Optimal Schedule for LTE in D2D Mode

Maciej Mühleisen, Jan Habermann
Outline

Motivation

D2D Communication

Optimal Schedule

Simulation Model

Results

Conclusion & Outlook
Motivation

Grand vision: the “Airborne Internet”


Scheduled Airline Routes (January 2012)


Challenges

- Coverage over the entire earth
- No ground infrastructure over large bodies of water
- Satellite-based communication is expensive and has high delays

Motivation / Idea: Using LTE in Device-to-Device (D2D) mode for direct Air-to-Air (A2A) communication between aircraft
Motivation

D2D communication allows nodes to communicate with each other without involving a base station!

Default “star”-topology (Cellular Network)  Network with D2D communication

Serving link  Interfering link
Different Modulation and Coding Schemes (MCS) are used based on the Channel Quality Indicator (CQI)

A2A communication can not rely on a deployed ground infrastructure (e.g. across the Atlantic Ocean)

**Resource Management Challenges**

- No central coordinator (base station)
- Transmitting on the same Physical Resource Blocks (PRBs) causes interference
- Interference becomes a major issue
Optimal Schedule

Example with four nodes and only two demands:

Traffic Demands
Optimal Schedule

Example with four nodes and only two demands:

Traffic Demands

Network States

(a) Only A gets to send to B  (b) Only C gets to send to D  (c) Both get to send in parallel

State 1  State 2  State 3
Example with four nodes and only two demands:

End of the scheduling round (t = 0.888 s) ➔ 22.5 Mbit/s
Optimal Schedule

Example with four nodes and only two demands:

- **Demand Time Series - State 3 + State 2**

- **Remaining Demand [Mbit]**
  - Node A
  - Node C

- **Time [s]**
  - 0.4 s
  - 0.333 s

- **End of the scheduling round**
  - \(t = 0.733\) s

- **27.3 Mbit/s**
Optimal Schedule

Feasible Network States

General assumptions:

- Every node either transmits on all PRBs or not at all (TDMA)

- No node can have more than one outgoing link (cannot sent to multiple recipients at once)

- No node can receive from more than one transmitter at a time

- No node can transmit and receive at the same time

**Problem:** Which states should be active and how long should they be active to maximize the capacity of the network?
Optimal Schedule

Problem formulation through Linear Programming


1. Generate all feasible network states and specify demands (buffer fill levels)

2. Convert network states and demands to matrices

3. Solve the Linear Program:

\[
S = \{S_1, S_2, ..., S_n\}
\]

\[
\min \sum_{i=1}^{\mid S \mid} t_i \quad \text{s.t.} \quad \sum_{i=1}^{\mid S \mid} t_i \cdot S_i \geq D
\]

Example

\[
\begin{align*}
\min & \quad t_1 + t_2 + t_3 \\
\text{s.t.} & \quad t_1 \cdot \begin{pmatrix} 0 & 30 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} + t_2 \cdot \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 18 \\ 0 & 0 & 0 & 0 \end{pmatrix} + t_3 \cdot \begin{pmatrix} 0 & 25 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \end{pmatrix} \geq \begin{pmatrix} 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \end{pmatrix}
\end{align*}
\]

Solution: \(0 \text{ s} \cdot S_1 + 0.333 \text{ s} \cdot S_2 + 0.4 \text{ s} \cdot S_3\)
Providing a baseline for comparison:

The baseline only uses one big network state in which **all nodes** with demands transmit simultaneously!

