Discrete Event Simulation

IN2045

Dr. Alexander Klein
Stephan Günther
Prof. Dr.-Ing. Georg Carle

Chair for Network Architectures and Services
Department of Computer Science
Technische Universität München
http://www.net.in.tum.de
Topics

- Mobility in General
  - Realistic Movement
  - Human Mobility Pattern

- Visualization
  - Density
  - Speed Histograms

- Characteristics of Mobility Pattern
  - Link Duration, Transient Phase, Node Distribution, Speed Distribution, Correlated Movement

- Synthetic Mobility Models
  - Random Waypoint
  - Random Direction
  - Random Walk
  - Levi-Flight
  - Brownian Motion
  - Group Mobility
Mobility

- What is (random) mobility?

- Nodes moving from one hotspot to another
- Fix Nodes
- Bypassing nodes
- Hotspot
Mobility

- **Why simulate mobility?**
  - Improvements in technology enable new technologies and result in cheaper hardware prices
  - Number of powerful mobile devices increases very quickly (Smartphones with high data rate interfaces)
  - Number of applications for mobile devices increases
  - Impact on the system performance can often not be predicted in advance

- **Impact on wireless networks:**
  - Topology depends on the user mobility
  - Routing protocols have to react on topology changes (link duration)
  - Frequent changes of the user density result in variation of the interference
  - May lead to a collapse of the network if the applied protocols are not optimized (overhead, dissemination of outdated information)
  - Enables new information dissemination strategies (Delay-Tolerant-Networking)
Mobility

- What is realistic movement?
  - Random movement?
  - Correlated movement?
  - Movement of humans?

- Mobility Pattern
  - Pedestrians
  - Police patrol / avalanche rescue
  - Cars on the road
  - Trains
  - Air planes
  - Animals (hunter and prey)
  - Constraint by obstacles / infrastructure

Antony Gormley’s *Quantum Cloud* sculpture in London (based on a random walk model)
Mobility

- Human mobility pattern:
  - Short-term and long-term characteristics
  - Often approximated by the levy-flight synthetic mobility model which is derived from the random walk model
  - High probability that the next position is close to the previous one
  - Low probability that the individual travels long distances
  - High variation between different individuals

Mobility

- Simulation
  - Area (circle, square, rectangle, sphere, torus, …)

- Long-term simulation
  - Transient phase of the model
  - Node distribution
  - Speed distribution
  - Partitioning of the network
Mobility

- Simulation
  - Bouncing rule:
    - bounce
    - reflect
    - wrap-around
    - delete & replace

Node Distribution changes depending on the applied bouncing rule

Article (Bettstetter2001)
Bettstetter, C.
Mobility Modeling in Wireless Networks: Categorization, Smooth Movement, and Border Effects
Mobility

- Visualization
  - Movement (Debugging)
    - Debugging
    - Detect correlated movement
    - Evaluation
  - Density
    - Spatial node distribution
    - Border effects
    - Estimation of transient phase
  - Histograms
    - Node speed distribution
    - Link duration
    - Estimation of transient phase
Mobility

- **Characteristics:**
  - Link duration
    - Important wireless communication parameter
    - Represents the time interval during which two nodes are able to communicate with each other
  - Transient phase
    - One or more parameters change significantly during this phase
    - Duration of the transient phase varies between different synthetic mobility models
  - Spatial node distribution
    - Depends on the mobility model
    - Often affected by the shape of the simulation plane
    - Influenced by the applied bouncing rule
Mobility

- **Characteristics:**
  - Speed distribution
    - Good indicator for the duration of the transient phase
    - Mainly influenced by the following parameters:
      - Time-based or distance-based movement decision
      - Pause time
      - Shape of the simulation plane
  - Correlated / Constraint movement
    - Each move is affected by the previous one
    - Objects may interact with each other
  - Group mobility
    - The movement of objects is a composition of the movement of the individual and a common (group leader) object
Mobility

- Synthetic Mobility Models
Random Waypoint

Algorithm:

Scenario Boundary

Mobile Node
Mobility

- Random Waypoint

Algorithm:

Node chooses a random position.
Mobility

- Random Waypoint

Algorithm:

Scenario Boundary

Node selects a uniform distributed movement speed
Mobility

Random Waypoint

Algorithm:

Scenario Boundary

Node moves towards the new position
Random Waypoint

Algorithm:

Node waits a certain period of time
Mobility

- **Random Waypoint**

Algorithm:

```
Node selects new destination
```

---

Scenario Boundary

Node selects new destination
Mobility

Random Waypoint

Algorithm:

Step 1: Select a random destination within the scenario

Step 2: Select a random speed \( speed \in [speed_{\text{Min}}; speed_{\text{Max}}] \)

Step 3: Move until the destination is reached

Step 4: Wait a random period of time \( pause \in [0; pause_{\text{Max}}] \)

Step 5: Go to step 1
Random Waypoint

- Characteristics:
  - Node density decreases towards the border
  - Highest node density in the center
  - The fraction of slow moving nodes increases over time
  - Long transient phase
  - Individual nodes recognize density waves while moving through the center
  - Average node speed decreases over time  

- Advantage:
  - Simple to implement
  - Challenging mobility due to changing node density

- Disadvantage:
  - Has to be configured carefully (Minimum speed and pause duration)
  - Movement affected by the shape of the simulation plane
Random Waypoint

- Node speed distribution:

(a) 100 Seconds
(b) 200 Seconds
(c) 400 Seconds
(d) 800 Seconds
Mobility

- Random Waypoint
  - Node density:

(a) 100 Seconds
(b) 200 Seconds
(c) 400 Seconds
(d) 800 Seconds
Mobility

- Random Direction

Algorithm:

Scenario Boundary

Mobile Node
Random Direction

Algorithm:

Node selects a random direction and speed
Random Direction

Algorithm:

Node moves until it reaches the border of the scenario
Random Direction

Algorithm:

Node waits a random period of time
Mobility

- Random Direction

Algorithm:

Node selects a random direction and speed
Random Direction

Algorithm:

Step 1: Select a random direction $direction \in [0; 2\pi]$ (such that the node does not leave the scenario)

Step 2: Select a random speed $speed \in [speed_{Min}; speed_{Max}]$

Step 3: Move until the border of the scenario is reached

Step 4: Bouncing rule:
   a. Wait a random period of time $pause \in [0; pause_{Max}]$
   b. Delete the node and replace it with a new node in the center or at a random position
   c. Place the node at the opposite side of the simulation plane

Step 5: Go to step 1
Mobility

- Random Direction
  - Characteristics:
    - Node density increases towards the border
    - Highest node density at the border and in the corners
    - The fraction of slow moving nodes increases over time
    - Short transient phase
    - Nodes in the corner are strongly affected by the applied bouncing rule
  - Advantage:
    - Simple to implement
    - Uniform distributed node density (depends on the bouncing rule)
  - Disadvantage:
    - Has to be configured carefully (Minimum speed and pause duration)
    - Movement affected by the shape of the simulation plane
    - Large impact of the bouncing rule
Random Walk (time-based / distance-based)

Algorithm:
Random Walk (time-based / distance-based)

Algorithm:

Node selects a random direction and speed.
Random Walk (time-based / distance-based)

Algorithm:

Node moves in that direction for a random period of time or a random distance
Random Walk (time-based / distance-based)

Algorithm:

Node pauses a random period of time
Random Walk (time-based / distance-based)

Algorithm:

Node selects a random direction and speed
Random Walk (time-based / distance-based)

Algorithm:

Step 1: Select a random speed \( speed \in [speed_{Min}; speed_{Max}] \)

Step 2: Select a random direction \( direction \in [0;2\pi] \)

Step 3: Move into that direction
   a. for a pre-defined period of time
   b. for a certain distance
   c. if the border of the scenario is reached, select a new direction (bouncing rule)

Step 4: Wait a random period of time \( pause \in [0; pause_{Max}] \)

Step 5: Go to step 1
Mobility

- Random Walk (time-based)
  - Characteristics (time-based):
    - Node density (almost) uniform distributed
    - Nodes in are affected by the applied bouncing rule
    - Node speed uniform distributed
  - Advantage:
    - Simple to implement
    - Uniform distributed node density (depends on the bouncing rule)
  - Disadvantage:
    - Has to be configured carefully
      - Minimum speed
      - Pause duration
      - Travel duration
    - Affected by the bouncing rule
    - Required computational power depends on the movement duration
Mobility

