Testing UAVs

Alexander Pretschner

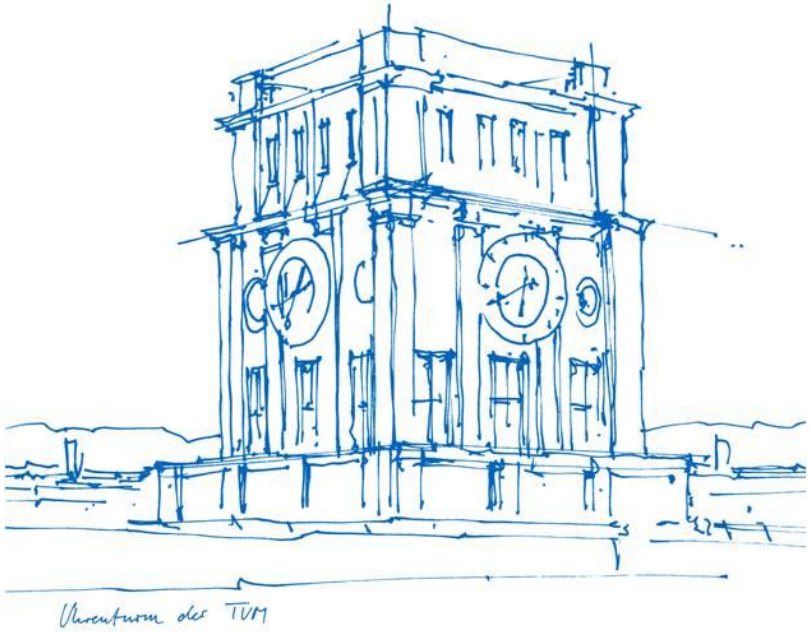
TUM | fortiss | bidt

jww Tabea Schmidt, Florian Hauer

Assured CPS Autonomy for 3D Urban Transportation:

Drones, Flying Cars and Beyond

June 9th 2021



Motivation

https://www.youtube.com/watch?v=crHdX7ODvYo

Background: Testing ADAS

Test case generation for automated driving systems: try to find an environment (street layout, other cars, their environment) that leads to a violation of safety distance

Cannot reuse pre-recorded drives –
but can cluster recorded drives to infer scenario types; then use these to find „extreme“ instances

Testing cars is simple: We have regulation and
scenario types!

Background: Testing ADAS

Test case generation for automated driving systems: try to find an environment (street layout, other cars, their environment) that leads to a violation of safety distance

Cannot reuse pre-recorded drives – but can cluster recorded drives to infer scenario types; then use these to find „extreme“ instances

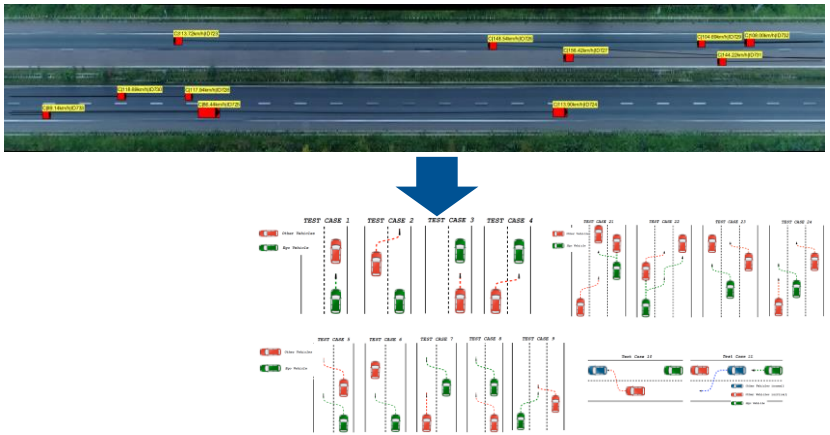
Testing cars is simple: We have regulation and scenario types!

Background: Testing ADAS

Test case generation for automated driving systems: try to find an environment (street layout, other cars, their environment) that leads to a violation of safety distance

Cannot reuse pre-recorded drives – but can cluster recorded drives to infer scenario types; then use these to find „extreme“ instances

Testing cars is simple: We have regulation and scenario types!

