

Mobile Ad-hoc Networks and Wireless Sensor Networks

Magnus Jensen

5. Mobile Ad-hoc Networks and Wireless Sensor Networks

- Mobile Ad-hoc Networks and Wireless Sensor Networks
 - Motivation
 - Ad-Hoc networking
 - MANET routing
 - Goals
 - Pro vs Reactive
 - Protocols
 - DSDV
 - AODV
 - DSR
 - Energy Efficient
 - Wireless Sensor Networks
 - Routing
 - Data aggregation
 - Data centric routing

~~Using link state each peer periodically sender deres naboer, deres pris for linket, deres ID og sådan, og så ved man altid hvordan man router til den billigste. Det er dyrt at gøre sådan, kræver en del trafik~~

~~Der er også noget der hedder en distance vector, tror det er noget med at noder deler deres priser for at route til andre steder; og så kan en node via andre noder finde den billigste ting.~~

1. Motivation

2. Ad-Hoc networking

Ad hoc networking is the idea of enabling radio connected devices to collaborate with each other without having to rely on a centralised station.

Doing such thing, encounters a number of challenges.

- Problems

- limited devices these devices are often limited in range, bandwidth, cpu, storage or battery life - making it necessary to take such limitations into account

- portable well guess what, portable devices are .. portable!
They move around! And that is also important to think about

Also being that it typically is radio waves which is used, that itself brings some problems, where signal strength can be unpredictable, radio waves can be blocked or absorbed, radio communication can be unidirectional where a can hear b, but b cant hear a, and broadcasting in itself can be a security risk.

It is clear that working with these technologies brings a lot of problems with it, but what they enable can be used for remote areas, unplanned meetings, to extend existing infrastructure, emergency situations, military, sensor networks and so on.

3. MANET routing

Okay so routing in a mobile adhoc network.. that sound hard..

3.1. Goals

For mobile adhoc network protocols, we have some goals which they should adhere to.

- **Goals**
- minimal overhead being that it is mobile and often weak devices, we have finite power resources and protocols should respect that.
- minimal processing overhead this is also why processing should be kept at a minimum, since processing takes power and power is limited and cpu typically also are. Simple routes are therefor great, where complex graphs dont need to be calculated.
- Multi hop routing often we could end in a situation where need send something to a peer, but we cant communicate directly with it, so we need to do multiple hops to get there.
- dynamic topology being that mobile devices is mobile, that can move. So protocols must be flexible in the topology of the networks.
- loop prevention and lastly, having a loop really just is a waste of time, why the protocol should prevent this.

3.2. Pro vs Reactive

There are two ways to route in a MANET, where the routing either can be proactive or reactive.

- **Proactive** Routing table In a proactive routing environment, each node knows the network which is laid out in a routing table. Route changes therefore, some bandwidth and energy will also be used to maintain route tables when they change, like if a node is moved. Low latency but that it does know the difference routes, makes the proactive approach have low latency. Control overhead but it is at a cost of high control overhead by maintaining and keep updating the routing table.
- **Reactive**
- (almost) No state each node contains no state, or close to, meaning that they only keep what is necessary.
- route on demand therefore, routes are discovered on demand, when a message is needed to be sent, but is remembered though. If a remembered route is broken, it needs to discover a new again on demand.
- routing = costly when suddenly a routing is needed, it is more costly though, because energy is needed for discovery.

3.3. Protocols

Great to let's look at a few different protocols!

3.3.1. DSDV

Destination Sequence Distance Vector. Is a great name, am I right? It is the first protocol we are gonna look at though, and it is proactive!

- **DSDV**
- Proactive
- Routing table being that it is proactive, it of course has a routing table, where it keeps track of:
 - * Destination
 - * Next Hop
 - * Cost
 - * sequence number and a sequence number. Whenever a new route is received, it will be stored in the routing table if it has a higher sequence number; if it equal though, it will keep the one with the lowest cost.
- damped updates When ever a node moves, it waits a bit

before telling about it, until it is settled; so we don't get updates of a ton of routes which will replace each other. A node will infrequently send a full update, but whenever a change is found in the routes, if it can send that information as a piggy bag ride in a single network package - it does so.

- **loop prevention** To combat loops, the sequence number is an important aspect of the protocol. Each node will keep track of its own number, and whenever it sends out an update of its route, it bumps the number by two. If a node finds that the route suddenly is broken, it will send out an update with a cost of ∞ and a sequence number bumped by one.

3.3.2. AODV

Adhoc On-demand Distance Vector is another protocol, which is reactive this time.

- **AODV**
- Reactive
- **no routing table** and there has no routing table!
- **route request** when a route is needed, a route request is broadcasted into the network, which each other node will propagate
- **route response** when the request comes to the destination or a node that knows it, a *route response* is sent back using the reverse path. It is helped by every node which remembers a broadcast ID from the request, which now will begin to remember to path.
- **destination sequence number** such a response contains a sequence number, so if multiple responses come back, the peer will use the route with the highest number

In AODV link failures are handled by also marking it at a ∞ cost, propagated to a peer's neighbors which uses the route. Now whenever a route to the node is needed again, it will be discovered using a higher sequence number.

- properties
 - cheaper only maintained active routes
 - small, simple table lookups, comparing numbers etc
 - functions well in low traffic

- in high traffic, the initial cost can be high

3.3.3. DSR

And just for fun, lets get a quick overview of yet another reactive protocol, called Dynamic Source Routing.

