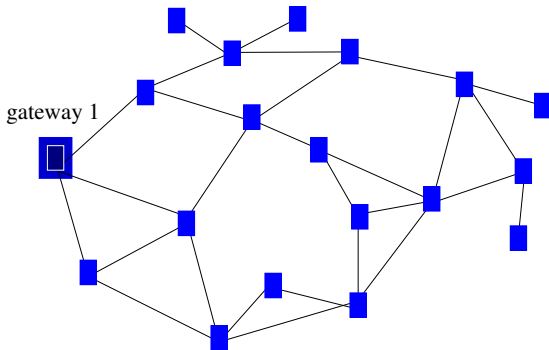


Exploring Semantic Interference in Heterogeneous Sensor Networks

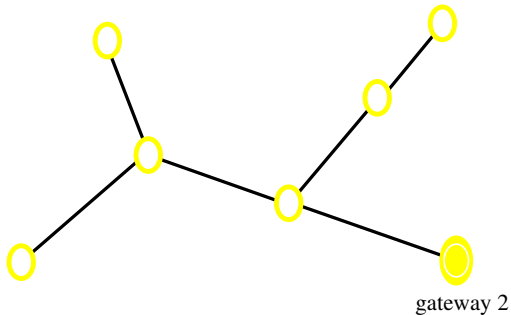
Laura Marie Feeney
Communication Networks and Systems Laboratory
Swedish Institute of Computer Science

27 May 2008

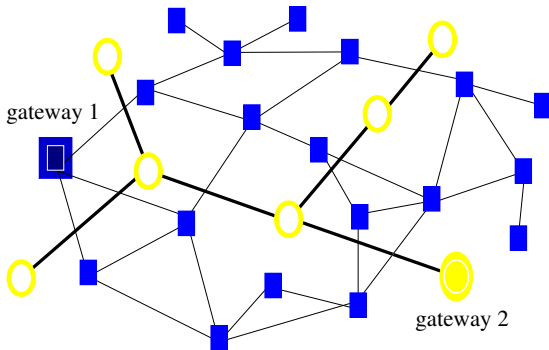
- what happens when multiple **independent** sensor networks operate in close proximity?
- frames transmitted in one network can be decoded by nodes in the other network → “semantic interference”
- two aspects of semantic interference
 - ▶ basic ways to mitigate problem
 - ▶ opportunities for cooperative architecture



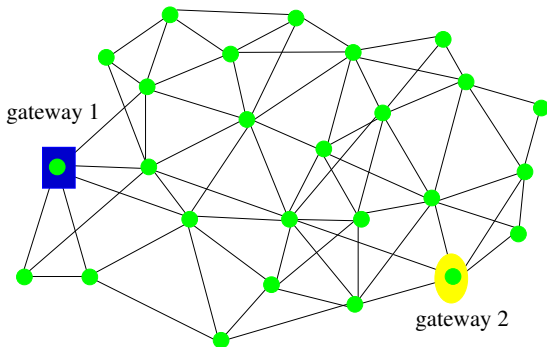
blue network is deployed to control HVAC



yellow network is deployed for building security



networks are co-located, but managed independently



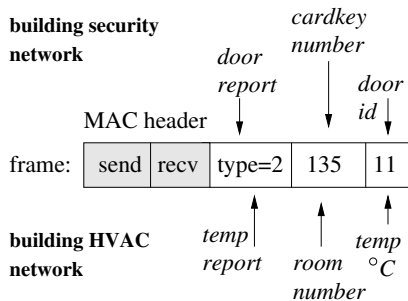
common radio creates a larger, denser network

