A Reference Ontology for Collision Avoidance Systems and Accountability

Authors

- D. Martín-Lammerding
- José Javier Astrain
- Alberto Córdoba

Presenter

D. Martín-Lammerding
Department of Stats. Comput. Sci. Math
Public University of Navarre (UPNA)
david.martin@unavarra.es

Resume

Telecommunications Engineer, 2000, at the Public University of Navarre.

Specialized in computer security and digital identity projects. I have developed my professional activities in different companies and sectors for more than 20 years.

My research work is focused on ontology-based autonomous systems and how to apply it to unmanned aircraft (UAS).

Introduction

- Unmanned Aerial Systems (UAS) traffic increase
- Safety risk
- Different Collision Avoidance Systems (CAS)
- Incidents with UAS, collisions.

Challenges

- How to manage CAS data?
- How to improve CAS operation?
- How to improve UAS safety?

Related Work

- Ontologies for sensors
- Air traffic data inter-operability
- Multiple CAS implementations, ACAS-Xu and Daidalus

Related Work

- Hardware: Black-boxes, flight recorder

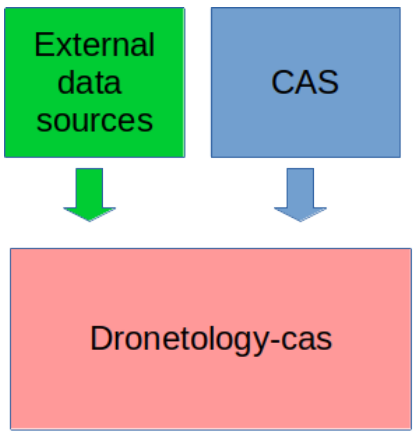
Contribution

- A novel ontology for CAS, Dronetology-cas
- Dronetology-cas constitutes a Knowledge base (KB) for CAS
- A foundation for a knowledge-based CAS

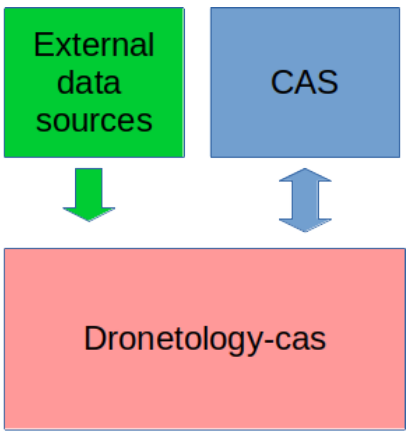
Dronetology-cas integration

a) Repository mode: The CAS data stored in a semantic structure

b) Knowledge mode: The CAS query for knowledge



(a)

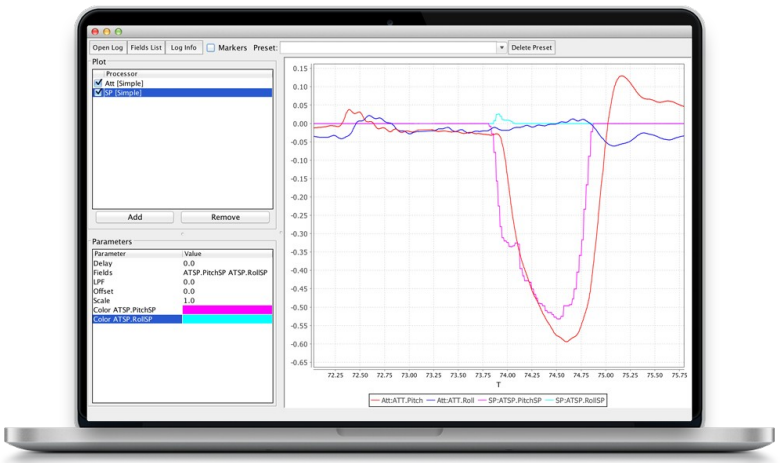


(b)

Repository mode

CAS’s data available for any audit process as triplets.

