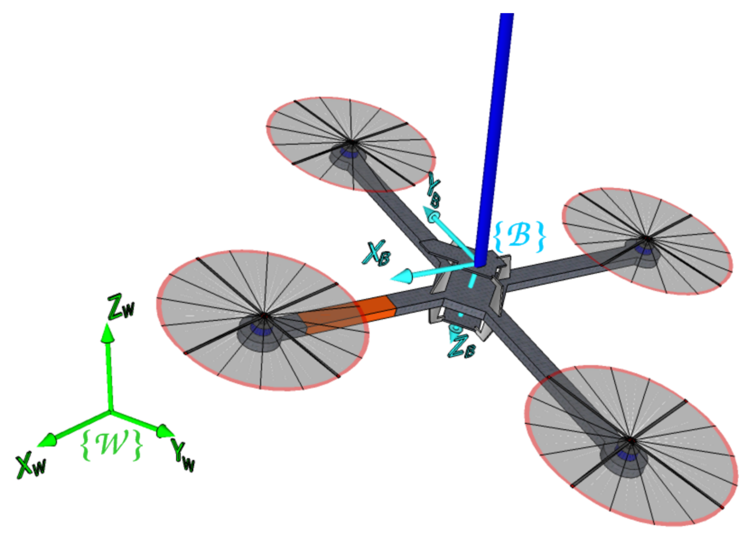
Balancing an inverted pendulum on a quadcopter with reinforcement learning

Pierre Lachèvre, Javier Sagastuy, Elise Fournier-Bidoz, Alexandre El Assad

Stanford University

Motivation

- Current quadcopter stabilization is done using classical PID controllers. They usually perform well expect for :
  - **Altitude control** due to complex airflow interactions present in the system
  - when **non-linearities** are introduced, which is the case in clustered environments.
- RL avoids creating manually-tuned control algorithms to complete specific tasks.



Dynamics of inverted pendulum on a quadcopter [2]

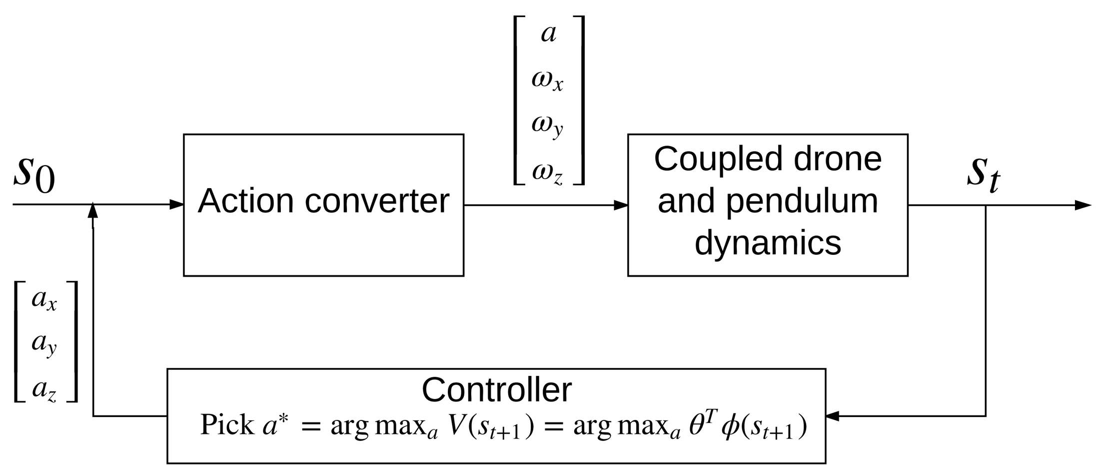
- Our approach builds an RL controller to balance an inverted pendulum on a drone modelled with the dynamics shown in [1].We aim to keep the drone stable at a position in 3D space while balancing the pendulum even when perturbations such as wind are introduced.

Problem definition

Three main parts constitute the project :

- The simulator of the quadcopter and inverted pendulum
- The reinforcement learning algorithm
- The reward function maximized by the RL algorithm

Once we have learned a controller using RL, the system evolves as follows:



A first approach: flying platform

We coded a Matlab simulator based on the copupled dynamics outlined in the paper by Hehn et.al. [1]. To get a better understanding of how to solve this problem we started with a simplified dynamics model:

