Drone Search and Rescue Project

EL2425 - Automatic Control Project Course

Allan Andre do Nascimento
Cheerudeep Chintha
Damien Rannenberg

Lukas Bjarre
Marc Sigonius

Contents

- Introduction
- Project overview
- PDDL
- Trajectory generation
- Controllers
- Demonstration

Drones

CrazyFlie 2.0 - Small & affordable drone

Applications

03
Project Motivation
01
02

Beyond Search and Rescue

Domino’s case of pizza delivery (New Zealand - 2016) [1]

Catastrophical Events

Inaccessible Zones

War Zones

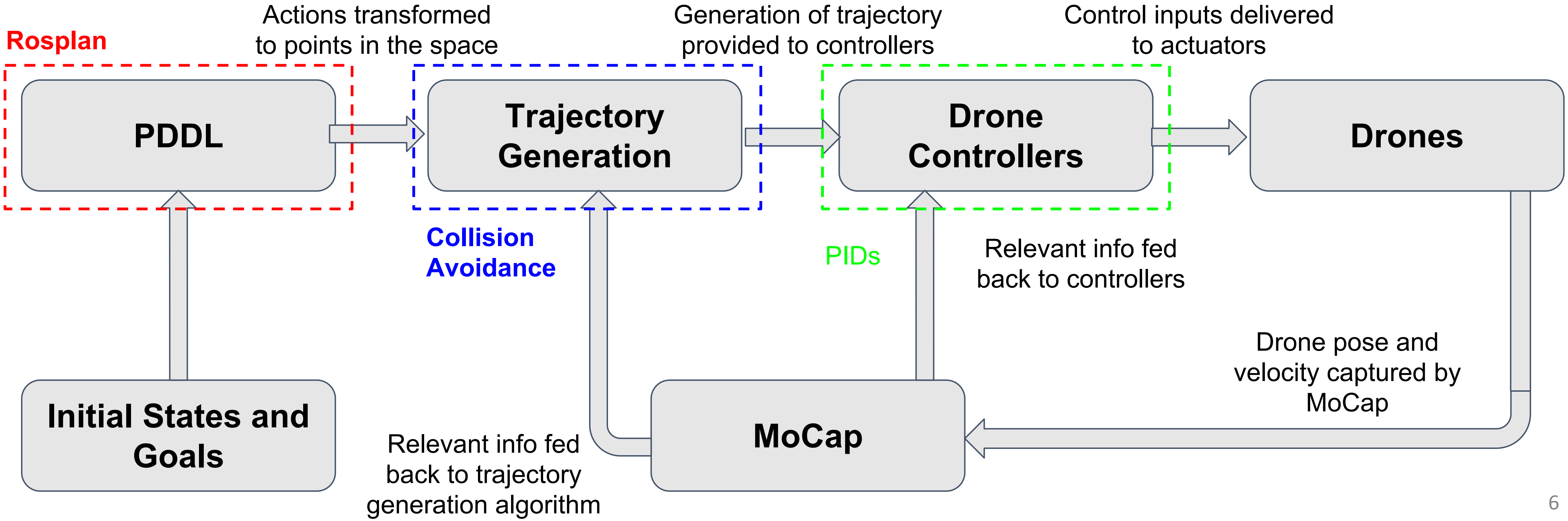
Let the drones take the risk

Introduction

- Project Main Task:
  - Design and implement a multi-agent system capable of transporting (virtual) supplies to selected regions.
  - Drones must work together without collision.
  - Plan of actions must be done automatically

