



Universitatea  
Politehnica  
Bucureşti



Facultatea de  
Automatică și  
Calculatoare



Catedra de  
Calculatoare

# Formation Flight for Unmanned Aerial Vehicles

---

Bachelor Thesis: July 10<sup>th</sup>, 2013

## Author:

Alexandru George Burgelea  
[aburghalea@rosedu.org](mailto:aburghalea@rosedu.org)

## Scientific Advisers:

Prof. Adina Magda Florea  
Ing. Mihai Trăscău  
Ş.I. Cătălin Leordeanu



- ▶ Domain
- ▶ Autonomous UAV
- ▶ Hirrus
- ▶ Platform Objectives
- ▶ Platform Simulation Architecture
- ▶ Autopilot Architecture
- ▶ Thesis Objectives
- ▶ Formation Flight
- ▶ Types of Formation
- ▶ Tested Formations
- ▶ Evaluation and Results
- ▶ Conclusions and Future Work



## **UAV** (*Unmanned Aerial Vehicle*)

- ▶ no pilot **on board**
- ▶ remote controlled or
- ▶ completely autonomous
- ▶ envisioned by N. Tesla in 1915
- ▶ used in military and civil missions
- ▶ rotor based or fixed-winged



Figure 1: Fixed-wing UAV with surmountable camera [1]



## About:

- ▶ in collaboration with *Teamnet International S.A.*
- ▶ aims to build a management platform for a fleet of UAVs

## Objectives:

- ▶ development of an autonomous flight software agent embedded on Raspberry PI for Hirrus
- ▶ development of a software platform for programming, monitoring and autonomous mission deployment for UAVs



- ▶ Ground Control System (based on QGroundControl)
- ▶ Mission Monitoring System
- ▶ Adding *collision avoidance* and *formation flight* capabilities to the Hirrus Autopilot
- ▶ Designing and implementing an embedded AI software agent responsible for mission management (re)planning and execution



Figure 2: Hirrus UAV [2]



Destination law enforcement, reconnaissance, search and rescue, cartography

Dimensions Wingspan 2.35 m / Length 1.1 m / Weight 7 kg

Speed Max 130 km/h, Cruise 90 km/h

Payload 0.7 kg

Propulsion Electric

Endurance 180 min

Range 15 km



## Used Technologies

- ▶ simulated using Rascall Drone

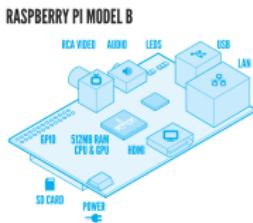


Figure 3: Raspberry PI



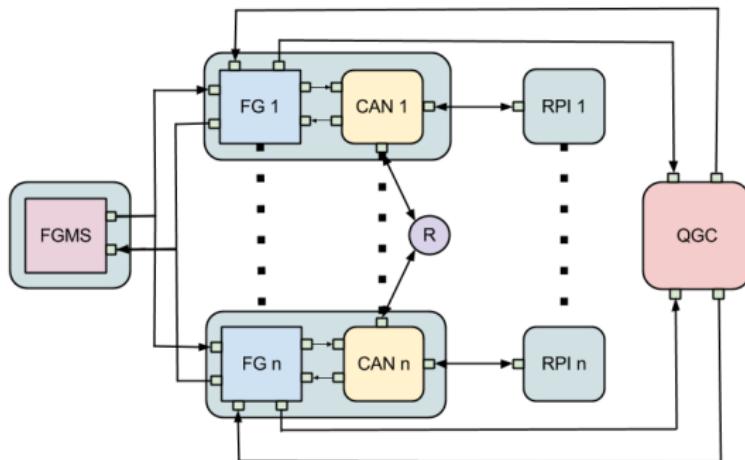
Figure 4: Flight Gear Flight Simulator



Figure 5: QGroundControl



# Platform Simulation Architecture



## Legend:

- FGMS** - Flight Gear Multi-player Server
- FG** - Flight Gear Flight Simulator
- CAN** - CAN bus simulator
- R** - Telemetry Router
- RPI** - Raspberry Pi
- QGC** - QGroundControl