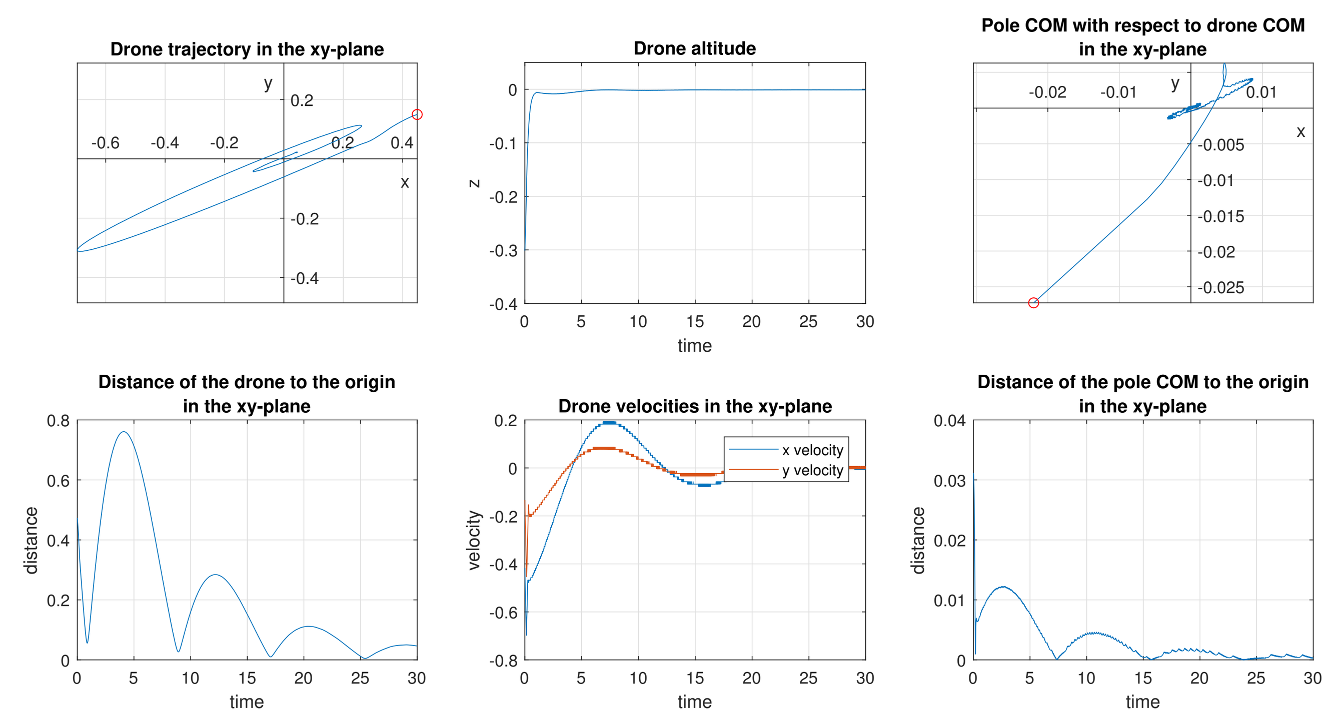
- The state vector X is $[x, y, z, \dot{x}, \dot{y}, \dot{z}, a, b, \dot{a}, \dot{b}] \in \mathbb{R}^{10}$ (position and velocity of the drone and the pendulum center of mass)
- The actions are the acceleration rates for the center of mass of the drone: $[a_x, a_y, a_z] \in \mathbb{R}^3$. They were discretized into a 5^3 dimensional space.
- Time was updated discretely at 0.01 second intervals.

Solving strategy

- Since our state space is continuous, we approximate our value function as $V(s) = \theta^T\phi(s)$ where $\phi(s)$ is a high dimensional feature mapping of our state space. We obtain the approximation $V(s)$ using fitted value iteration.
- $\phi(s)$ is a combination of quadratic terms from s.
- Our reward function was simply -1, whenever the drone was too far from the origin or the displacement of the center of mass of the pendulum was too large, 0.01 otherwise.

Without perturbations

We first analyzed how the system evolved when we had no external perturbations.



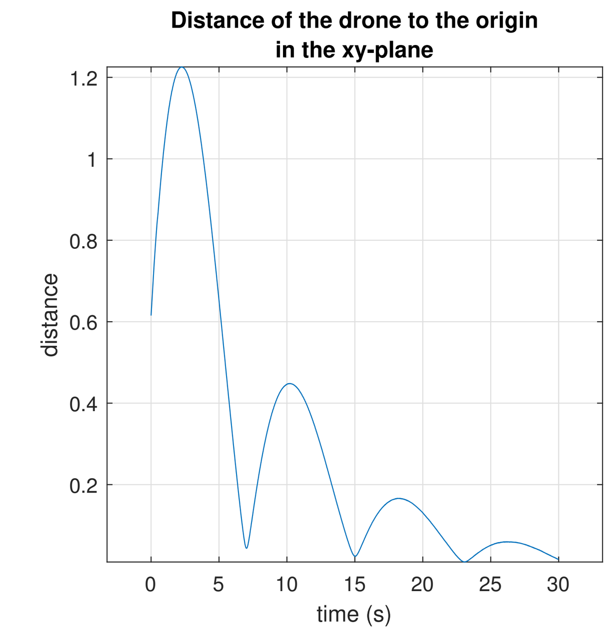
Performance of the system without perturbations

