Self-Organization

Advanced Mobile Communication Networks
Winter Semester 2013/14

Integrated Communication Systems Group
Ilmenau University of Technology
Motivation for Self-Organization

- Problem of today's networks
  - Heterogeneity
  - Dynamics
  - Scalability

- Method: Self-organization
  - Fast, autonomous reaction to problems
  - Automation of control
  - Distributed control

- Some application scenarios:
  - Severe network impact due to disasters
  - Energy savings
  - Privately operated femto cells
## Self-Organization – Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| No central control   | No global control system  
                        | No global information  
                        | Subsystems perform completely autonomous                                                                 |
| Emerging structures  | Global behavior or functioning of the system emerges in form of observable pattern or structures                                       |
| Resulting complexity | Even if the individual subsystems can be simple as well as their basic rules, the resulting overall system becomes complex and often unpredictable |
| High scalability     | No performance degradation if more subsystems are added to the system  
                        | System performs as requested regardless of the number of subsystems                                   |
Self-Organization and Emergence

Definition **Self-Organization**
Self-organization is a process in which structure and functionality (pattern) at the global level of a system emerge solely from numerous interactions among the lower-level components of a system without any external or centralized control. The system's components interact in a local context either by means of direct communication or environmental observations without reference to the global pattern.

Definition **Emergence**
Emergent behavior of a system is provided by the apparently meaningful collaboration of components (individuals) in order to show capabilities of the overall system (far) beyond the capabilities of the single components.
Basic Methods in Self-Organizing Systems

- Positive and negative feedback
- Interactions among individuals and with the environment
- Probabilistic techniques
Positive and Negative Feedback

Simple feedback

Amplification problems

Feedback

Input

System state

Output

Measurement

Snowballing effect

Implosion effect
Interactions among Individuals and with the Environment

- Direct communication among neighboring systems
- Indirect communication via the environment (stigmergy)
- Interaction with (stimulation by) the environment
Probabilistic Techniques

- Examples: stochastic processes, random walk
- Objectives: leaving local optima, stabilization

Diagram:

1. Initialization
2. Random walk
3. Bumped into a wood chip
   - Yes: Carry a wood chip
   - No: Pick up chip
4. Drop chip
Self-Organized vs. Centralized Control

Self-organizing networks

- Local state
- Neighbor information
- Probabilistic methods

Networking functions for global connectivity and efficient resource usage

Implicit coordination

Centralized networks

- Global state (globally optimized system behavior)
- Explicit coordination
Swarm Intelligence
Swarm Intelligence (SI)

- A computational technique for solving distributed problems inspired from biological examples provided by
  - social insects such as ants, termites, bees, and wasps and by swarm, herd, flock, and
  - shoal phenomena such as fish shoals

- An approach for controlling and optimizing distributed systems

- Resilient, decentralized, self-organized technique
SI Organizing Principles

SI has the following notable features:

• Autonomy:
  ➔ The system does not require outside management or maintenance. Individuals are autonomous, controlling their own behavior both at the detector and effector levels in a self-organized way

• Adaptability:
  ➔ Interactions between individuals can arise through direct or indirect communication

• Scalability:
  ➔ SI abilities can be performed using groups consisting of a few, up to thousands of individuals with the same control architecture

• Flexibility:
  ➔ No single individual of the swarm is essential, that is, any individual can be dynamically added, removed, or replaced
SI Organizing Principles

SI has the following notable features:

• Robustness:
  ➔ No central coordination takes place, which means that there is no single point of failure

• Massively parallel:
  ➔ Tasks performed by each individual within its group are the same

• Self-organization:
  ➔ The intelligence exhibited is not present in the individuals, but rather emerges somehow out of the entire swarm
SI Communication Forms

**Indirect Communication**
- Implicit communication that takes place between individuals via the surrounding environment.
- Known as Stigmergy communication.

