Propagation Models in VANET research
e.m.vaneenennaam@utwente.nl

University of Twente - Design and Analysis of Communication Systems
(DACS) - Faculty EEMCS
Communicating vehicles, wouldn’t that be cool?

It’s called a VANET!
• “Some ITS applications require over-the-horizon awareness”
• Nice thing about traffic engineers: they know traffic
• VANET research is multidisciplinary;
• so I come up with a means to supply the over-the-horizon awareness:

Multi-hop communication by means of a slotted flooding mechanism
(and some improvements along the way)
• Idea
• Formalisation $\Rightarrow$ Protocol
  - Does it work as I think?
    - Does it work well?
    - Does it meet the application’s requirements?
• Three options:
  - Build 1200 cars and see if it works
  - Derive a detailed analytical model
  - or... simulate!
Imagine... your own parameterised world where things happen based on what you want and with a controllable amount of randomness...
Steps in a simulator study:

1. Implement protocol (C, C++, Java, protocol description language)
2. Define test scenarios (how many nodes, topology, mobility)
3. Run simulator (may take a while)
4. Evaluate results (is it any good?)
Basically, you are building a system with:

- Node Mobility
- Radio Wave Propagation
- Hardware (PHY, modulation)
- Behaviour Logic (a protocol)

To see how your protocol would work in the *real world*

But these are all MODELS!!1

(ok so they should reflect reality pretty well)
Simulating VANETs:

- Focus usually is on protocols (re-use existing model in simulator)
- Aim is to reduce complexity, increase runtime efficiency
- Different timescales: Traffic in seconds or minutes, propagation at much finer scale
- Propagation is often abstracted from

Propagation model is “necessary evil”
Dependencies

Mobility determines:

- Node topology (distribution)
- Node density (and local fluctuations in space and time)
- Contact moments (how long within range)

Propagation determines:

- + Reachability (which nodes can communicate)
- - Interference (which nodes interfere)

Propagation + Mobility $\rightsquigarrow$ dynamic sets of nodes sharing medium
So... the flooding algorithm is basically a Medium Access scheme!

- ... based on who can communicate
- limited by who can interfere

this determines the number of nodes for which we need to mediate access to the medium.

PROPAGATION MATTERS!
Propagation in a Simulator (OMNeT++ MFw)

1. Based on pos(x,y) ChannelControl determines nodes involved
   - → All nodes for which $P_r > \text{noise} \rightarrow \text{interference distance}$

2. Determine $\frac{\text{signal}}{\text{noise} + \text{interference}}$: SINR

3. Based on PHY, modulation and BER decide on success
Propagation Models

From implementation p.o.v.:

**Deterministic** - Based on simulated real-world state (e.g. distance between T and R)

**Probabilistic** - Take a deterministic model and add an appropriate degree of randomness to model the uncontrollable chaos in real world
Deterministic Models

- Nodes in Free Space: Free Space model
- Nodes on a plane: Two-ray Ground model
- Nodes in a 3D environment: Ray tracing models

Increasingly complex, both to model correctly and to compute
Probabilistic Models

Provide an abstraction to the simulated real-world problem with a \textit{mean} and a \textit{deviation}:

\[ P_r(d) = P_{r_{\text{det}}}(d) + \chi(d, \sigma) \]
Examples:

- Log-Normal shadowing
- Rayleigh
- Longley-Rice
- Nakagami

Nakagami:

\[ P_r(d; m) \sim \text{Gamma} \left( m, \frac{P_{r_{det}}(d)}{m} \right) \]

Generic:

\[ m = 1 \sim \text{Rayleigh}, \lim_{m \to \infty} \sim \text{Free Space} \]
Why researchers like simple models:

- I want to compare 2 protocol options
- ... 10 different traffic conditions $\in \{100 \ldots 1250\}$ nodes
- for statistical significance, 50 runs
- one run is 300s $\rightarrow$ 100 floods
- one flood $\sim$ 50 hops

1 000 simulation runs, 100 000 floods, 5 000 000 hops, per hop propagation model is eval’d for ALL relevant nodes to determine success or failure.

(my simulation took approx. 60 hours)
Steps to decide on correct reception:

1. Calculate received signal strength $P_r(d)$
2. then, \texttt{foreach interferer i add i.P_r(d) to noise}
3. based on modulation model, SINR and BER threshold, \texttt{eval packetOk()}

So function $P_r(d)$ called often, hence this is the first place to optimise for runtime speed.

And that is why most VANET research does not use very elaborate propagation models.
Conclusions

- Propagation model has large influence on results, impacts:
  - Which nodes are able to communicate
  - The probability of correct reception
  - Thus the speed at which messages propagate
  - The overhead (collisions, medium utilisation)
  - and end-to-end delay

- So the real-world implementation could behave different from the simulation

- So carefully map model and parameters to the target environment
Then, which propagation models are used?

- Free Space, TRG – standard in most sims
- Log-Normal shadowing
- Nakagami – used a lot and “reasonably realistic”
- Trade-off between runtime speed and accuracy
- Allow statistic evaluation: “on average, it performs well”
A few remarks on present-day simulation tools:

- Vehicles themselves are never considered in propagation models – this must have an impact if we design a system which should operate both in scarce traffic and traffic jams.
- The propagation environment is assumed to be homogenous – although it seems reasonable propagation properties are not in the real world!

Should we model VANET simulators to include more accurate propagation models?

(i.e. consider vehicles as part of the propagation environment, is it worth the effort?)
Questions?
Backup Slides

– intentionally left empty –