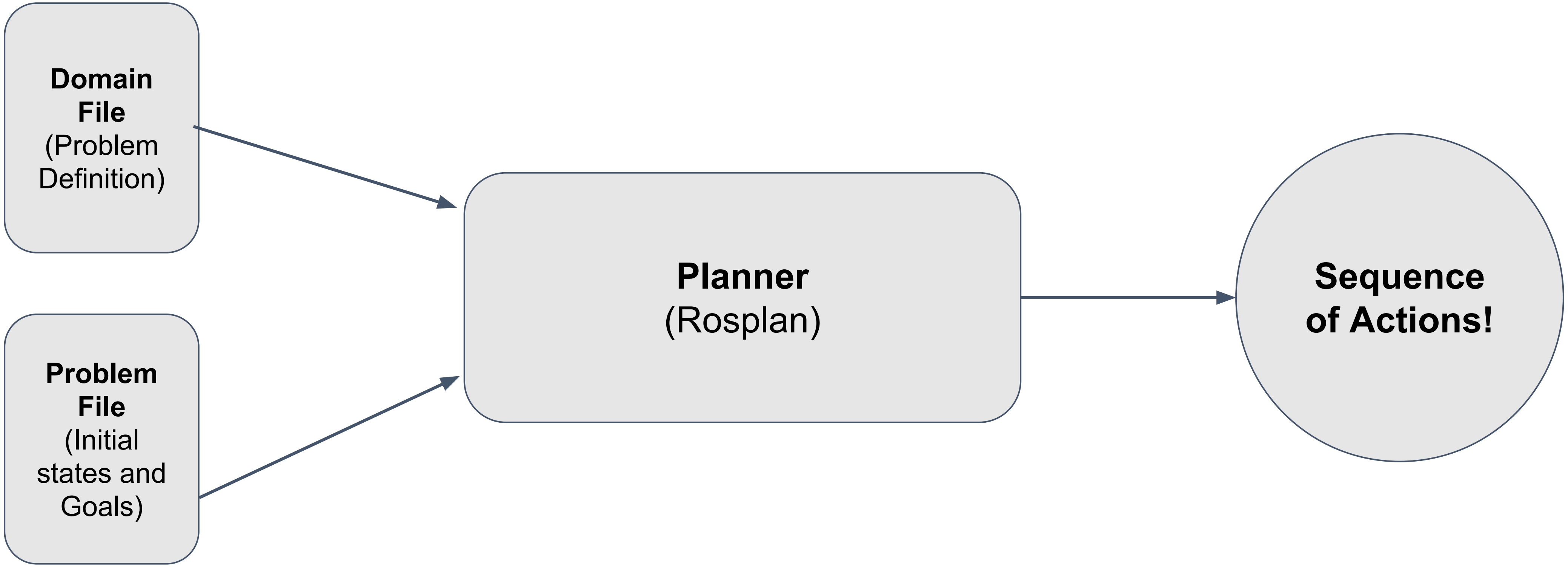
Project Overview

- **How does it work?**


Rosplan
Actions transformed to points in the space
Generation of trajectory provided to controllers
Control inputs delivered to actuators
PDDL
Trajectory Generation
Drone Controllers
Drones
Collision Avoidance
PIDs
Relevant info fed back to controllers
Drone pose and velocity captured by MoCap
Initial States and Goals
MoCap
Relevant info fed back to trajectory generation algorithm

PDDL

- Planning Domain Description Language



Domain File (Problem Definition)

Problem File (Initial states and Goals)

Planner (Rosplan)

Sequence of Actions!

Domain File

```
(define (domain automated-search-rescue)

    (:requirements :typing)

    (:types drone package location)

    (:constants warehouse1 - location)

    (:predicates
        (at-drone ?d - drone ?l - location)
        (drone-ground ?d - drone)
        (drone-fly ?d - drone) )

    (:action TAKEOFF
        :parameters (?d - drone)
        :precondition (drone-ground ?d)
        :effect (and (not (drone-ground ?d))
                    (drone-fly ?d))
    )
```