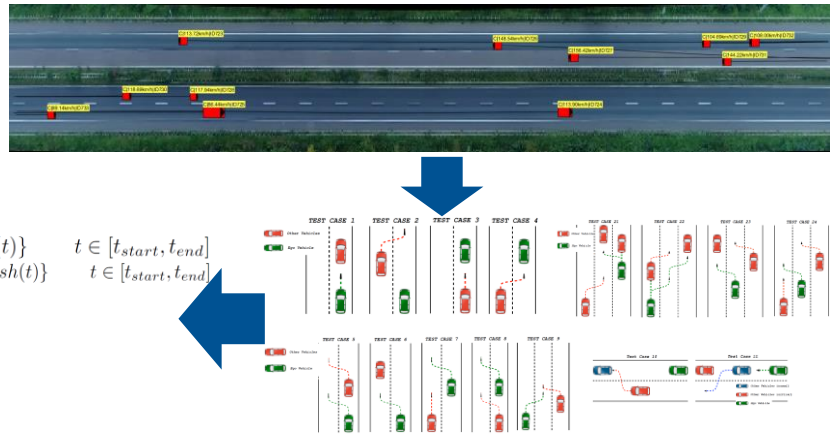


Background: Testing ADAS

Test case generation for automated driving systems: try to find an environment (street layout, other cars, their environment) that leads to a violation of safety distance

Cannot reuse pre-recorded drives – but can cluster recorded drives to infer scenario types; then use these to find „extreme“ instances

Testing cars is simple: We have regulation and scenario types!



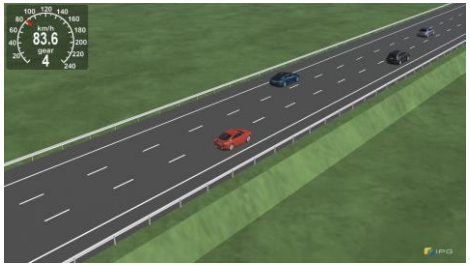
$$f = \min\{d(t) - safeDist(t)\} \quad t \in [t_{start}, t_{end}]$$
$$f = \min\{actualValue(t) - envelopeTrsh(t)\} \quad t \in [t_{start}, t_{end}]$$

Background: Testing ADAS

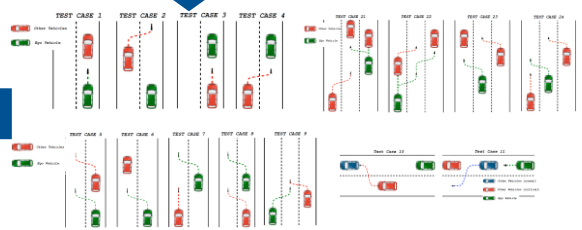
Test case generation for automated driving systems: try to find an environment (street layout, other cars, their environment) that leads to a violation of safety distance

Cannot reuse pre-recorded drives – but can cluster recorded drives to infer scenario types; then use these to find „extreme“ instances

Testing cars is simple: We have regulation and scenario types!



$$f = \min\{d(t) - safeDist(t)\} \quad t \in [t_{start}, t_{end}]$$
$$f = \min\{actualValue(t) - envelopeTrsh(t)\} \quad t \in [t_{start}, t_{end}]$$



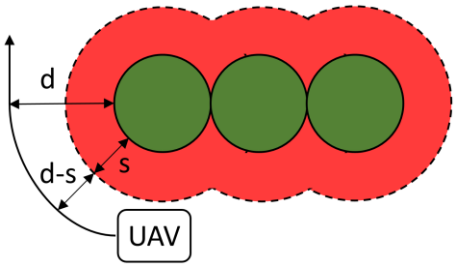
Understanding Safety of UAVs in Urban Areas – Motivation

Challenges: No systematic tests for UAVs!

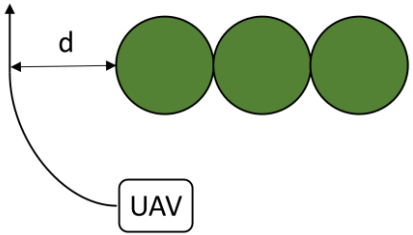
A. Testing the behavior of the UAV in various scenarios.
B. Ensuring that it behave safely even in the most challenging situations (worst-case situations).
C. Explicitly defining the safe behavior of a UAV in each possible situation.
D. “Good” tests? Those that reveal *potential* defects, with good cost effectiveness. Optimization problem.