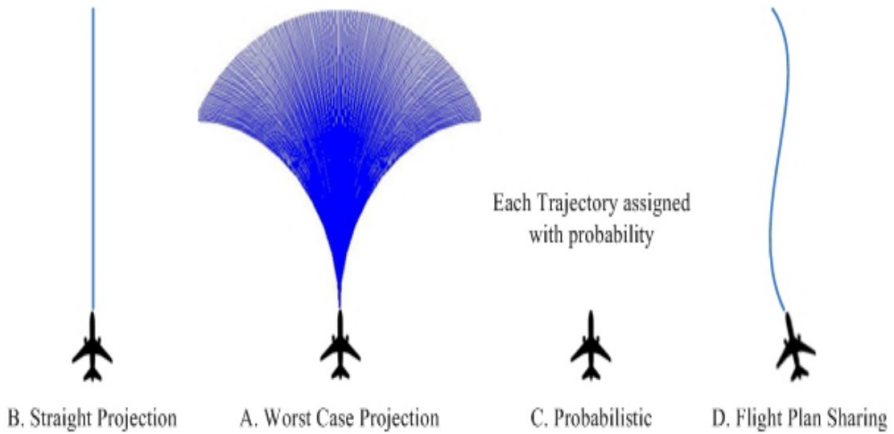
External data sources add additional conflicts



Knowledge mode

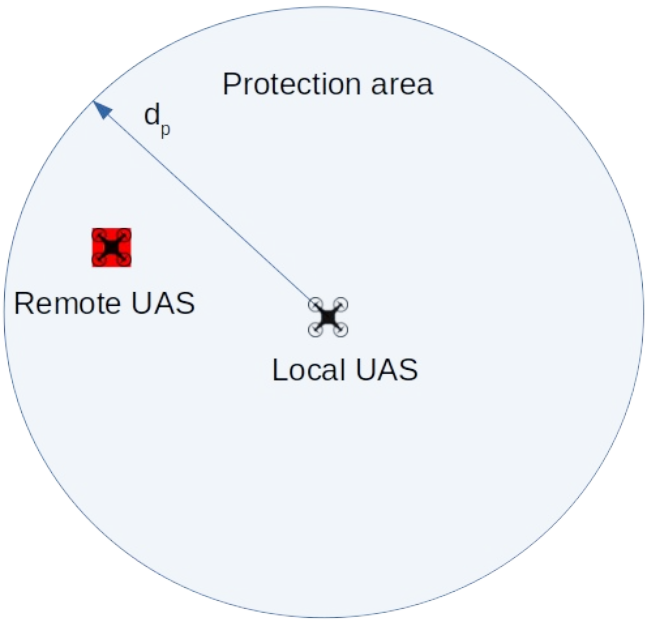
The CAS queries the KB

E.g CAS obtain a method to estimate conflict's position depending the conflict's knowledge available



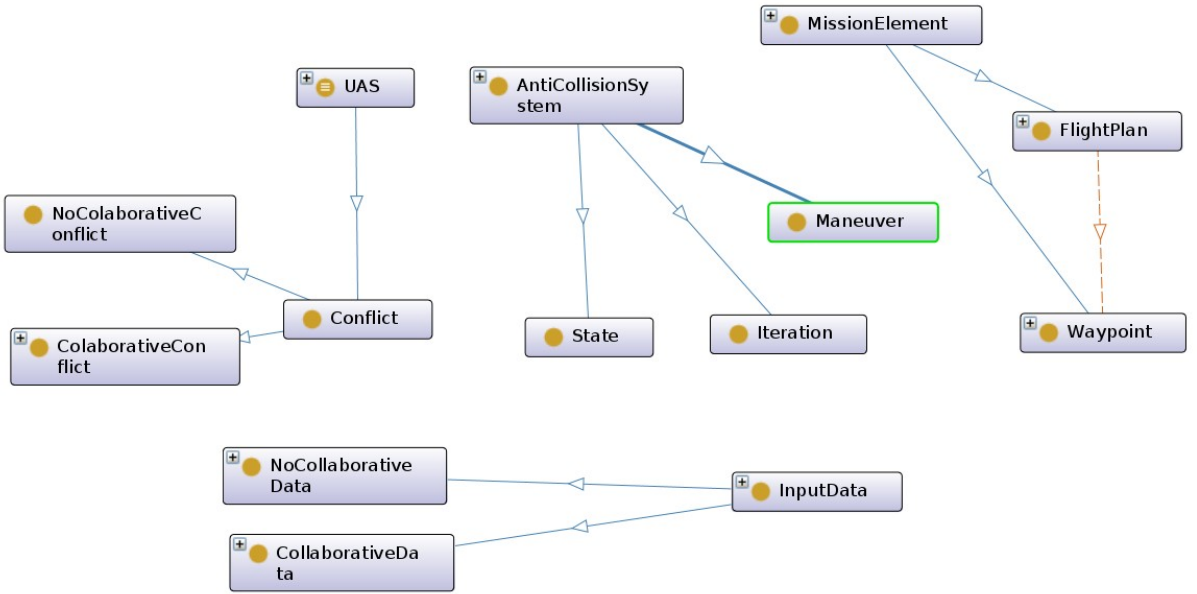
CAS's concepts in Dronetology-cas

- Protection distance
- Time to collision
- Conflict
- Input Data
- Iteration
- State
- Maneuver



Dronetology-cas: The application ontology

Generic concepts for any CAS



Competency questions

- How many conflicts are detected?
- Which conflict has the shortest time to collision?
- How has been detected the conflict with a given UAS?
- How long it has taken to resolve a conflict?