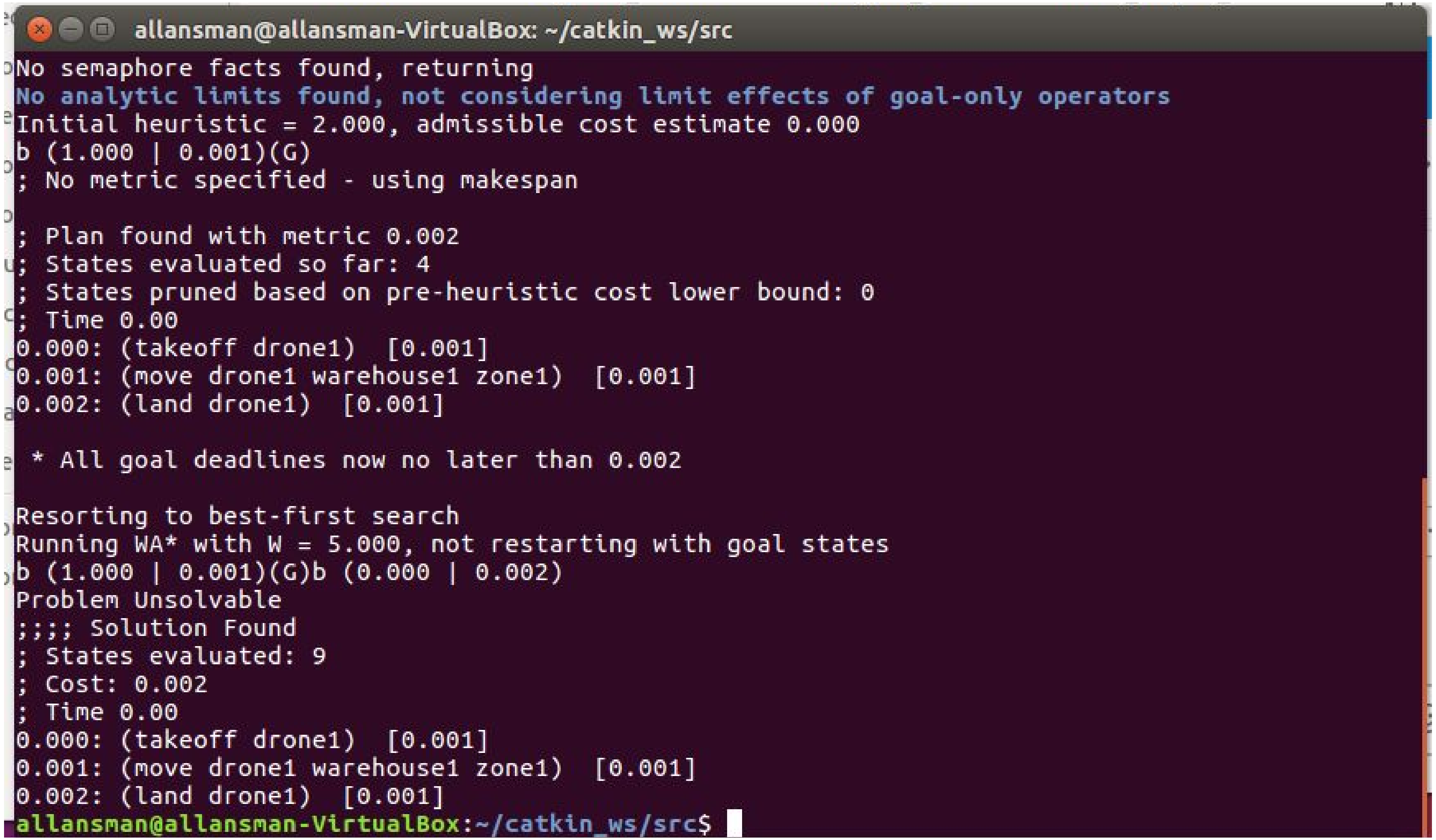
```
(:action LAND
        :parameters (?d - drone)
        :precondition (drone-fly ?d)
        :effect (and (drone-ground ?d)
                    (not (drone-fly ?d)))
    )


    (:action MOVE
        :parameters (?d - drone ?from ?to - location)
        :precondition (and (at-drone ?d ?from)
                            (drone-fly ?d))
        :effect (and (at-drone ?d ?to)
                    (not (at-drone ?d ?from)))
    )
)
```