MAC header

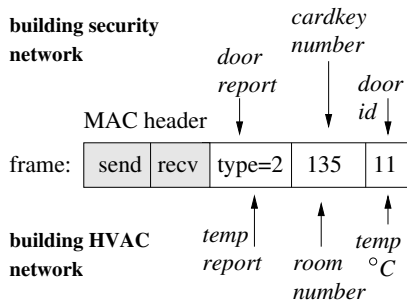
frame:

send	recv	type=2	135	11
------	------	--------	-----	----

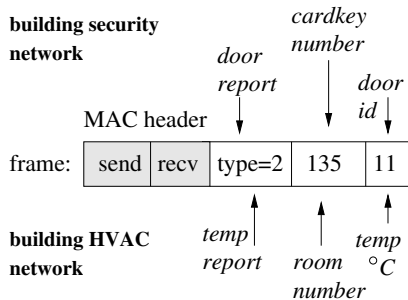
ESB node (TR 1001 transceiver) + Contiki(x-mac)



frame for building security network



door 11 is opened \iff room 135 is 11°C



mis-interpretation is called **semantic interference**

- independent sensor networks in close proximity
 - ▶ potentially **many** networks (e.g. railway terminal)
- no one will have knowledge or control over all networks in a location
 - ▶ variety of applications and users
 - ▶ set of co-located networks is **dynamic**
- different providers use common radio technology
 - ▶ complete system (e.g. tMote Sky + TinyOS + ?)
 - ▶ semi-custom hardware and commercial radio (e.g. RFM TR1001)

- **semantic interference**: a frame is mis-interpreted by a receiver in a co-located network
 - ▶ contrast with radio interference and contention
- already seen a (silly) example
- more likely, frame is eventually discarded
 - ▶ processing consumes resources, may affect future operation
 - ▶ receiver may infer sensor or network failure
 - ▶ receiver software may crash
- special case of wakeup radio → **wakeup interference**

- problem is obvious, but has not been systematically addressed
- some isolation mechanism is **required** to ensure safe co-existence
 - ▶ **otherwise your network will break**
- identify and filter out foreign frames
- can be done at various layers, or using signatures
- the higher the layer, the stronger the assumption that frame has a known format at that layer

- PHY layer
 - ▶ separate channels good for many reasons
 - ▶ not enough channels, dynamic environment
- MAC layer
 - ▶ **requires** identifiable MAC format
 - ▶ network ID in MAC header (e.g. IEEE 802.15.4)
- network and higher layers
 - ▶ **requires** identifiable higher layer frame
 - ▶ TinyOS Networks Service Manager
 - ▶ TCP/IP vendor-specific registered ports

- signatures
 - ▶ each network has shared network key
 - ▶ attached signature to all frames
 - ▶ appropriate signature?
- strong cryptographic signature is very effective
 - ▶ strong authentication/integrity is costly
 - ▶ hardware support only in larger devices (IEEE 802.15.4)
 - ▶ smaller devices in heterogenous networks

- wakeup radio
 - ▶ primary (data) radio sleeps
 - ▶ wakeup radio always listens on wakeup channel
 - ▶ transmission preceded by announcement on wakeup channel
- wakeup radio must be very low power (low data rate)
- same wakeup radio
 - ▶ announcements shared across all networks - false wakeups
- ...and different primary radio
 - ▶ no good communication channel for resolving conflict
- example
 - ▶ IEEE 802.15.4 and 6LowPAN (short address)
- overhead cost to reduce risk of collision

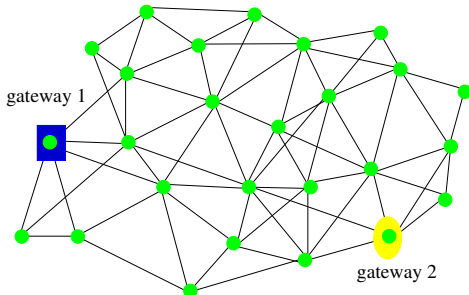
- tradeoff between isolation and overhead
 - ▶ some isolation mechanism is necessary
 - ▶ isolation is not likely to be cheap
- doubt performance results based on very minimal systems
 - ▶ small devices and minimal protocols
 - ▶ isolation overhead may be significant

- defensive programming
 - ▶ application logic at each node usually simple
 - ▶ important to minimize memory usage → minimize code
 - ▶ tempting to not include various checks
- check for mal-formed frames and implausible payload data
 - ▶ especially if not strongly isolated
 - ▶ don't over-react(?)