**Direct Communication**
- Explicit communication that can also take place between individuals.
- Examples:
  - waggle dance of the honeybee,
  - trophallaxis (food or liquid exchange, e.g., mouth-to-mouth food exchange by honeybees),
  - …
Stigmergy

- Stigmergy: stigma (sting) + ergon (work) → “stimulation by work”

- Characteristics of stigmergy
  - Indirect agent interaction by modification of the environment
  - The information is local: it can only be accessed by insects that visit the location in which it was released
  - Work can be continued by any individual

$S_i$: Pillar construction state
$R_i$: Response
Ant Colony Optimization (ACO)
Ants

• Why are ants interesting?
  – Ants solve complex tasks by simple local means
  – Ants productivity is better than the sum of their single activities
  – Ants are grand masters in search and exploitation
Foraging behavior of Ants (Double bridge experiment)

Ants start with equal probability
Foraging behavior of Ants (Double bridge experiment)

The ant on shorter path has a shorter time from it’s nest to the food.
Foraging behavior of Ants (Double bridge experiment)

The density of pheromone on the shorter path is higher
Foraging behavior of Ants (Double bridge experiment)

The next ant takes the shorter route
Foraging behavior of Ants (Double bridge experiment)

After some time, the shorter path is almost exclusively used.
Ant Colony Optimization (ACO)

- Developed by Dorigo and Di Caro
- It is a population-based metaheuristic used to find approximate solutions to difficult optimization problems
- ACO is structured into three main functions:
  1. AntSolutionsConstruct( )
     - Performs the solution construction process
  2. PheromoneUpdate( )
     - Performs pheromone trail updates
     - Includes also pheromone trail evaporation
  3. DaemonActions( )
     - An optional step in the algorithm which involves applying additional updates from a global perspective
ACO Metaheuristic

- Pheromone value update
- Probabilistic solution construction
- Initialization of pheromone values

- ACO problem
- Solution components
- Pheromone model

Integrated Communication Systems Group

Advanced Mobile Communication Networks, Master Program
Ant Foraging and ACO

<table>
<thead>
<tr>
<th>Biology (Ant Foraging)</th>
<th>ACO Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ant</td>
<td>Individual (agent) used to build (construct) a solution</td>
</tr>
<tr>
<td>Ant Colony</td>
<td>Population (colony) of cooperating individuals</td>
</tr>
<tr>
<td>Pheromone Trail</td>
<td>Modification of the environment caused by the artificial ants in order to provide an indirect mean of communication with other ants of the colony. Allows assessment of the quality of a given edge on a graph</td>
</tr>
<tr>
<td>Pheromone Evaporation</td>
<td>Reduction in the pheromone level of a given path due to aging</td>
</tr>
</tbody>
</table>
Properties of the Artificial Ant

• Each artificial ant has an internal memory

• Starting in an initial state $S_{\text{initial}}$ each ant tries to build a feasible solution to the given problem, moving in an iterative fashion through its search space

• The guidance factors for ant movements is a transition rule which is applied before every move from state $S_i$ to state $S_j$

• The amount of pheromone each ant deposits is governed by a problem specific pheromone update rule

• Pheromone deposition may occur at every state transition during the solution construction (pheromone trail update)

• Ants may retrace their paths once a solution has been constructed and only then deposit pheromone, all along their individual paths
The original Ant System

- Developed by Dorigo et al. (1996)

- Ant system (AS)
  - First ACO algorithm
  - Pheromone updated by all ants in the iteration

- Travelling Salesman Problem (TSP) was used as a test-bed for this algorithm
Travelling Salesman Problem (TSP)

- TSP description:
  - Visit cities in order to make sales
  - Save on travel costs
  - Visit each city once (Hamiltonian circuit)

- A Hamiltonian cycle (or Hamiltonian circuit) is a cycle in an undirected graph which visits each vertex exactly once and also returns to the starting vertex.
Solution for TSP

- A connected graph $G=(V,E)$, where
  - $V$ is a set of vertices (cities)
  - $E$ is a set of edges (connection between cities)
- A variable called pheromone is associated with each edge and can be read and modified by ants
- Ant system is an iterative algorithm. At each iteration,
  - A number of artificial ants are considered
  - Each ant build a solution by walking from vertex to vertex
  - Each vertex is visited one time only
  - An ant selects the following vertex to be visited according to a stochastic mechanism that is biased by the pheromone
- At the end, the pheromone values are updated in order to bias ants in the future iteration to construct solutions similar to the previously constructed
Ant System and the TSP

1. Pheromone trail
   - Iteration is defined as the interval in \((t,t+1)\) where each of the \(N\) ants moves once
   - Epoch \(\Rightarrow n\) iterations (when each ant has completed a tour)
   - Intensity of trail: \(\tau_{ij}(t)\)
   - Trail update function after each epoch:
     \[
     \Delta \tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \sum_{k=1}^{N} \Delta \tau_{ij}^k
     \]
     
