Project Final

Trajectory Generation and Following with Position Correction

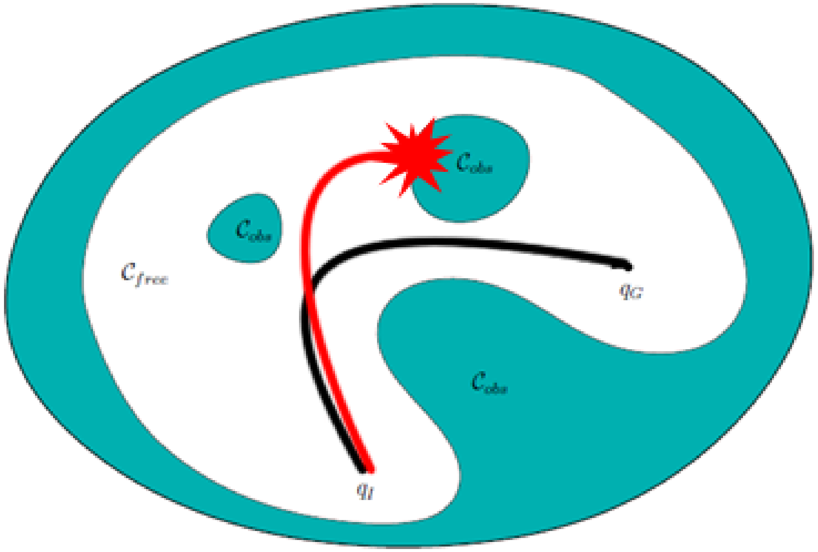
Team: Crash Pilots

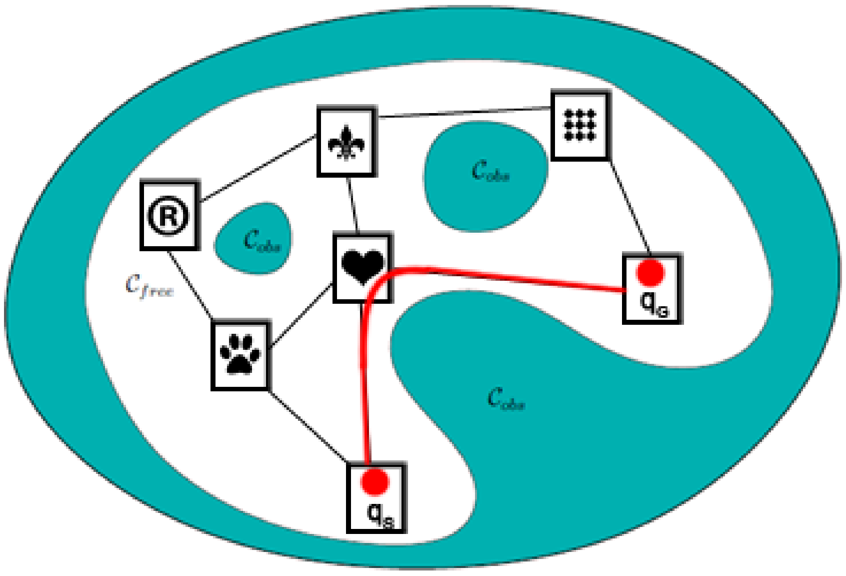
Sören Jentzsch
Sebastian Riedel
Peter Gschirr

Date: 17.07.2012

Project Proposal

- Autonomous flying requires **path-planning**
  ➔ for **optimality** and **collision avoidance**
- Quadrocopter should fly along defined path as closely as possible
  ➔ **use visual landmarks**





Problem Formulation

1. Given a set of predefined **waypoints** $V = \{q_s, q_1, \ldots, q_{n-1}, q_g\}$
   with **edges** $E = \{(q_s, q_1), (q_1, q_2), \ldots, (q_{n-1}, q_g)\}$
   a **trajectory** $\theta(t) = (x(t), y(t))^T$ (using parabolic blends)
   should be generated for the quadrocopter such that:

$$\theta(0) = q_s \qquad \frac{d\theta(t)}{dt} < v_{max}, \forall t$$
$$\theta(t_{end}) = q_g$$
$$\theta(t) \in \mathcal{C}^1 \qquad \frac{d^2\theta(t)}{d^2t} < a_{max}$$

   Each node V consists of a **landmark** for localization.