Figure 6: Platform Architecture



# Embedded Software Agent Architecture

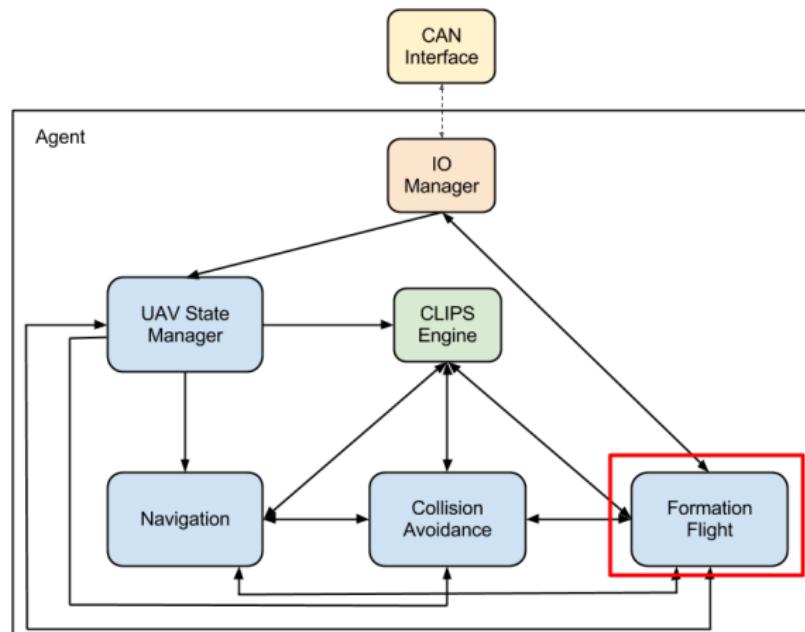


Figure 7: Embedded Software Agent Architecture

- ▶ close range formation flight module
- ▶ using decentralized communication
- ▶ multi agent system with reactive agents inspired by swarm structures and behavior models
- ▶ 3 or more UAVs flying in formation
- ▶ drones flying at close range