The baseline may violate the constraints of the feasible network states!

e.g. a node may receive from multiple sources at the same time
Optimal Schedule

Problem:

\textbf{SINR} changes (\textit{mobility}) while the actual transmissions take place

\=> the “optimal” solution gets quickly out-of-date

Schedule Processing Strategies:

\begin{itemize}
  \item RandomShuffle
  
  \item ShortestStateFirst
    
    Scheduling short states first should be an improvement over a random order because more states can profit from a more recent channel estimation
\end{itemize}
Simulation Model

Used simulation software:

• **openWNS**: Open source system level simulation platform

• **IMTApHy**: LTE / LTE-Advanced system level simulator

Assumptions / Limitations:

• Only demands to the nearest neighbor (e.g. Black-Box data transmission)

• Demands (amount of bits to transmit) are constant

• The solver is an omniscient entity that knows the locations, velocities, etc. of every node
Simulation Model

Assumptions:

- Earth is assumed to be a perfect sphere
- Free-Space-Pathloss used to calculate SINRs and estimate throughputs

\[ d_{\text{horizon}}(h_1, h_2) = \sqrt{2 \cdot r \cdot h_1 + h_1^2} + \sqrt{2 \cdot r \cdot h_2 + h_2^2} \]

\[ \text{Pathloss}[dB] = \begin{cases} 20 \cdot \log_{10}(d) + 20 \cdot \log_{10}(f) + 20 \cdot \log_{10} \left( \frac{4\pi}{c} \right), & d \leq d_{\text{horizon}} \\ \infty, & d > d_{\text{horizon}} \end{cases} \]
## Simulation Model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bandwidth</td>
<td>10 MHz (50 PRBs)</td>
</tr>
<tr>
<td># of Nodes</td>
<td>20</td>
</tr>
<tr>
<td>Channel Model</td>
<td>Free Space Path Loss with Radio Horizon; no Fast Fading</td>
</tr>
<tr>
<td>TX Power</td>
<td>50 W (30 dBm) per PRB</td>
</tr>
<tr>
<td>Altitude</td>
<td>10 km</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value Without Mobility</th>
<th>Value With Mobility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sim. Time</td>
<td>60 s</td>
<td>300 s 😞</td>
</tr>
<tr>
<td>Speed</td>
<td>0 km/h</td>
<td>1000 km/h</td>
</tr>
<tr>
<td>Schedule Order</td>
<td>Random</td>
<td>Random</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ShortestStateFirst</td>
</tr>
<tr>
<td>Scenario Size</td>
<td>100 x 100 km</td>
<td>500 x 500 km</td>
</tr>
<tr>
<td></td>
<td>500 x 500 km</td>
<td>1000 x 1000 km</td>
</tr>
</tbody>
</table>
Results: No Mobility
Results: No Mobility

Total Spectral Efficiency, 20 UEs, with 99% confidence intervals (100 placements)
Results: With Mobility

[Graphs showing Spectral Efficiency and Block Error Probability for Shortest State First and Random methods.]
Conclusion

- The optimal schedule obtained by Linear Programming assures high spectral efficiency
- The longer it takes to process the schedule the higher the error rate due to mobility
- Performance is increased if the schedule is processed starting with the shortest states

Outlook

- Calculate more realistic traffic demands from buffer fill levels
- Include HARQ buffers in traffic demands
- Include frequency domain
- Schedule order “Highest Relative Velocity First”
- Specify a real system using satellites for centralized scheduling
Conclusion & Outlook

Conclusion
- The optimal schedule obtained by Linear Programming assures high spectral efficiency
- The longer it takes to process the schedule the higher the error rate due to mobility
- Performance is increased if the schedule is processed starting with the shortest states

Outlook
- Calculate more realistic traffic demands from buffer fill levels
- Include HARQ buffers in traffic demands
- Include frequency domain
- Schedule order “Highest Relative Velocity First”
- Specify a real system using satellites for centralized scheduling
Conclusion
- The optimal schedule obtained by Linear Programming assures high spectral efficiency
- The longer it takes to process the schedule the higher the error rate due to mobility
- Performance is increased if the schedule is processed starting with the shortest states

Outlook
- Calculate more realistic traffic demands from buffer fill levels
- Include HARQ buffers in traffic demands
- Include frequency domain
- Schedule order “Highest Relative Velocity First”
- Specify a real system using satellites for centralized scheduling
Thank you! Questions?

www.tuhh.de