Two cases:

(i) A safety distance s is specified,
UAV needs to keep distance $d > s$.
Objective: minimize d-s



(ii) No safety distance can be specified,
worst-case situations need to be found.
Objective: minimize d



Generating “Good” Test Cases – Boundary Analysis Testing

In addition to minimizing distance, we need to encode geometry.

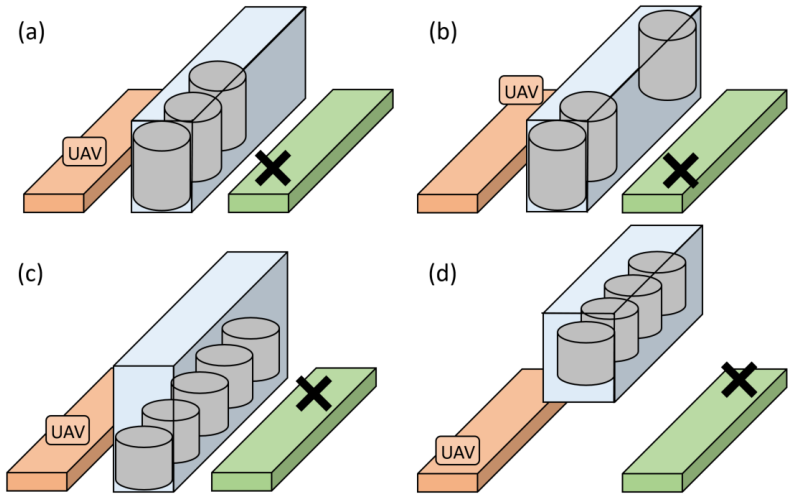
Fitness functions for Boundary Analysis Testing:

(a) fly around obstacles or (c) fly above them:

$$f = \begin{cases} \min\big(d(t)\big), & \text{if given logical scenario is displayed} \\ \infty, & \text{otherwise} \end{cases}$$

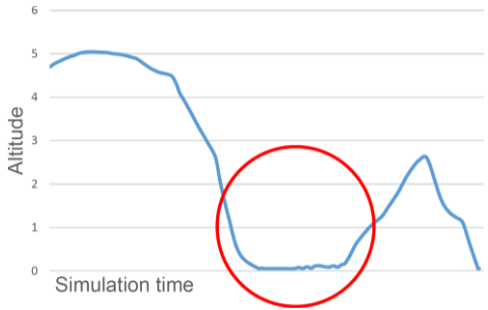
(b) fly through a gap of width w between two obstacles or (d) fly below them:

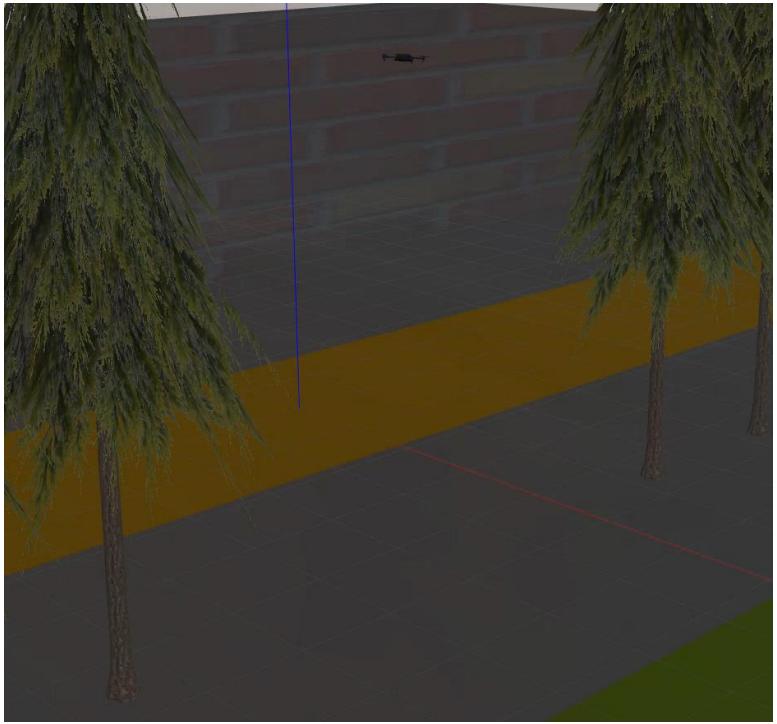
$$f = \begin{cases} \min(w), & \text{if given logical scenario is displayed} \\ \infty, & \text{otherwise} \end{cases}$$



