Self-Organization in Autonomous Sensor/Actuator Networks

Dr.-Ing. Falko Dressler

Computer Networks and Communication Systems
Department of Computer Sciences
University of Erlangen-Nürnberg

http://www7.informatik.uni-erlangen.de/~dressler/
dressler@informatik.uni-erlangen.de
Overview

- Self-Organization
  Introduction; system management and control; principles and characteristics; natural self-organization; methods and techniques

- Networking Aspects: Ad Hoc and Sensor Networks
  Ad hoc and sensor networks; self-organization in sensor networks; evaluation criteria; medium access control; ad hoc routing; data-centric networking; clustering

- Coordination and Control: Sensor and Actor Networks
  Sensor and actor networks; communication and coordination; collaboration and task allocation

- Self-Organization in Sensor and Actor Networks
  Basic methods of self-organization – revisited; evaluation criteria

- Bio-inspired Networking
  Swarm intelligence; artificial immune system; cellular signaling pathways
Basic Methods of Self-Organization – Revisited

- Positive and negative feedback
- Interactions among individuals and with the environment
- Probabilistic techniques
### Networking aspects

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC</td>
<td>positive and negative feedback for controlling the used transmission energy, e.g. in PCM; enforcement of synchronization between multiple nodes to a common schedule</td>
</tr>
<tr>
<td>Ad hoc routing</td>
<td>positive feedback for route discovery in most table-driven routing protocols; negative feedback for suppression further data messages over erroneous paths, both used e.g. in AODV and DYMO</td>
</tr>
<tr>
<td>Data-centric networking</td>
<td>positive feedback in form of interest messages controlling the behavior of sensor nodes, e.g. in directed diffusion; energy levels and timeouts as negative feedback to suppress unnecessary communication, e.g. in rumor routing</td>
</tr>
<tr>
<td>Clustering</td>
<td>feedback is provided for example in form of energy levels controlling system-inherent parameters such as the probability to become clusterhead in HEED</td>
</tr>
</tbody>
</table>
## Positive and negative feedback

### Coordination and control

<table>
<thead>
<tr>
<th>Communication and coordination</th>
<th>Feedback loops are inherently used by all time synchronization techniques; positive and negative feedback enables adaptive coordination among nodes, e.g. for optimizing the utility by ASCENT or to ensure a latency bound by DEPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaboration and task allocation</td>
<td>Positive feedback in the biddings in auction-based task allocation, e.g. in MURDOCH, and negative feedback through re-allocation; feedback based probability adaptation in the case of emergent cooperation</td>
</tr>
</tbody>
</table>
## Interactions among individuals and with the environment

### Networking aspects

<table>
<thead>
<tr>
<th>MAC</th>
<th>Intensive protocol inherent interactions between neighboring nodes to detect or prevent collisions, e.g. MACA based protocols; synchronization according to local message exchanges; indirect information exchange using signal strength measurements, e.g. in PCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ad hoc routing</td>
<td>State and topology maintenance for address-based routing; interactions among neighboring and remote nodes to search shortest path information, e.g. in AODV and DYMO; duplicate address detection based on node interactions, e.g. in PDAD and DAA</td>
</tr>
<tr>
<td>Data-centric networking</td>
<td>Optimized gossiping strategies exploiting the local topology information; agent based approaches relying on stigmergic information exchange and on local interactions between neighboring nodes, e.g. in rumor routing; adaptation of source-sink relationships according to remote interactions in directed diffusion</td>
</tr>
<tr>
<td>Clustering</td>
<td>Interaction provides the basis for clustering techniques; transmission power estimation and cluster affiliation using local interactions, e.g. in LEACH and HEED</td>
</tr>
</tbody>
</table>
## Interactions among individuals and with the environment

**Coordination and control**

<table>
<thead>
<tr>
<th>Communication and coordination</th>
<th>time synchronization based on exchanged data or specific time protocol messages (used by all the discussed algorithms); topology maintenance and clustering techniques based on local interactions, e.g. in Span, ASCENT, and DEPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaboration and task allocation</td>
<td>intentional coordination based on directed communication to a central decision taker, either for periodic state maintenance, e.g. in OAA, or for auction systems, e.g. in MURDOCH and mediation; local interactions among neighboring agents and stigmergic communication in emergent cooperation</td>
</tr>
</tbody>
</table>
### Probabilistic techniques