Problem File

```
(define (problem search-rescue)

    (:domain automated-search-rescue)

    (:objects drone1 - drone
              zone1 - location
    )

    (:init
        (at-drone drone1 warehouse1)
        (drone-ground drone1)

    )
```

```
(:goal (and
        (at-drone drone1 zone1)
        (drone-ground drone1)

        )
    )
)
```

PDDL

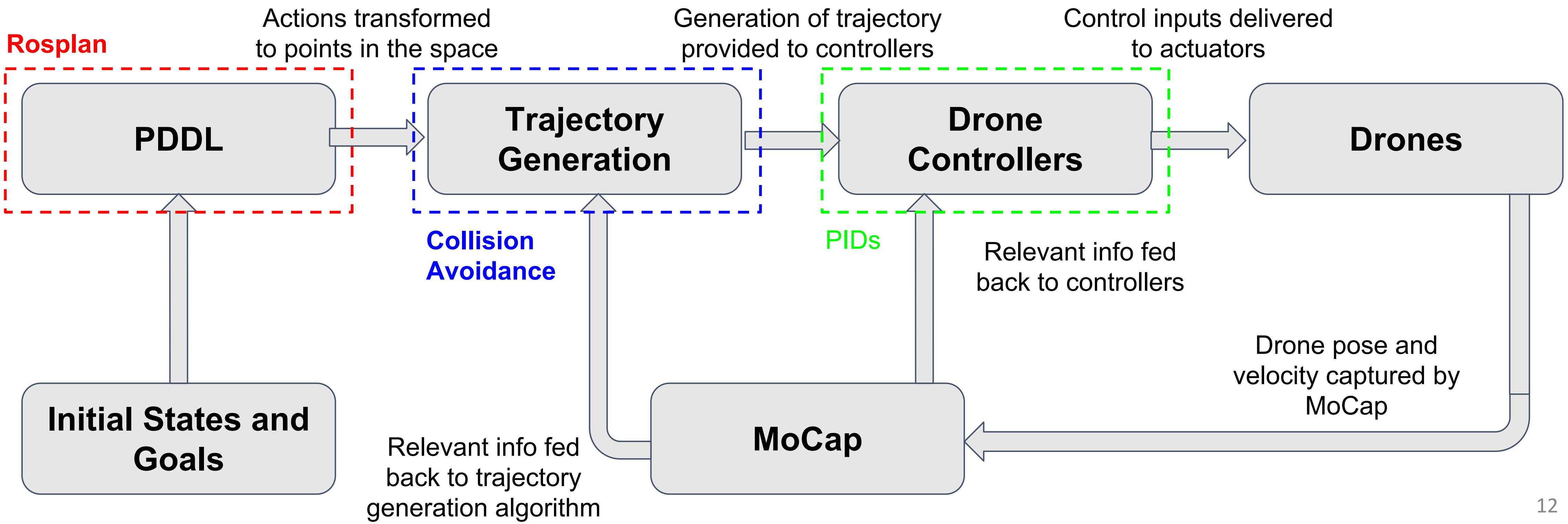


```
No semaphore facts found, returning
No analytic limits found, not considering limit effects of goal-only operators
Initial heuristic = 2.000, admissible cost estimate 0.000
b (1.000 | 0.001)(G)
; No metric specified - using makespan

; Plan found with metric 0.002
; States evaluated so far: 4
; States pruned based on pre-heuristic cost lower bound: 0
; Time 0.00
0.000: (takeoff drone1)  [0.001]
0.001: (move drone1 warehouse1 zone1)  [0.001]
0.002: (land drone1)  [0.001]

 * All goal deadlines now no later than 0.002

Resorting to best-first search
Running WA* with W = 5.000, not restarting with goal states
b (1.000 | 0.001)(G)b (0.000 | 0.002)
Problem Unsolvable
;;;; Solution Found
; States evaluated: 9
; Cost: 0.002
; Time 0.00
0.000: (takeoff drone1)  [0.001]
0.001: (move drone1 warehouse1 zone1)  [0.001]
0.002: (land drone1)  [0.001]
allansman@allansman-VirtualBox:~/catkin_ws/src$ 
```