- **Random Walk (distance-based)**
  - Characteristics (distance-based):
    - Node density (almost) uniform distributed
    - Nodes in the corner are affected by the applied bouncing rule
    - Node speed decreases over time (similar to RWP)
    - Speed decay problem
  - Advantage:
    - Simple to implement
    - Uniform distributed node density (depends on the bouncing rule)
  - Disadvantage:
    - Has to be configured carefully
      - Minimum speed
      - Pause duration
      - Travel distance
    - Movement affected by the shape of the simulation plane
    - Required computational power depends on the travel distance
Mobility

- Random Walk (time-based)
  - Node speed distribution:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>speed_{min}</td>
<td>1 m/s</td>
</tr>
<tr>
<td>speed_{max}</td>
<td>20 m/s</td>
</tr>
<tr>
<td>pause_{min}</td>
<td>0 s</td>
</tr>
<tr>
<td>pause_{max}</td>
<td>0 s</td>
</tr>
<tr>
<td>Movement</td>
<td>time-based</td>
</tr>
<tr>
<td>Movement Duration</td>
<td>10 s</td>
</tr>
</tbody>
</table>
Mobility

- Random Walk (distance-based)
  - Node speed distribution:

```
<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>speed_{min}</td>
<td>1 m/s</td>
</tr>
<tr>
<td>speed_{max}</td>
<td>20 m/s</td>
</tr>
<tr>
<td>pause_{min}</td>
<td>0 s</td>
</tr>
<tr>
<td>pause_{max}</td>
<td>0 s</td>
</tr>
<tr>
<td>Movement</td>
<td>distance-based</td>
</tr>
<tr>
<td>Travel Distance</td>
<td>200 m</td>
</tr>
</tbody>
</table>
```
Mobility

- Random Walk (time-based)
  - Node density:

![Graphs showing node density over time](image)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>speed_{min}</td>
<td>1 m/s</td>
</tr>
<tr>
<td>speed_{max}</td>
<td>20 m/s</td>
</tr>
<tr>
<td>pause_{min}</td>
<td>0 s</td>
</tr>
<tr>
<td>pause_{max}</td>
<td>0 s</td>
</tr>
<tr>
<td>Movement</td>
<td>time-based</td>
</tr>
<tr>
<td>Movement Duration</td>
<td>10 s</td>
</tr>
</tbody>
</table>
Mobility

- Random Walk (distance-based)
  - Node density:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>speed_{Min}</td>
<td>1 m/s</td>
</tr>
<tr>
<td>speed_{Max}</td>
<td>20 m/s</td>
</tr>
<tr>
<td>pause_{Min}</td>
<td>0 s</td>
</tr>
<tr>
<td>pause_{Max}</td>
<td>0 s</td>
</tr>
</tbody>
</table>

Movement: distance-based

Travel Distance: 200 m
Mobility

- **Random Walk**
  - Lévy flight
    - Distance-based random walk
    - Distance is chosen according to a heavy-tailed distribution
    - Probability is high that the object only moves a short distance
    - Probability is low that the object moves straight over a long distance
    - Often used to simulate the movement of humans and animals

- **Brownian Motion**
  - Distance-based random walk
  - Travel distance between subsequent points is close to zero
  - Describes the movement of small particles in liquids

Example: Lévy flight
Example: Brownian Motion
Mobility

- **Random Walk** (according to Turchin)
  - **Uncorrelated random walk:**
    - Previous move does not affect the following move
    - Each move is independent from the previous one
  - **Correlated random walk:**
    - Previous move affects the following move
      - High probability of moving into the same direction
      - Long travels are followed by short travels with high probability
  - **Biased random walk:**
    - The probability of moving in a certain direction is higher than moving into other directions (non-uniform selection of the direction)
  - **Biased correlated random walk:**
    - Each move is affected by the previous one and an absolute direction
  - **Constrained random walk:**
    - Measured parameters and estimated distributions are used as input for the synthetic mobility model
    - The direction and speed are chosen with respect to the measurements
Random Group Mobility

Algorithm:

Define a group of Nodes

Scenario Boundary
Random Group Mobility

Algorithm:

- Select one node as group leader.
Random Group Mobility

Algorithm:

Choose a maximum allowed distance between fellow nodes and the group leader
Random Group Mobility

Algorithm:

- All other nodes are fellow nodes who follow the leader.
Random Group Mobility

Algorithm:

Select a random distributed speed and direction for the group leader.
Random Group Mobility

Algorithm:

Calculate the position of the group leader at the next step
Random Group Mobility

Algorithm:

Scenario Boundary

Calculate the allowed area of the fellow nodes
Random Group Mobility

Algorithm:

Select new positions of the fellows
Random Group Mobility

Algorithm:

Scenario Boundary

Move to the next destination
Random Group Mobility

Algorithm (1/2):

Preliminary steps

Step 1: Define a group of nodes
Step 2: Select one node as group leader and mark the others as fellows
Step 3: Choose the maximum allowed distance between a fellow node and the group leader

Group leader

Step 4: Select a random speed $speed \in [speed_{Min}; speed_{Max}]$
Step 5: Select a random direction $direction \in [0; 2\pi]$
Step 6: Go to step 10
Step 7: Move into that direction
   a. for a pre-defined period of time / remaining movement duration
   b. for a pre-defined distance
   c. Go to step 15 if the border of the scenario is reached before the movement is complete
Step 8: Wait a random period of time $pause \in [0; pause_{Max}]$
Step 9: Go to step 4
Mobility

Random Group Mobility

Algorithm (2/2):

Fellow nodes

Step 10: Calculate the position of the group leader at the next movement / bouncing position
Step 11: Calculate the allowed area around the group leader at the next movement / bouncing position
Step 12: Choose a random position within the allowed area
Step 13: Calculate speed and direction such that the new position is reached at the same time the group leader reaches its next movement / bouncing position
Step 14: Go to step 7

Group leader

Step 15: Select a new direction of the group leader
Step 16: Go to step 10
Random Group Mobility

- Node speed distribution:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leader Mobility</td>
<td>Random Walk (time-based)</td>
</tr>
<tr>
<td>Leader-Fellow Distance</td>
<td>&lt; 200m</td>
</tr>
<tr>
<td>Fellow Area Distance</td>
<td>circle</td>
</tr>
<tr>
<td>speed_{Min}</td>
<td>5 m/s</td>
</tr>
<tr>
<td>speed_{Max}</td>
<td>20 m/s</td>
</tr>
<tr>
<td>pause_{Max}</td>
<td>0 s</td>
</tr>
<tr>
<td>Movement Duration</td>
<td>20 s</td>
</tr>
</tbody>
</table>
Mobility

- Random Group Mobility
  - Node density:

<table>
<thead>
<tr>
<th>Leader Mobility</th>
<th>Random Walk (time-based)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leader-Fellow Distance</td>
<td>&lt; 200 m</td>
</tr>
<tr>
<td>Fellow Area</td>
<td>circle</td>
</tr>
<tr>
<td>speed_{Min}</td>
<td>5 m/s</td>
</tr>
<tr>
<td>speed_{Max}</td>
<td>20 m/s</td>
</tr>
<tr>
<td>pause_{Max}</td>
<td>0 s</td>
</tr>
<tr>
<td>Movement Duration</td>
<td>20 s</td>
</tr>
</tbody>
</table>
Mobility

- **Obstacles:**
  - Movement of objects is usually constraint by
    - obstacles
    - pre-defined pathways
  - Bouncing rule becomes more important with an increasing number of obstacles
  - Obstacles block movement but do not necessarily affect the signal propagation (e.g. river or lake)
  - Some models use Voronoi diagrams as predefined paths
Mobility

- How to describe position and orientation?
  - Position:
    - Geographic
      Latitude φ, Longitude λ, Altitude
    - Cartesian
      X, Y, Z
  - Orientation:
    - Yaw
    - Pitch
    - Roll

Picture taken from nasa.gov
Mobility

- **Implementation:**
  - **Types of mobility**
    - **Direct**
      - Change the position and orientation of objects directly at a given simulation time
    - **Trajectory**
      - Sequence of triples [position, orientation, simulation time] which describe the position and orientation at a given simulation time
      - The movement is usually interpolated between subsequent triples
    - **Vector**
      - Bearing, ground speed, ascent rate
      - Trajectories can be described by [bearing, ground speed, simulation time] triples
    - **External modification**
      - Co-simulation
      - Hardware-In-The-Loop
      - Can use any type of mobility