- **DSR**
- Reactive
- request and replies it too uses request and replies, but in a very different way, where the intermediate nodes wont remember anything from it
- Request builds path when the request is broadcaster, it is modified to keep the path in itself. Meaning that it start by containing A, then AB, then ABC and such
- replies if the network is bidirectional, the reply can simply go through the reverse path, but is it unidirectional, the destination can simple send a request against the source, and in that message let the request-path piggyback to prevent loop.
- radio geared Here it is shown that it truly is a radio-geared protocol because it supports being unidirectional. Also it can con
- ~~if memory is available, it can store the routes now if it has multiple to the same, if one breaks, it has a backup~~

3.4. Energy Efficient

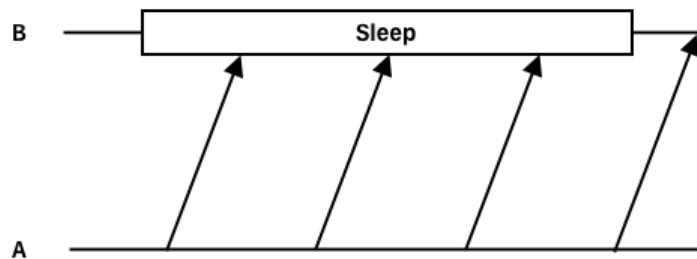
Lets briefly discuss how MONATs could be more energy efficient.

We could implement some power control, where we adjust what the cost metric is measured on, to measure the power consumption instead of f.x. hops. Therefor, we could end up in routes that are longer, but will use less power to transmit. It could also be, that these routes allow for the transmission range to be downgraded again to control the power usage.

- **Power Control**
- longer, cheaper paths
- Lower transmission range

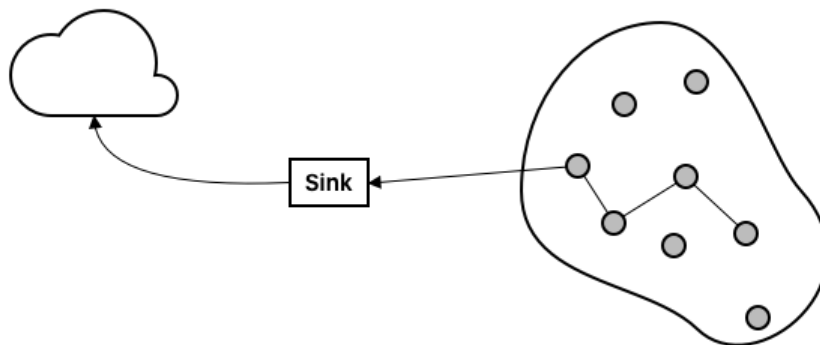
Or we could go ahead simply and save some of the power. There is no doubt that transmission is the priciest state of the node and that sleeping is the cheapest. So why dont we just sleep? Well how will we communicate?

- **Power save**
- Sleep
- **Clocks** each node could have a precise clock, and wake up a specific times to communicate, but this is unpractical because the clocks.
- **Retransmission** So lets instead use some retransmission: if we know how long a node is supposed to sleep, so if we retransmit a message several time, we can be sure to hit a window where it does not sleep



4. Wireless Sensor Networks

A wireless sensor network, is a special class of MANETs where the energy efficiency is very important.



It is a field of sensor, made up of many small, cheap and limited nodes. Their only purpose is to record data and route it towards a component outside of the field, called a sink; which will handle the distribution of the data to the internet.

- **WSN**
- Node are cheap it is a very important, since the nodes init them selves dont mean much
- Survival of network instead it is the survival of the network which is the important part

4.1. Routing

Now, all this routing where the transmission will occur will become costly, so its need to be node right. Lets go trough four strategies, which each are all possible to use and is field specific.

- **Maximum PA route**
- Route with most PA the first strategy could be to use the route with the most power available. This could be so nodes almost out of energy wont be used.
- **Least energy**
- minimum cost here we would prefer the route which have the lowest total transmission cost.
- **Least hops**
- Shortest path taking the shortest path could also be a viable concerned option, because it is really easy to compute
- **maxmin PA** largest of smallest PA at last we could prefer the route along, which have the largest, of the smallest PA on each route.

Again, what strategy to use, is case specific.

4.2. Data aggregation

Imagine if each recording had to be sent trough the network, well again - that is a lot of messages that need to be sent. So when a node receives a message from another sensor, it could wait a while to collect several more measurements as long as it would be able to fit within a single package, and combine them all. Thereby, we minimise the amount of packages sent.

- **Data**
- Combine

But we could also use other strategies, like if a node retrieves temperatures from its neighbours, instead of sending them all, the network could also be designed to work on averages, why the node only would send a single average measurement towards the sink.

- average

4.3. Data centric routing

- Sink send queries

In ordinary MANET, we might request resource held by specific node, but in a wireless sensor network, it is also possible for the sink to request data from nodes which measures some specific data, like temperatures higher than 20 degrees. This could possible lower transmissions a lot.

- Nodes announce new data

But it can also be, that nodes instead of just sending the new data constantly, the advertise that they have new data, and if interested they can be asked to deliver it.

So yeah, we have seen now, that in the field of MANETS there are a lot of challenges.