With perturbations

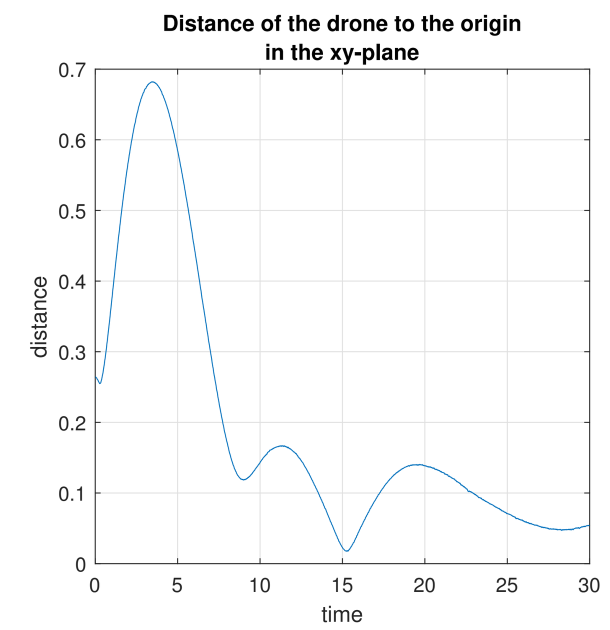
When we introduced randomly distributed wind, our approach was still able to stabilize the drone and keep the inverted pendulum upright. We also observed the average action was compensating for the added wind.

Incorporating full quadcopter dynamics

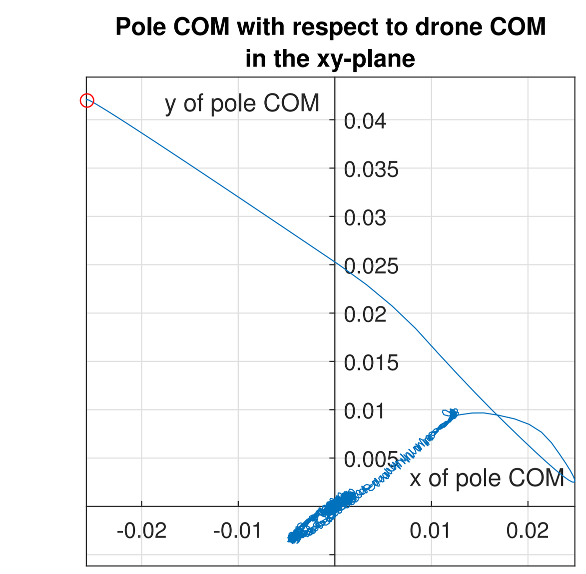
- The state vector X is $[x, y, z, \dot{x}, \dot{y}, \dot{z}, \alpha, \beta, \gamma, a, b, \dot{a}, \dot{b}] \in \mathbb{R}^{13}$ (position and velocity of the drone and the pendulum center of mass and the Euler angles).
- The actions are $[a, w_x, w_y, w_z] \in \mathbb{R}^4$ (the mass normalized collective thrust, rotational rates about the vehicle body axis).



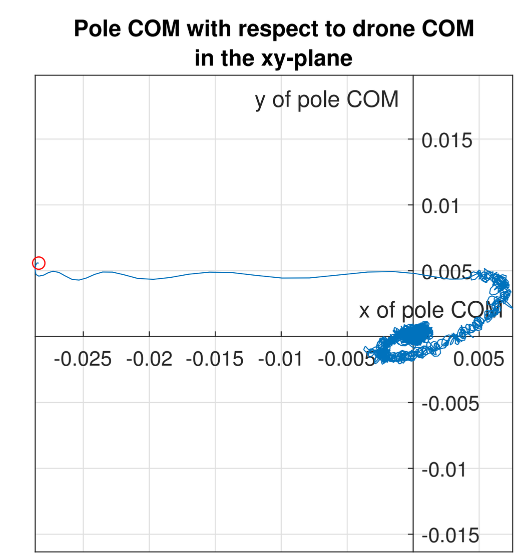
(a) Distance to origin without perturbations



(b) Distance to origin with perturbations



(c) Pendulum center of mass displacement without perturbations



(d) Pendulum center of mass displacement with perturbations

Conclusions and Future Work

- The main difficulty was to find an appropriate reward function and suitable feature mapping. An approach using deep learning to find a better feature mapping could be explored.
- We hope to test our learned policy on a real drone.

References

[1] Hehn, Markus, and Raffaello D'Andrea. "A flying inverted pendulum." Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011.

[2] Figueroa, Rafael, et al. "Reinforcement learning for balancing a flying inverted pendulum." Intelligent Control and Automation (WCICA), 2014 11th World Congress on. IEEE, 2014.

{efb, lpierre, jvrsgsty, aelassad}@stanford.edu