, \quad u_i \in \mathcal{U}$$

N:	Prediction horizon	$H_L(u_{data})$:	Collected input data
x_i:	State (of X) in i^{th} step	$H_L(x_{data})$:	Collected state data
u_i:	Input (of U) in i^{th} step	u_{past}:	Input of last step
x_{des}:	Desired landing position	x_{past}:	State of last step
Q:	State stage cost matrix	$\mathcal{X}$:	State constraints
R:	Input stage cost matrix	$\mathcal{U}$:	Input constraints