Drone Mounted Sensor Networks

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

Introduction

A great amount of work has gone into research surrounding sensor networks in recent years, and the explosion in popularity of the internet of things (IoT) means there is a great deal of support from both academia and industry for development in the field.
The use of quadcopters (and drones in general) has also increased in both the civilian and military sectors. By combining the power of distributed sensor networks with the manoeuvrability of unmanned drones it is possible to quickly survey and map properties over a large area.

Simulation Tools

NS3 is a "discrete-event network simulator" that allows for a network of nodes to be setup using a complete simulation of network stacks and communication. This is done using a number of helper classes for creating variations of specific setups, which can then be visualised in a separate program called NetAnim. However, discussion began on the benefits of using NS3 and dealing with the overhead of doing so, or pursuing an alternative method.

Using NS3 has proven to be more time consuming than originally expected. Due to this, the network simulation will now be built from the ground up, allowing the abstraction or inclusion of features and elements as is appropriate. This will be done in C++, for the following reasons:

- Ability to run the code natively on the drones, something not possible with NS3.
- Custom network stacks can be easily constructed
- Low level and efficient

Project Management

Weekly group meetings and an agile development methodology are used to manage project progress. Project material is handled by a git repository.

Objectives

This project focuses on methods by which such networks can be controlled and tasked. We are working to create a framework by which autonomous flying vehicles can be tasked remotely to map sensor properties over predefined areas. This encompasses:

1. Routing of drones through shared airspace
2. Routing of communications
3. Specification of tasking language
4. Fault tolerance of the network
5. Division and dissemination of instructions

Physical Routing

Pathfinding for aerial vehicles is often easier than their ground based counterparts given the relative lack of obstacles. Given that the scope of the project extends only to pre-defined areas, problems with pylons and buildings can be avoided by providing coordinates of these fixtures in advance of deployment.
In order to have drones avoid collisions with other drones, the following algorithm is proposed:

1) Each node begins with a pass counter $p = 0$.

2) When two nodes i and j identify themselves as being within a predefined collision radius they compare p_i and p_j.

3) If $p_i < p_j$ then node i waits for p_j to pass, or vice versa.

4) A passing drone sets $p = 0$ and a passed drone sets $p = p + 1$.

5) A drone undertaking a high priority task sets $p = \infty$.

Communications Routing

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:

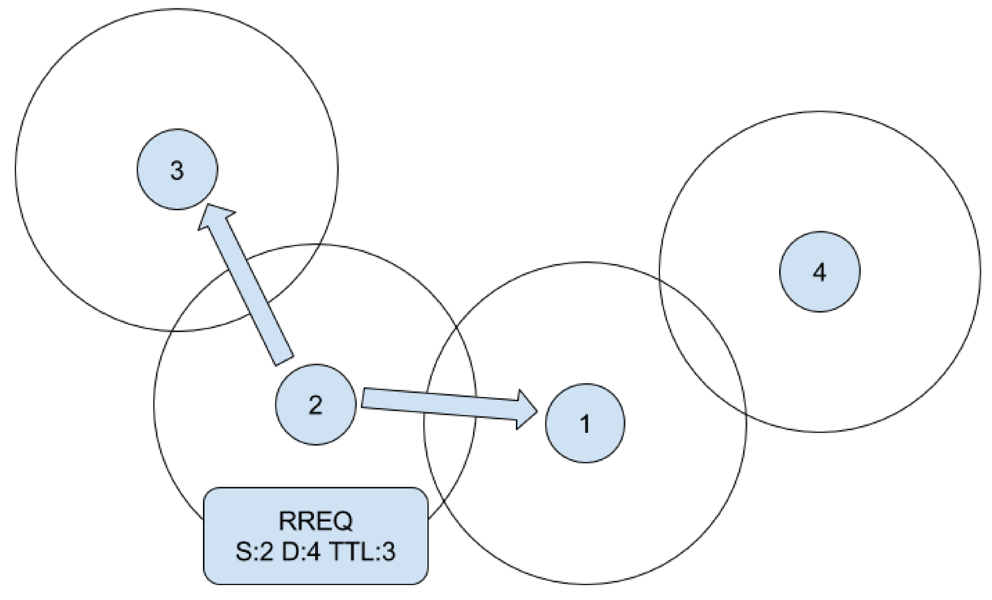
$$P_{rx} = P_{tx} G_{tx} G_{rx} \frac{\lambda^2}{4\pi D} \tag{1}$$

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.

Planned Work

1. Simulation of (scalable) drone network
2. Implement autonomous movement
3. Design and implement the base station
4. Implement communications routing
5. Collision detection and physical routing
6. Problem detection and node exclusion
7. User tasking input and execution

References

[1] J. Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations. 1999.

[2] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (aodv) routing. Technical report, 2003.

[3] A.E Abdallah, T. Fevens, and J. Opatrny. Hybrid position-based 3d routing algorithms with partial flooding. In *Electrical and Computer Engineering, 2006. CCECE'06. Canadian Conference on*, pages 227–230. IEEE, 2006.