octoDrone: Developing Heterogenous Drone Platforms

{ Alex Henson, Ben De Ivey, Jon Gibson, and William Seymour } Computer Science, University of Warwick

Objectives

The aim of the project was to create a platform for the simulation and deployment of drone-based sensor networks. We aimed to develop a generic solution to the problem of modelling airborne sensor networks, including those comprised of heterogeneous hardware.

By establishing a framework for implementing physical deployment without the requirement of changing program code, we hope that this solution will provide a basis for teaching and research into the way that these networks are used in the future.

Drone Networks

Autonomous drones present a powerful way of extending traditional sensor networks, and are currently available at low cost for recreational and commercial use. By allowing for sampling of the environment at any point in 3D space, quadcopters can be more efficient and reduce costs compared to existing systems.

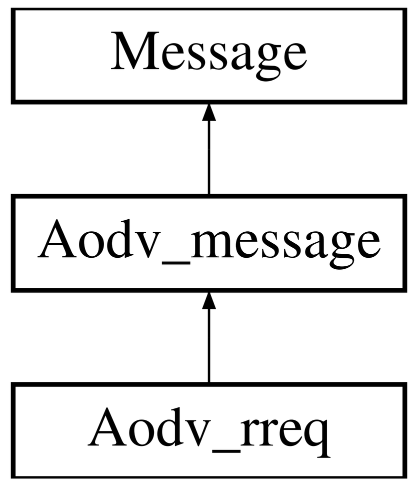
A Parrot AR 2.0 quadcopter over the Nevada desert [1]

Current popular uses for drone networks being researched include mapping, surveying, and delivery. Being able to quickly sample properties over a large area allows for more innovative and unusual uses, such as monitoring crops, managing parking spaces, and inspecting installations in dangerous conditions [2].

Simulation

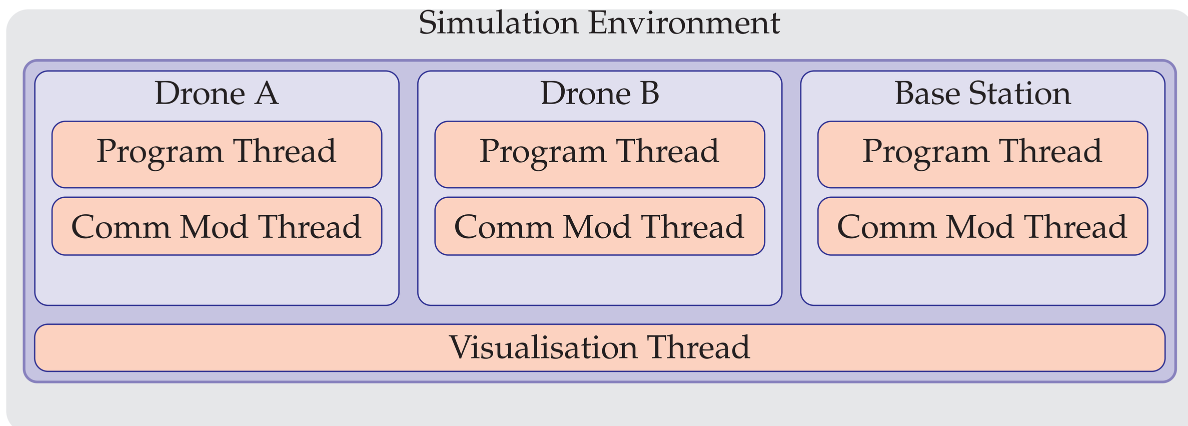
Our research found that network simulators used for research often become difficult to use after years of continual development. The most popular product, ns3, suffers from a steep learning curve and feature bloat. Thus, we designed our own simulator, octoDrone, to be accessible, extensible, and lightweight.

Accessibility was achieved by exposing a limited set of API commands which still allowed for the creation of powerful programs. More complicated actions, like routing messages between drones, are handled by separate modules to promote simplicity whilst still allowing for low level control.



In this way, developers can extend existing classes to create their own application specific instances. The AODV RREQ class is a good example of this, extending the AODV message class which itself inherits from a basic message class.

Each drone has separate threads representing its control and communication code. On top of this, the simulation and environment and visualisation layer have additional threads.



Communications modules allow for interrupt driven and time driven messaging. Inbound and outbound message queues are used to keep track of messages waiting to be processed, and the actual passing of messages (after processing) is handled by the simulation environment.

Deployment

Mobile Ad-hoc Networks (MANETs) have several distinguishing characteristics which must be accounted for when designing and evaluating routing algorithms[1]:

1. Dynamic topologies
2. Bandwidth-constrained, variable capacity links
3. Energy-constrained operation
4. Limited physical security

All electromagnetic signals dissipate as they propagate through space, limiting the range at which communications can be received. Under ideal conditions this can be modelled using the Friis free space equation:

$$P_{rx} = P_{tx} G_{tx} G_{rx} \frac{\lambda^2}{4\pi D} \tag{1}$$

Where P is the power of the radio signal, G the gain (or efficiency) of the antenna and D the distance between them. Under real world conditions additional path loss can be modelled using the log distance path loss model:

$$L = 10n\log_{10} d + C \tag{2}$$

Multiple algorithms exist to route packets within a network, which are usually classified into topological and positional. For mobile networks, topological models are ineffective due to the aforementioned dynamic topologies employed.

Ad hoc On-Demand Distance Vector (AODV) routing is one of the most popular reactive protocols used in MANETS[2]. When a node wishes to send a message but does not have a route to the destination, it broadcasts a route request (RREQ) packet to it's neighbours. Each neighbour either returns a route if it has one (RREP) or rebroadcasts the RREQ to it's own neighbours. Sequence numbers are used to eliminate suboptimal routes and by extension to prevent cyclical routes.
If the network topology has changed and a route is invalid, a node returns a route error (RERR) message. This causes other nodes to flush the parts of their routing tables which have become invalid. AODV does not have a security component, but this can be added separately. As all nodes in the network are known beforehand, keys can be distributed prior to deployment.

For MANETS, reactive protocols like AODV are preferred to proactive protocols (such as OLSR) due to low power and processing requirements.

Potential Uses

References

[1] Parrot SA, 2009. Promotional Materials. [online] Available at <http://ardrone2.parrot.com/ardrone-2/specifications/>.
[2] Mottola, L., et al., 2014. Team-level programming of drone sensor networks. 12th ACM Conference on Embedded Network Sensor Systems.
[2] Mottola, L., et al., 2014. Team-level programming of drone sensor networks. 12th ACM Conference on Embedded Network Sensor Systems.
[2] Mottola, L., et al., 2014. Team-level programming of drone sensor networks. 12th ACM Conference on Embedded Network Sensor Systems.