PDDL

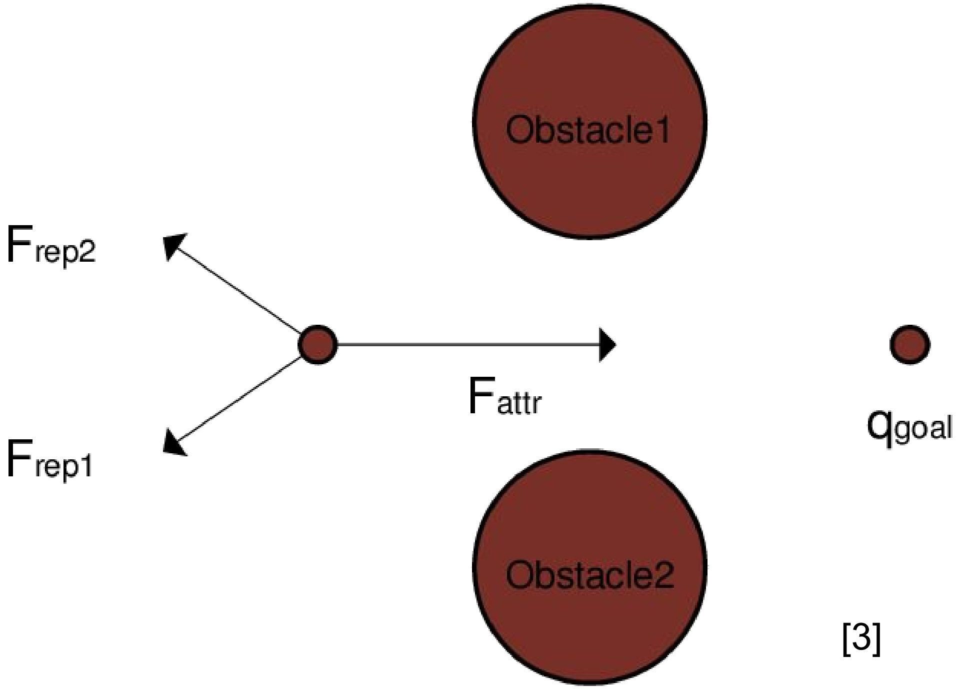
- In this project
  - A cost function was defined (optimization purposes)
  - Actions are: Takeoff, Land, Load, Unload and Move
  - Solver used is Optic (makes use of Cplex)
  - Plan is given in parallel to drones