Experimental Results – Boundary Analysis Testing

Scenario (b)
Gap width: 3.76 meters
→ Questionable behavior

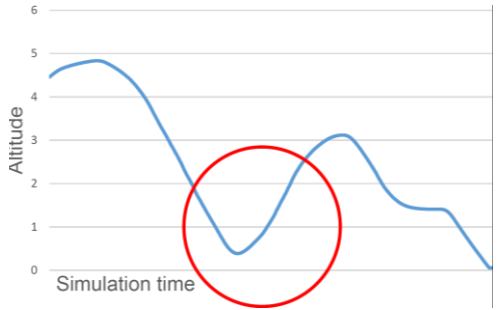


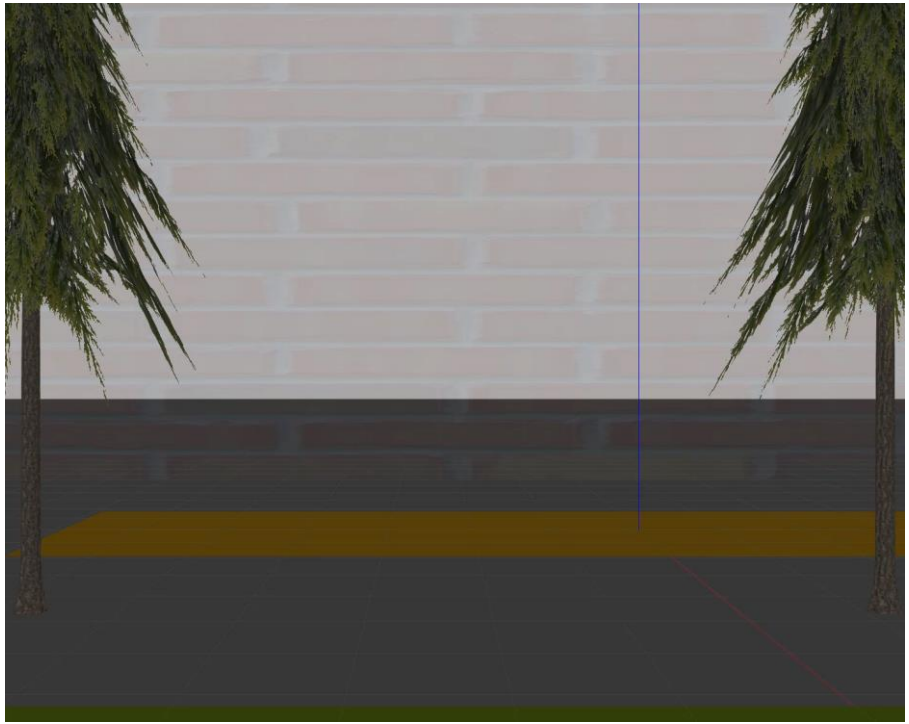


PX4 autopilot
+ obstacle avoidance

Experimental Results – Boundary Analysis Testing

Scenario (b)
Gap width: 4.77 meters
→ Safe behavior





PX4 autopilot
+ obstacle avoidance

Conclusion

Challenges for testing the safe behavior of UAVs:
Few rules! No scenario types yet! No definition of “safe” behavior!

Policy should include help, or a foundation, like RSS, on how to generate “good” test cases for testing the safe behavior of UAVs in urban environments?
→ use scenario-based testing and search-based techniques to generate challenging environments

In the **experiments**, we found several safety distance violations and questionable behaviors of the UAV.
→ shows the effectiveness and applicability of the proposed methodology

Outlook:

- More complex scenarios; co-operating drones; drone2X communication
- Estimate maximum number of obstacles necessary for testing; then shape/position/size in a second step