Wireless Sensor Networks
Chapter 7: Naming & Addressing

António Grilo

Courtesy: Holger Karl, UPB
Goals of this chapter

• This short chapter looks at non-standard options for denoting the senders/receivers of messages
  • Traditional (fixed, wireless, ad hoc): Denote individual nodes by their identity
  • WSN: Content-based addresses can be a good complement

• When addresses are not given a priori, they have to be determined “in the field”
  • Some algorithms are discussed
Names vs. addresses

**Name:** Denote/refer to “things”
- Nodes, networks, data, transactions, …
- Often, but not always, unique (globally, network-wide, locally)
- Ad hoc: nodes – WSN: Data!

**Addresses:** Information needed to *find* these things
- Street address, IP address, MAC address
- Often, but not always, unique (globally, network-wide, locally)
- Addresses often hierarchical, because of their intended use in, e.g., routing protocols

**Services to map between names and addresses**
- E.g., DNS

**Sometimes, same data serves as name and address**
- IP addresses are prominent examples
Issues in address management

- Address allocation: Assign an entity an address from a given pool of possible addresses
  - Distributed address assignment (centralized like DHCP does not scale)
- Address deallocation: Once address no longer used, put it back into the address pool
  - Because of limited pool size
  - Graceful or abrupt, depending on node actions
- Address representation
  - Format for representing addresses must be negotiated and implemented
- Conflict detection & resolution (*Duplicate Address Detection*)
  - What to do when the same address is assigned multiple times?
  - Can happen e.g. when two networks merge
    - Unnecessary Deallocation/Reallocation should be avoided
- Binding
  - Map between addresses used by different protocol layers
  - E.g., IP addresses are bound to MAC address by ARP
Why not globally unique addresses?

- Globally unique addresses significantly simplify address management.
- Must be judged relative to the impact on overhead.
- E.g., Ethernet 48-bit MAC address:
  - 500-octet frame (e.g., IP): overhead of 1.2%
  - 4-octet frame (e.g., WSN): overhead of 150%
- MAC addresses only need to be unique within 2-hop neighborhood.
Distributed assignment of networkwide addresses

- **Option 1:** Let every node randomly pick an address
  - For given size of address space, unacceptable high risk of duplicate addresses

- **Option 2:** Avoid addresses used in local neighborhood

- **Option 3:** Repair any observed conflicts
  - Temporarily pick a random address from a dedicated pool and a proposed fixed address
  - Send an *address request* to the proposed address, using temporary address
  - If *address reply* arrives, proposed address already exists
  - Collisions in temporary address unlikely, as only used briefly

- **Option 4:** Similar to 3, but use a neighbor that already has a fixed address to perform requests
Duplicate Address Detection (DAD)

- In distributed address assignment networkwide uniqueness may not be granted at all times

- **Strong DAD**
  - If address $x$ is assigned to A at time $t_0$ and to B at time $t_1$, the conflict must be detected within $t_1 + T$

- **Weak DAD**
  - Duplicate addresses are tolerated as long as they do not distort ongoing sessions.
Distributed assignment of locally unique addresses

- Schurgers et al [739]
- E.g., Assign local addresses to A and B

![](image)

- Requirements:
  1. Nodes know bidirectional and inbound neighbors
  2. Nodes communicate only with bidirectional neighbors
  3. Addresses of bidirectional neighbors of a node must be different than that node’s address
  4. Address of inbound neighbors must be different from the address of all bidirectional neighbors
Distributed assignment of locally unique addresses (2)

- **Address assignment algorithm**
  1. A performs neighbor discovery
  2. A broadcasts HELLO packet
  3. Listen for INFO with time bound
  4. Bidir. and outbound neighbors respond with INFO (globally/networkwide unique addr., MAC address, MAC addrs. of bidir. neighbors)
  5. A can only receive INFO from bidir. Neighbors
  6. One of two things now occurs:
     - If all of A’s 1-hop neighbors have different addresses, A just chooses different, unique address within 2-hop neighborhood
     - If there is conflict between A’s 1-hop neighbors, A issues CONFLICT, triggering new address selection round
Address Selection and Representation

- Schurgers et al [734, 739]
- Address selection is greedy (free lowest address selected first) instead of random
- Non-uniform address distribution (less than maximum entropy)
- This can be exploited to minimize the average length the address
  - Huffman coding optimal, within 1 bit of the entropy $H(X)$
- If we can know the MAC address distribution, Huffman algorithm gives optimum codebook
  - Variable length codewords
Address Selection and Representation (2)

- Algorithm is enhanced if selected address is chosen randomly among those that correspond to codewords with the same length as lowest address value.