Project Overview

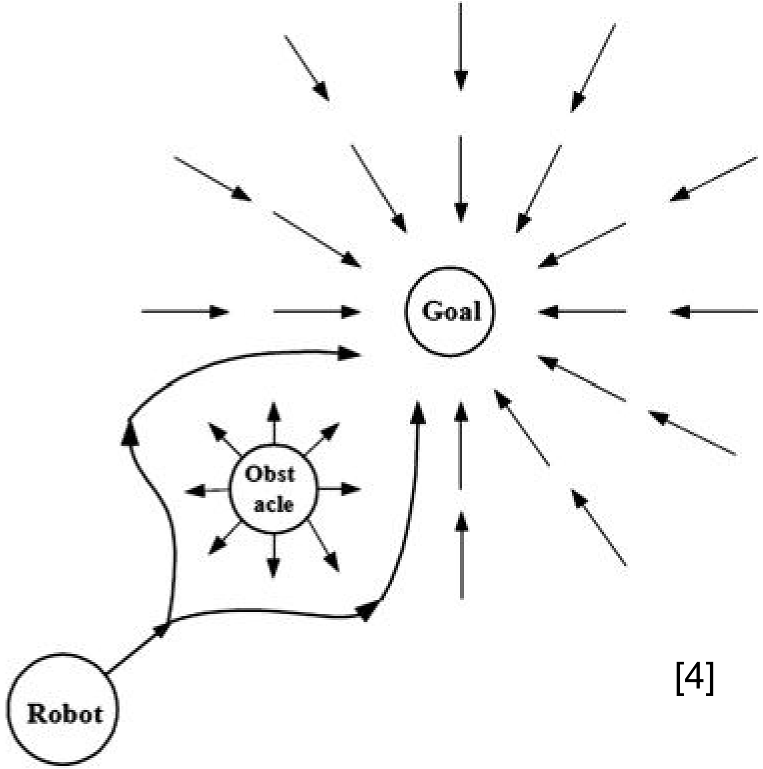

Rosplan
Actions transformed to points in the space
Generation of trajectory provided to controllers
Control inputs delivered to actuators
PDDL
Trajectory Generation
Drone Controllers
Drones
Collision Avoidance
PIDs
Relevant info fed back to controllers
Drone pose and velocity captured by MoCap
Initial States and Goals
MoCap
Relevant info fed back to trajectory generation algorithm

Trajectory generation

- **Originally based on the artificial potential field method [2]**



[3]



[4]

Trajectory generation

In this project:

- **Collision avoidance is included by the “repulsive term”**

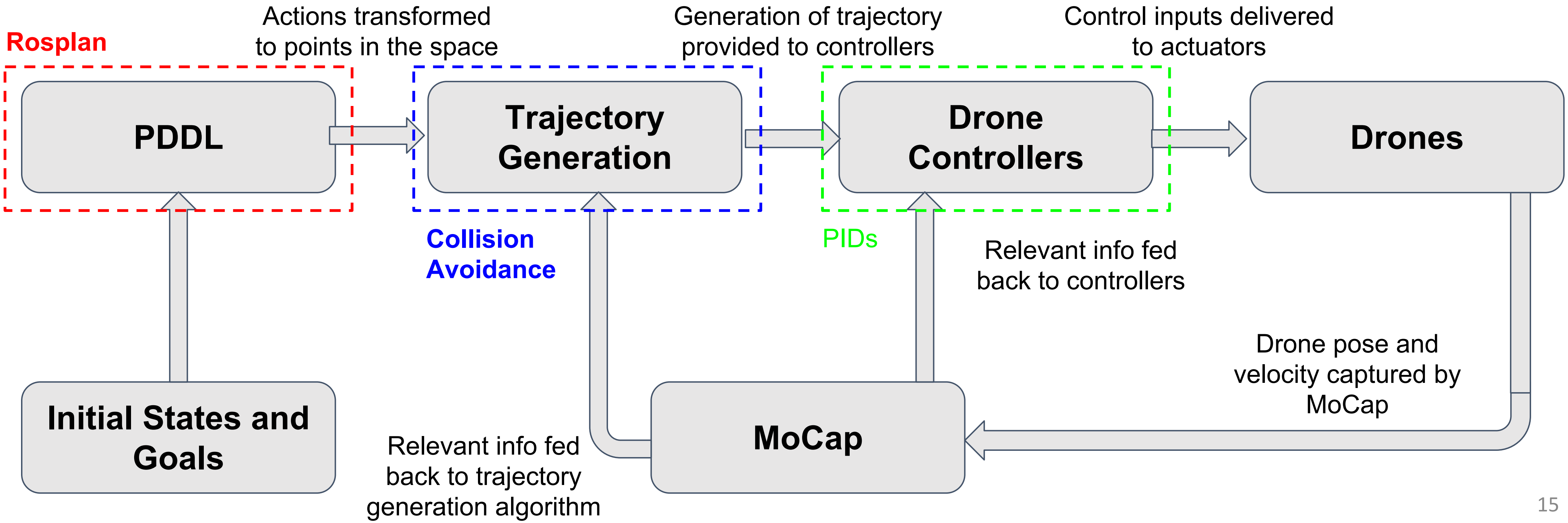
$$\sum_{combinations(i,j)} \frac{1}{Q_1||x_i - x_j||_2}$$