- detecting foreign traffic
 - ▶ several co-located networks → most traffic will be foreign
 - ▶ many frames won't pass signature check
 - ▶ corrupted, (legitimate) foreign, and hostile traffic
- don't over-react to frames with invalid signature

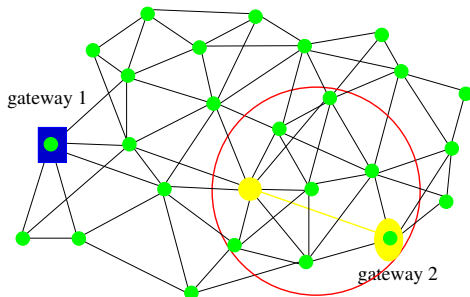
- dimensioning sensor networks
 - ▶ estimate network lifetime based on expected activity in own network (e.g. duty cycle, frames/hour, events/day)
 - ▶ relatively straightforward
- hard to predict existence and behavior of foreign networks
 - ▶ cost of filtering foreign frames is non-trivial (esp. crypto)
 - ▶ cost of false radio wakeups is very high
- harder to dimension network

- isn't this just sensor network security?
 - ▶ yes and no...
- obviously, cryptographic isolation to filter traffic from foreign network(s)
- attack model based on physical limitations...
 - ▶ attacker can't attack/compromise nodes everywhere, all the time, without energy limitation
 - ▶ probabilistic detection/avoidance
- but foreign network isn't an attacker...
 - ▶ legitimate presence
- cooperation is possible



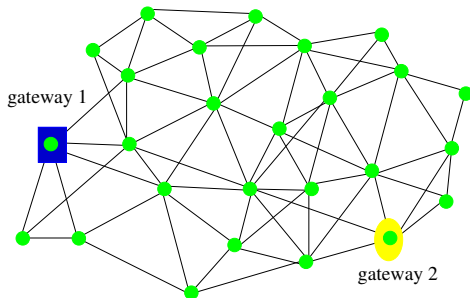
accept that this is the real network

- benefits of allowing networks to cooperate in dynamic environment



shared routing fabric

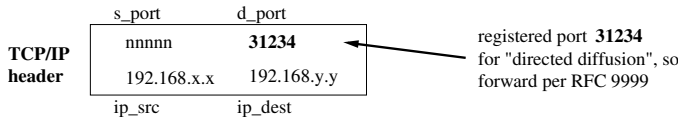
- higher density, shorter transmit range, path diversity, load balancing



joint prioritization of access to shared channel

- peak usage in emergency management networks

- cooperation mechanisms
 - ▶ require some way to detect what cooperation mechanism is used
 - ▶ can still detect that frame is foreign (?)
 - ▶ **not** require knowledge of what networks are operating
- some speculation about possibilities
 - ▶ Internet model
 - ▶ shared runtime environment
 - ▶ virtualization



- Internet/IETF model for coexistence in shared communication fabric
 - ▶ TCP/IP supported via μ IP or 6LowPAN
 - ▶ registered TCP/UDP port numbers allow multiple application endpoints
 - ▶ node can provide support for any standardized protocol
 - ▶ application layer forwarding, not sensor data fusion

- independently implemented applications all use basic functionality provided by common run-time environment
- run-time environment responsible for efficient per-application operation
 - ▶ e.g. applications register themselves with run-time, semi-centralized coordination among (shared) gateways
- potential separation of sensor infrastructure and applications
 - ▶ many applications are installed onto a shared infrastructure

- widely used to support divergent functionality on shared systems
- virtual machine architecture for sensor nodes
- deploy application specific data processing code
- powerful solution
 - ▶ only the most capable devices (SunSpot)
 - ▶ hard problems even on very capable devices

- explored and defined problem of 'semantic interference'
 - ▶ independent, co-located sensor networks
 - ▶ frames transmitted in one network are mis-interpreted in the other
 - ▶ somewhat different from general security problem
- avoiding semantic interference
 - ▶ lightweight signatures
 - ▶ practical tips
- using semantic interference
 - ▶ three cooperative models – lots of future work!