<table>
<thead>
<tr>
<th>Networking aspects</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MAC</strong></td>
<td>reduced collision probability through randomized medium access; fairness and mutual exclusion are achieved by using random startup delays for the RTS/CTS handshake, e.g. in all MACA based protocols</td>
</tr>
<tr>
<td><strong>Ad hoc routing</strong></td>
<td>gossiping techniques to reduce the flooding overhead in reactive routing approaches, e.g. in optimized AODV; dynamic address allocation based on stateless random address selections in combinations with DAD algorithms, e.g. in PDAD and DAA</td>
</tr>
<tr>
<td><strong>Data-centric networking</strong></td>
<td>probabilistic data forwarding in gossiping approaches; agent based techniques relying on random waypoint strategies, e.g. in rumor routing</td>
</tr>
<tr>
<td><strong>Clustering</strong></td>
<td>randomized clusterhead selection to maximize the network lifetime and to provide fair distribution of the energy load, e.g. in LEACH and HEED</td>
</tr>
</tbody>
</table>
## Probabilistic techniques

### Coordination and control

<table>
<thead>
<tr>
<th>Communication and coordination</th>
<th>randomization through variation of network latencies; randomly distributed back-off delay, e.g. in Span; probabilistic state transitions, e.g. in DEPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaboration and task allocation</td>
<td>probabilistic decision processes and task allocation according to estimations for winning the contest for a new task or the nest leaving probability</td>
</tr>
</tbody>
</table>
Evaluation Criteria

- Scalability
- Energy considerations
- Network lifetime
Scalability

- Protocol overhead
  - Number and size of state information that must be stored and maintained at each node in the network
  - Direct communication overhead – goodput vs. network load

- Capacity of wireless networks
  - Bounded capacity of wireless networks according to Gupta & Kumar

- Reduced determinism
  - Scalability vs. predictability

![Graph showing the relationship between determinism and scalability in different types of systems: centralized control, distributed systems, and self-organized systems.](image)
Energy considerations

- **Constraints on the battery source**
  - Battery size is direct proportional to its capacity

- **Selection of optimal transmission power**
  - Energy consumption increases with an increase in the transmission power (which is also a function of the distance between communicating nodes)
  - Optimal transmission power decreases the interference among nodes, which, in turn, increases the number of simultaneous transmissions

- **Channel utilization**
  - As seen before, a reduction of the transmission power increases frequency reuse → better channel utilization
  - Power control becomes especially important in CDMA-based systems
Battery Management

- Battery lifetime estimation
  - Manufacturer-specified rated capacity, discharge plot of the battery
  - Discharge current ratio can be computed
  - Efficiency is calculated by the interpolation of point in the discharge plot

- Recovery capacity effect
  - In idle conditions, the charge of the cell recovers by increasing the idle time, the theoretical capacity of the cell may be used
  - Battery scheduling
Battery-Scheduling Techniques

- **Delay-free approaches**
  - As soon as a job arrives, the battery charge for processing the job will be provided from the cells without any delay
    - Joint technique (JN) - the same amount of current is drawn equally from all the cells, i.e. each cell is discharged by 1/L of the current required
    - Round robin technique (RR) - batteries are selected in round robin fashion, the current job gets the required energy from the selected cell
    - Random technique (RN) - similar to RR but the cells are selected randomly
Battery-Scheduling Techniques

- No delay-free approaches
  - The batteries coordinate among themselves based on their remaining charge
  - E.g., by defining a threshold for the remaining charge → all the cells which have their remaining charge greater than the threshold value become eligible for providing energy
  - Delay-free approaches can be applied to the eligible cells
  - Non-eligible cells stay in recovery state to maximize their capacity

- Further enhancements
  - Heterogeneous battery-scheduling technique
Energy Consumption

- A “back of the envelope” estimation

- Number of instructions
  - Energy per instruction: 1 nJ
  - Small battery (“smart dust”): 1 J = 1 Ws
  - Corresponds: $10^9$ instructions!