- **Goal attained by minimization of “attractive term”**

$$Q_2||X - X_{target}||_2$$

- **Optimizing over multiple time steps with constraints in distance**

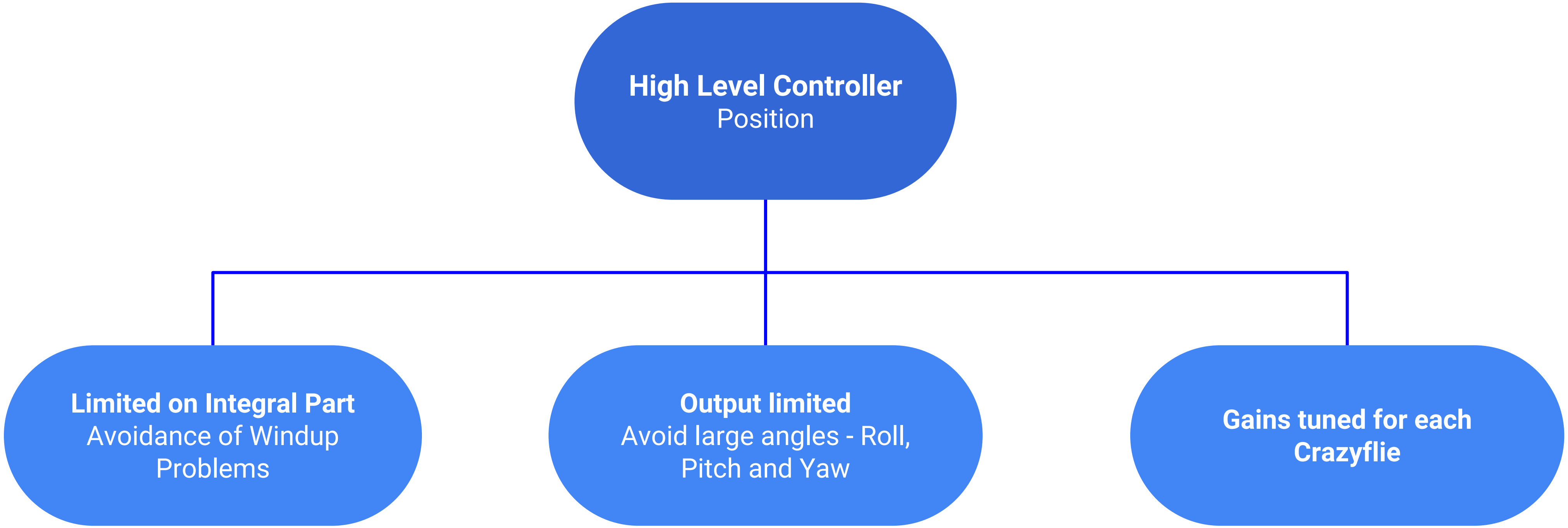
Project Overview


Rosplan
PDDL
Actions transformed to points in the space
Trajectory Generation
Collision Avoidance
Generation of trajectory provided to controllers
Drone Controllers
PIDs
Control inputs delivered to actuators
Drones
Initial States and Goals
Relevant info fed back to controllers
Drone pose and velocity captured by MoCap
MoCap
Relevant info fed back to trajectory generation algorithm

Controllers

	Low Level Controller Attitude (Roll, Pitch, Yaw)	**High Level Controller** Position
Type	PID	PID
Input	Target Attitude & Current Attitude	Target Position (Traj. gen) & Current Position (MoCap)
Output	PWM for individual motors	Yaw, Pitch, Roll - To low level controller
Frequency		30 Hz
Implementation	Controller already on Crazyflie	Controller Implemented as a ROS Node