# Formation Types

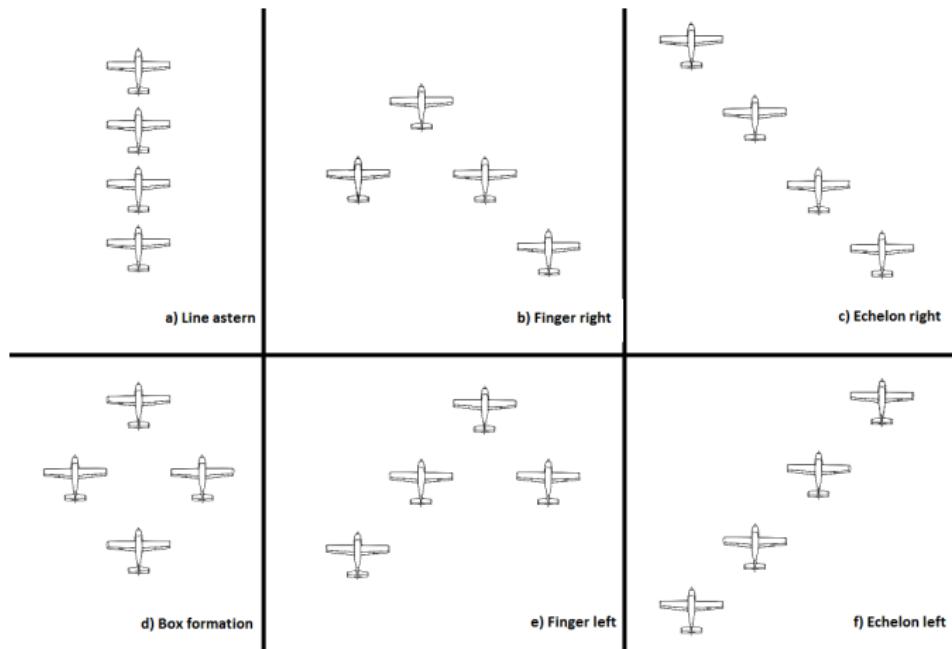


Figure 8: Formation Types

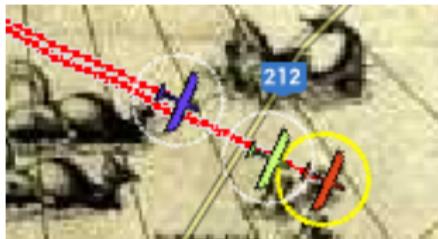


Figure 9: Top view of Line Astern



Figure 10: Top view of V Formation



## Implementation details

- ▶ follow the leader behavior
- ▶ GPS and ECEF coordinates based on WSG84 ellipsoid
- ▶ *variable matching behavior* for formation maintaining
- ▶ *follow behavior* for entering formation
- ▶ C++ with Boost library code base



- ▶ tested with dedicated leader
- ▶ leader with predefine mission
- ▶ other UAVs have a *follow the leader* behavior
- ▶ 9% to 25% of time spend for maintaining the formation
- ▶ 60% to 70% of time spend for entering the formation
- ▶ 15% to 20% of time relieving control for collision avoidance



- ▶ communication delay increases probability of breaking formation and increases efforts for formation maintaining
- ▶ 300 feet ( $< 100$  m) distance between UAVs
- ▶ for closer formations a more robust coordinate system is needed
- ▶ computational errors are induced by the ellipsoid model while converting coordinates



## Evaluation and Results

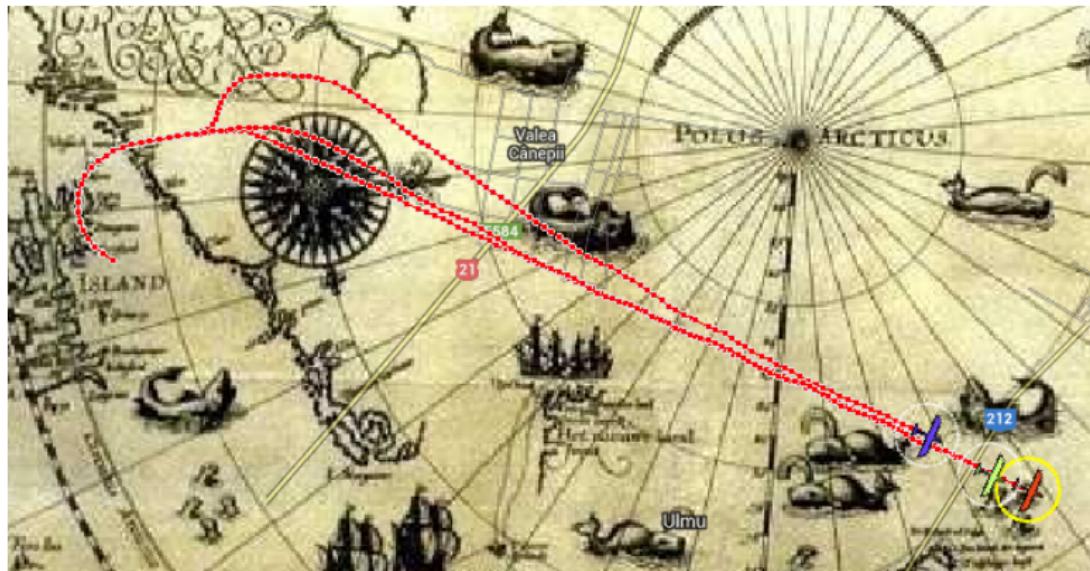


Figure 11: QGroundControl flight path for Line Astern simulation



## Conclusions:

- ▶ *Line Astern and V Formation*
- ▶ reactive software agent for formation flight
- ▶ decentralized communication

## Future Work:

- ▶ formations based on a virtual leader (geometrical center of formation)
- ▶ simulating with Flight Gear instances running on dedicated machines
- ▶ communication between Raspberry PI (holding AI software agents) and Hirrus via CAN bus
- ▶ PID controllers for speed and steering



## References

- ▶ [1] <http://aerosdb.com/uav-drone/>
- ▶ [2] <http://aft.ro/bro.pdf>



# Q&A