- Lifetime
  - Or: Require a single day operational lifetime = $24 \times 60 \times 60 = 86400$ s
  - $1 \text{ Ws} / 86400 \text{ s} \approx 11.5 \mu\text{W}$ as max. sustained power consumption!

$\rightarrow$ Not feasible!
Multiple Power Consumption Modes

- Way out: Do not run sensor node at full operation all the time
  - If nothing to do, switch to **power safe mode**
  - Question: When to throttle down? How to wake up again?

- Typical modes
  - Controller: Active, idle, sleep
  - Radio mode: Turn on/off transmitter/receiver, both

- Multiple modes possible, “deeper” sleep modes
  - Strongly depends on hardware
  - TI MSP 430 (@ 1 MHz, 3V):
    - Fully operation 1.2 mW
    - Deepest sleep mode 0.3 μW – only woken up by external interrupts (not even timer is running any more)
  - Atmel ATMega
    - Operational mode: 15 mW active, 6 mW idle
    - Sleep mode: 75 μW
Processor Power Management Schemes

- Power-saving modes
  - Key idea: remain in sleep mode as long as possible
  - Example: RAS – remote activated switch
    - Receiver and control logic can be turned off until a packet is received
    - Caution: the preamble must be long enough for turning on and initializing the receiver
Transmitter Power/Energy Consumption for $n$ Bits

- Amplifier power: $P_{\text{amp}} = \alpha_{\text{amp}} + \beta_{\text{amp}} P_{\text{tx}}$
  - $P_{\text{tx}}$ \textit{radiated power}
  - $\alpha_{\text{amp}}, \beta_{\text{amp}}$ constants depending on model
  - Highest efficiency ($\eta = P_{\text{tx}} / P_{\text{amp}}$) at maximum output power
  - In addition: transmitter electronics needs power $P_{\text{txElec}}$

- Time to transmit $n$ bits: $n / (R \times R_{\text{code}})$
  - $R$ nominal data rate, $R_{\text{code}}$ coding rate

- To leave sleep mode
  - Time $T_{\text{start}}$, average power $P_{\text{start}}$

\[ E_{\text{tx}} = T_{\text{start}} P_{\text{start}} + n / (R \times R_{\text{code}}) (P_{\text{txElec}} + \alpha_{\text{amp}} + \beta_{\text{amp}} P_{\text{tx}}) \]

- Simplification: Modulation not considered
Computation vs. Communication Energy Cost

- Tradeoff?
  - Directly comparing computation/communication energy cost not possible
  - But: put them into perspective!
  - Energy ratio of “sending one bit” vs. “computing one instruction”:
    → anything between 220 and 2900 in the literature
  - Transmitting (send & receive) one kilobyte ≈ computing three million instructions!

- Hence: try to compute instead of communicate whenever possible

- Key technique in WSN – *in-network processing!*
  - Exploit compression schemes, intelligent coding schemes, …
Network lifetime

- Considered as a comprehensive evaluation metric for sensor networks

- Individual parameters $\zeta(\tau)$
  - Active nodes, alive nodes, availability / service disruption tolerance
  - Area coverage, target coverage, $k$-coverage
  - Latency, loss, connectivity
  - Connected coverage

- Livelihood
  - $\zeta(t)$: if all $\zeta(\tau)$ are provided

- Lifetime measures
  - Accumulated network lifetime $Z_a$ is the sum of all times the network is alive
  - Total network lifetime $Z_t$ is the time at which the liveliness criterion is lost for a time period longer than the service disruption tolerance
Summary (what do I need to know)

- **Self-organization techniques**
  - Basic methods (positive and negative feedback, interactions among individuals and with the environment, probabilistic techniques)
  - Applicability in sensor and actor networks

- **Evaluation criteria**
  - Scalability – limiting factors
  - Energy considerations (limitations, battery management)
  - Network lifetime
References