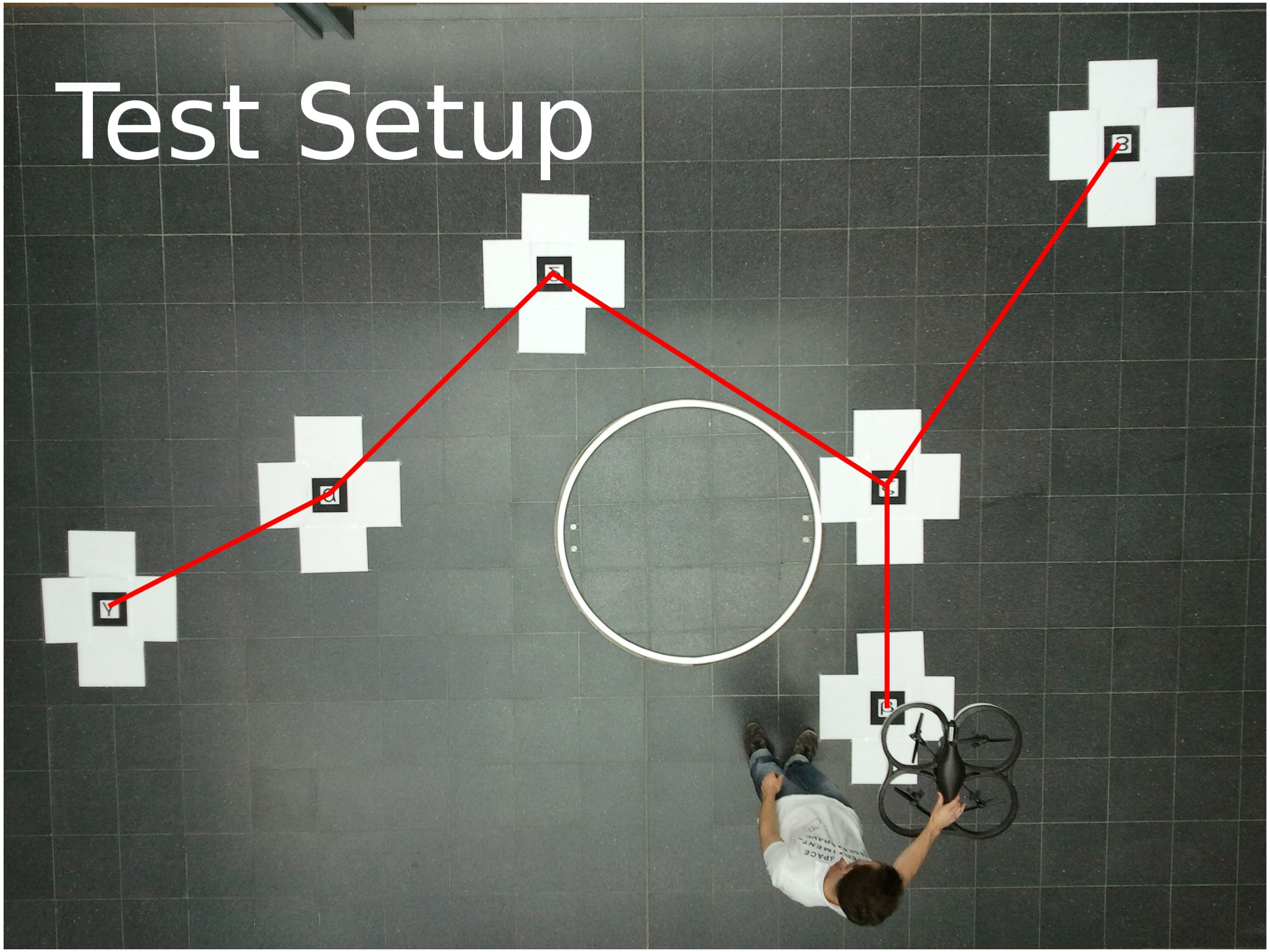
2. The quadrocopter should fly along the computed trajectory, **update** and **correct** its position according to **external sensor measurements.**

   ➔ **Minimizing distance to real path**

Implementation

1. **configurable** graph representation with **Boost Graph Library**
2. **user selection** of a destination marker
3. calculation of shortest distance ==> generates **list of via-points**
4. **2 approaches** for following the path
   - trajectory generation with parabolic blends
   - iterative step-by-step destination relocation (for evaluation)
   - → both with the same **PD-Controller**

Test Setup



Results

- Evaluation hard to make due to some **airflow** and some other random fluctuations → **Video-based evaluation**
- trajectory-based control **smoother** than the basic waypoint-based control
- trajectory hard to see, differs from the one in Rviz (reference control) quite a bit due to a **simplified model**
- **unsteady accurancy**
- need for **active slow-down procedure** due to the lack of friction

Summary and Conclusion

- Despite facing problems with the **accuracy**, we think that trajectory based controller has **more potential** than a step-by-step regulation (control gains and desired velocity are independent).
- For a more precise evaluation, we would need an **external tracking system**
  - calculate RMS-position error for trajectory following
  - exact time-to-destination measurements
- For the next step of this project, we would recommend implementing a **more sophisticated control model**
  - take into account: **inertia, dynamics, friction, ...**