  > Probability of conflict decreases, and hence the number of CONFLIT packets also decreases
Address Selection and Representation (3)

- Fixed length vs variable length schemes:
  - For the same network size, variable-length → lower average length
  - Average length converges as $N \to \infty$
  - Variable length pays-off, but exact turning point depends on the density

- Average size of variable scheme depends on the node density
Other address selection schemes

- Although centralized address assignment does not scale, it eliminates conflict → Hybrid schemes
  - Clustering approach
    - Clusterheads have locally unique addresses
    - Clusterheads responsible for conflict-free assignment within clusters
Content-based addresses

- Recall: Paradigm change from id-centric to data-centric networking in WSN
- Supported by content-based names/addresses
  - Do not described involved nodes (not known anyway), but the *content* itself the interaction is about
- Classical option: Put a naming scheme on top of IP addresses
  - Done by some middleware systems
Content-based addressing: Describe *interests*

- **Interests** describe relevant data/event
  - Used, e.g., by directed diffusion
  - Nodes match these interests with their locally observed data
- **Format: Attribute-Value-Operation**
  - `<attribute, value, operation>`, e.g.: `<TEMP, 20° C, GE>`
  - Attributes: temperature, pressure, concentration, position...
- **Operations:**

<table>
<thead>
<tr>
<th>Operator name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>EQ</td>
<td>Matches if actual value is equal to value</td>
</tr>
<tr>
<td>NE</td>
<td>Matches if actual value is not equal to value</td>
</tr>
<tr>
<td>LT</td>
<td>Matches if actual value is smaller than value</td>
</tr>
<tr>
<td>GT</td>
<td>Matches if actual value is greater than value</td>
</tr>
<tr>
<td>LE</td>
<td>Matches if actual value is smaller or equal to value</td>
</tr>
<tr>
<td>GE</td>
<td>Matches if actual value is larger or equal to value</td>
</tr>
<tr>
<td>EQ_ANY</td>
<td>Matches anything, value is meaningless</td>
</tr>
<tr>
<td>IS</td>
<td>Specifies a literal attribute</td>
</tr>
</tbody>
</table>
Directed Diffusion brief overview

- Basic algorithm (explained in more detail later on)
  1. **Sink** nodes generate interests
  2. Interests are disseminated in the WSN
  3. Nodes store interest and upstream neighbors in cache
  4. Nodes whose description matches interest become **sources**
  5. Sources generate data packets
  6. Intermediate nodes match data packets against cached interest
  7. Intermediate node forward the data packets to the sink
Matching algorithm

- Check whether an interest matches the locally available data

```java
parameters: attribute sets A and B

// A corresponds to the interest, B to the data message

foreach attribute a in A where a.op is formal {
    matched = false
    foreach attribute b in B where
        a.key == b.key and b.op is actual {
            if b.val satisfies condition
                expressed by a.key and a.val then {
                    matched = true
                }
            }
        }
    if (not matched) then {
        return false
    }
}
return true; // matching successful!
```
Geographic addressing

- Express addresses by denoting physical position of nodes
  - Can be regarded as a special case of content-based addresses
  - Attributes for x and y coordinates (and maybe z)

- Options
  - Single point
  - Circle or sphere centered around given point
  - Rectangle by two corner points
  - Polygon/polytope by list of points
  - …
Conclusion

- Addresses can be assigned in a distributed way

- Non-id-centric addresses give additional expressiveness, enables new interaction patterns than only using standard addresses

- These addresses have to be supported by specific protocols, in particular, routing protocols