Controllers



High Level Controller
Position

Limited on Integral Part
Avoidance of Windup Problems

Output limited
Avoid large angles - Roll, Pitch and Yaw

Gains tuned for each Crazyflie

Demonstration

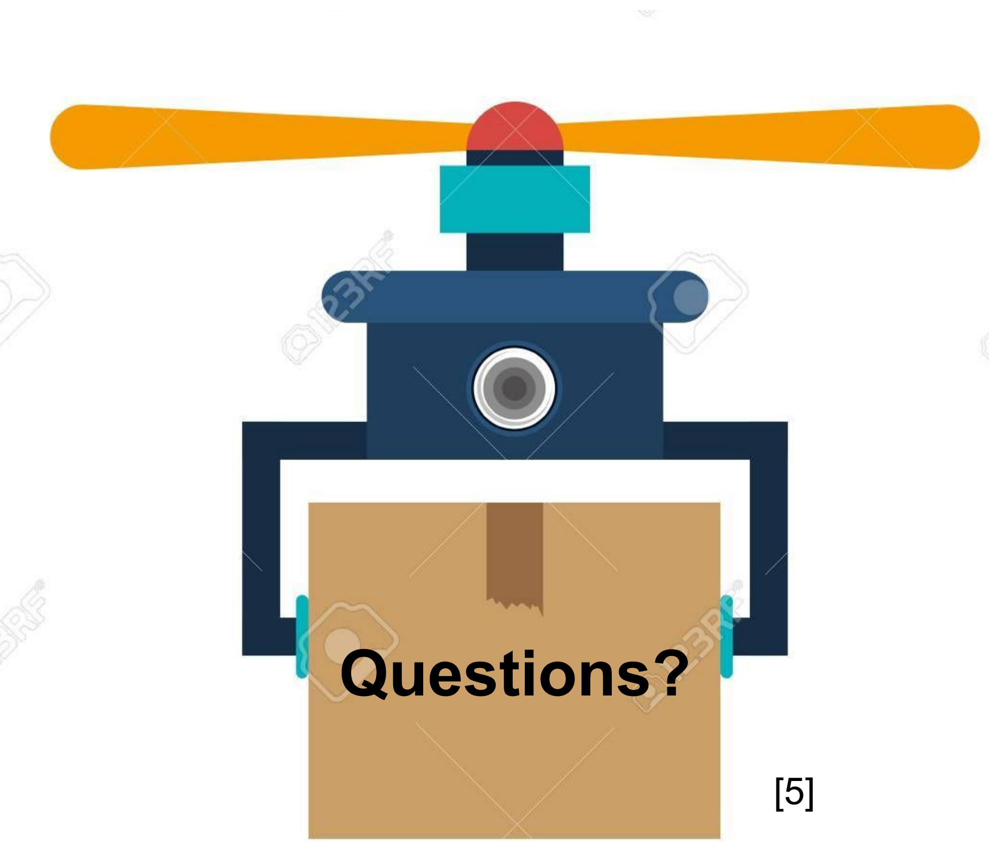
- **Today’s demonstration will be done with:**
  - 4 Drones
  - 4 Departure areas
  - 4 Rescue zones

Demonstration

Four working drones from 4 departure areas and 4 rescue zones

Thank you!

[5]



Questions?

[5]

References:

[1]-https://www.cnbc.com/2016/11/16/dominos-has-delivered-the-worlds-first-ever-pizza-by-drone-to-a-new-zealand-couple.html

[2] - http://journals.sagepub.com/doi/abs/10.1177/027836498600500106

[3]-https://www.researchgate.net/profile/Nadeem_Javaid/publication/237054021/figure/fig1/AS:393251568799747@1470769888555/Fig1-Dead-end-scenario-of-Artificial-Potential-Field-method-symmetric-obstacles.png

[4] - https://ars.els-cdn.com/content/image/1-s2.0-S1367578812000430-gr7.jpg

[5]-https://previews.123rf.com/images/grgroup/grgroup1610/grgroup161001207/63909993-drone-carrying-box-with-one-airscrew-vector-illustration-Stock-Vector.jpg

Demonstration

Slide possibly to be shown in case things don’t work as expected