BeelP: Bee-Inspired Protocol for Routing in Mobile Ad Hoc Networks

Alexandros Giagkos

Aberystwyth University

March 16, 2010

Supervised by Dr Myra S. Wilson and Mr David E. Price
Overview

1. Introduction
2. Background
3. Design Model
4. Experimental Results
5. Conclusion
Challenges in MANETs - Motivation

Facts:

- Two major challenges: mobility and resource constraints
- Terrain may change
- Frequent link breaks

Needs:

- Adaptation
- Optimality
- Speed and efficiency
# Overview

1. **Introduction**
2. **Background**
3. **Design Model**
4. **Experimental Results**
5. **Conclusion**

---

*Alexandros Giagkos*  
Aberystwyth University  

*BeelIP: Bee-Inspired Protocol for Routing in MANETs*
Routing Protocols in MANETs

- Internet-Inspired
  - Proactive
  - Reactive
  - Hybrid

- Nature-Inspired
  - Ant Colony Optimization
  - Honeybee Inspired

Alexandros Giagkos
Aberystwyth University

BeelP: Bee-Inspired Protocol for Routing in MANETs
Honeybee foraging behaviour in Nature

"The bee’s life is like a magic well: the more you draw from it, the more it fills with water” – Prof. Karl von Frisch (1886–1982)
Honeybee foraging behaviour in Nature

"The bee’s life is like a magic well: the more you draw from it, the more it fills with water" – Prof. Karl von Frisch (1886–1982)

Quality factors that affect a bee dance:

- Is it sweet enough? Is it pure enough?
- Can it be easily obtained? Is it too far from the hive?
- Is there any improvement over time?
Overview

1. Introduction
2. Background
3. Design Model
4. Experimental Results
5. Conclusion
Artificial agents

- **Scouts**
  - Sent to discover new paths (broadcast)
  - Are propagated in order to reach their destinations
  - Introduce neighbouring nodes

- **Ackscouts**
  - Sent to acknowledge a successful path (unicast)
  - Traverse the path in reverse to reach the source node
  - Acknowledge path at intermediate and source nodes

- **Foragers**
  - Encapsulate 'real data' in the form of payload
  - Carry the sender’s *node state*
  - Collect *local reliability levels* from each pair of nodes in path
Node’s state and link information

A node’s state consists of:

- Speed (m/s)
- Energy (Joules or W*h)
- MAC Queue size (bits)

Additional information:

- Transmission delay (s)
- Signal’s strength (dBm)

Cross-layering allows accessing to various parameters from other layer protocols (MAC, PHY).

Fig. 1. A simple scouting
Evaluating paths: Local reliability level

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Signal Pow</th>
<th>Speed</th>
<th>Energy</th>
<th>Q-Delay</th>
<th>Tx-Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>min</td>
<td>1.258925e-10 W (-69 dBm)</td>
<td>0 m/s</td>
<td>0 W*h</td>
<td>0 s</td>
<td>0.0006 s</td>
</tr>
<tr>
<td>max</td>
<td>7.943282e-10 W (-61 dBm)</td>
<td>10 m/s</td>
<td>10 W*h</td>
<td>0.075 s</td>
<td>0.0120 s</td>
</tr>
</tbody>
</table>

Table 1. Local reliability parameters and scales

- Parameters have different influence on the performance
- Solution is to use a weighting system

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Signal Pow</th>
<th>Speed</th>
<th>Energy</th>
<th>Q-Delay</th>
<th>Tx-Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight (w):</td>
<td>0.40</td>
<td>0.20</td>
<td>0.20</td>
<td>0.15</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Table 2. Weighting system and factors
Evaluating paths: Local reliability level (cont.)

The local reliability level of a pair is calculated by the formula:

\[ \text{rel}_{local} = \text{pow}' \times w_{\text{pow}} + \text{speed}' \times w_{\text{speed}} + \text{energy}' \times w_{\text{energy}} + qd' \times w_{qd} + txd' \times w_{txd} \]

where \( \text{pow}' \) is the normalized value of the signal’s power, etc.

- Once calculated, it is compared with the previous obtained
- Their difference represents the improvement (or deterioration) of the pair
- It is stored both locally and into the bee forager
Evaluating paths: Global reliability level

The global reliability level of a path is calculated by the formula:

\[
rel_{\text{global}} = \sum_{n=1}^{m} (rel_{\text{local}}_{N_{n+1} \rightarrow N_n} - rel_{\text{local}}_{N_{n+1} \rightarrow N_n})
\]

where \( m \) is the total number of nodes in an numerically ordered path, and \( N_{n+1} \rightarrow N_n \) the pair of nodes with direction towards the source node \((N_1)\)

- The global reliability level of a whole path, is compared with the one from the previous flight
- Their difference represents the improvement (or deterioration) of the whole path
Evaluating paths: Global reliability level (cont.)

- Every improvement/deterioration describes **one flight only**!
- However, each incoming bee forager triggers new calculations
- Regression analysis for two variables: time \( t \) and \( rel_{\text{global}} \)

Using Pearson’s correlation coefficient, we calculate \( r \):

\[
r = \frac{\sum_{i=1}^{k} (t_i - \mu_t)(rel_{\text{global}_i} - \mu_{rel_{\text{global}}})}{\sqrt{\sum_{i=1}^{k} (t_i - \mu_t)^2} \sqrt{\sum_{i=1}^{k} (rel_{\text{global}_i} - \mu_{rel_{\text{global}}})^2}}
\]

where \( t_i \) the time of receiving \( rel_{\text{global}_i} \), \( \mu_t \) the mean of the time values, and \( k \) the number of samples (10 by default)

- A threshold -0.8 is used detect future link breaks
Simulation set up

- BeeIP (v1): first implementation in ns-2
- Compared with AODV (reactive) and DSDV (proactive)
- 20, 40, 60, 80, and 100 nodes
- 300x300 $m^2$, 500x00 $m^2$, .., and 1100x1100 $m^2$ areas
- ORiNOCO11b Wireless Card, 802.11b open range, 11 Mbit/s
- Speed range: 1m/s (walking speed) to 10m/s
- Starting energy: 36000 Joules (or 10 watt-hours)
- FTP/TCP between two nodes
- Simulation time: 600s
Results

Fig. 2. *Packet loss*

Fig. 3. *Control overhead*
Results (cont.)

Fig. 4. Packet delivery ratio

Table 3. Successfully established links during simulation (packets sent)
Overview

1. Introduction
2. Background
3. Design Model
4. Experimental Results
5. Conclusion
Conclusion and future work

Conclusion:

- Competitive approach compared to state-of-the-art protocols
- More successful packet transmissions using less path discoveries
- Ability to monitor the quality of links, and detect future breaks

Future work (BeelPv2):

- Improve weighting system (learning)
- Multiple paths for each transmitting session
- Selection algorithm between paths, based on bee dances
- Designing artificial swarms (support for UDP)
Thank you!

Happy to accept your questions, comments, hints, etc.

e-mail: aag07@aber.ac.uk
location: C57 (Llandinam)