     - \(\rho\) is the evaporation rate
     - \(\Delta \tau_{ij}^k\) is the quantity of pheromone laid on path \((i,j)\) by the ant \(k\) and is given by:
     \[
     \Delta \tau_{ij}^k = \begin{cases} 
         Q / L_k & \text{If ant } k \text{ used edge } (i,j) \text{ in its tour,} \\
         0 & \text{otherwise,}
     \end{cases}
     \]
     where \(Q\) is a constant and \(L_k\) is the tour length of \(k\)th ant.
Ant System and the TSP

2. Memory
   - Prevents town repeats
   - Tabu list

3. Awareness of environment
   - City distance
   - Visibility: \( \eta_{ij} = \frac{1}{d_{ij}} \)
     where \( d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \)

4. Probability function
   \[
   p_{ij}^k(t) = \begin{cases} 
   \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{j \in \text{allowed}} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta} & \text{if } j \in \text{allowed} \\
   0 & \text{otherwise} 
   \end{cases}
   \]
Some applications of ACO

• Scheduling

• Routing problems
  – Traveling Salesman Problem (TSP)
  – Vehicle routing
  – Network routing
  – Message Ferrying in Delay Tolerant Networks (DTNs)

• …
Conclusion

Self-organization: what does it mean?

– distributed, autonomous control of systems
– simple local algorithms or behavior
– interaction with neighbors and/or environment only

Lots of applications of self-organization already

– self-organization keeps today’s Internet together
– development and tuning of self-organized algorithms takes time
– stability is an issue

Swarm Intelligence

– is a rich source of inspiration for our computer systems.
– has many features that are desirable for distributed computing such as auto-configuration, auto-organization, autonomy, scalability, flexibility, robustness, emergent behavior, and adaptability
Example from ICS Research
Common Control Channel Design
in Multichannel Ad Hoc Networks
(P. M. R. dos Santos, O. Artemenko)
Motivation

• Problem
  – Establishment and maintenance of common control channels (CCC) in a dynamic environment

Which network to join?

- Distributed consensus agent
- \(n^{th}\) ring-participant
- Co-located interferer
Distributed Consensus Algorithm [1]

Set of spectrum opportunities

\[ M = \{1, \ldots, m\} \]

Utility Function

\[ U(m) = \frac{B_m}{N_m^2} \sum_{n \in N_m} \log_2 \left( 1 + \frac{1}{2} \text{SINR}_n \right) \]

Handover solution

\[ \arg \max_m U(m) \]

- Distributed consensus agent
- \( n \)th ring-participant
- Co-located interferer
- Received signal vector
- Interference vector
- Direction of token rotation

Token-Ring Timing Diagram

Research Idea:
Token-embedded pilot tone for SINR estimation

Payload P1  T2  ■ ■ ■  Pn  T1

Ring-participant 1  Ring-participant n

Token Holding Time

Maximum Token Rotation Time
# Simulation Setup

<table>
<thead>
<tr>
<th>Simulation parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation area</td>
<td>1 km²</td>
</tr>
<tr>
<td>Network spatial deployment</td>
<td>Random</td>
</tr>
<tr>
<td>Propagation model</td>
<td>Free space path loss</td>
</tr>
<tr>
<td>Channel bandwidth</td>
<td>200 kHz</td>
</tr>
<tr>
<td>CR TX power (EIRP)</td>
<td>30 dBm</td>
</tr>
<tr>
<td>SINR threshold</td>
<td>20 dB</td>
</tr>
<tr>
<td>Receiver noise floor</td>
<td>-147 dB</td>
</tr>
<tr>
<td>Network mobility model</td>
<td>Random</td>
</tr>
</tbody>
</table>
Simulation Results – Network Capacity

\[ \Delta = 31\% \]
\[ \Delta = 43\% \]
\[ \Delta = 59\% \]
Demonstration Video

- Random channel hopping vs distributed consensus

- Simulation parameters
  - 200 Cognitive Radios
  - 1 km² simulation area
  - Free space path loss propagation model

3 min video
References

Self-organization in general


Application of Self-organization to communication networks


Swarm Intelligence

Contact

Integrated Communication Systems Group
Ilmenau University of Technology

Prof. Dr.-Ing. habil. Andreas Mitschele-Thiel
Dr.-Ing. Oleksandr Artemenko

fon: +49 (0)3677 69 2819/4123
fax: +49 (0)3677 69 1226
e-mail: mitsch, oleksandr.artemenko@tu-ilmenau.de

Visitors address:
Technische Universität Ilmenau
Helmholtzplatz 5
Zuse Building, room 1032/1071
D-98693 Ilmenau

www.tu-ilmenau.de/ics