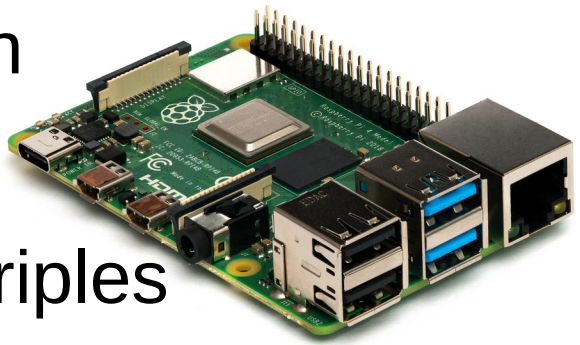
Performance evaluation

Used 2 CQ for knowledge extraction

Executed in a low cost computer, Pi3

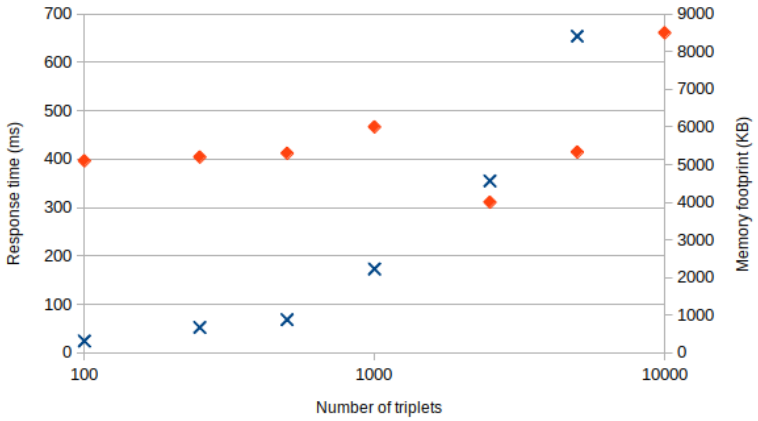
15 minutes fight, 10 triples/s, 10000 triples

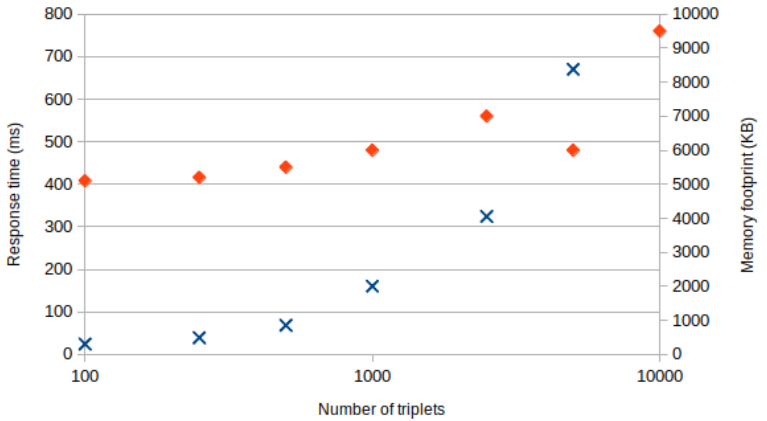
Software components developed in Java 8



Performance evaluation

- KB grows during UAS's flight
- Response time increase
- Response time (x) and memory footprint (◊)





Performance evaluation

- Response time is below the refreshing rate of incoming data when ≈ 1000 triplets
- Worst response time is at the end of a flight.

Conclusion

- Dronetology-cas integration modes facilitate its application in any CAS
- A CAS integration in a kowledge-mode implementation requires to balance memory consumption and response time

Future Works

- Implement and test Dronetology-cas with two KB, one for each mode.
- Integration of Dronetology-cas with an existing CAS.

Future Works

- Implementation of a CAS for UAS based on Dronetology-cas
- Create a dataset with semantic mission data
- Define an ontology standard for autonomous UAS.

Thank you