Implementation and Performance Analysis of the Least Cluster Change (LCC) Algorithm in MANETs

Yunus Emre Keleş
2521722

The Problem & Motivation

- Mobile Ad Hoc Networks (MANETs) operate without fixed routers or base stations.
- In a "Flat Topology," finding a destination requires broadcasting requests to everyone.
    - High packet collisions.
    - Excessive battery consumption.
    - Network congestion.
- To stop the chaos, we need to organize nodes hierarchically (**Clustering**).

The Stability Challenge in Clustering

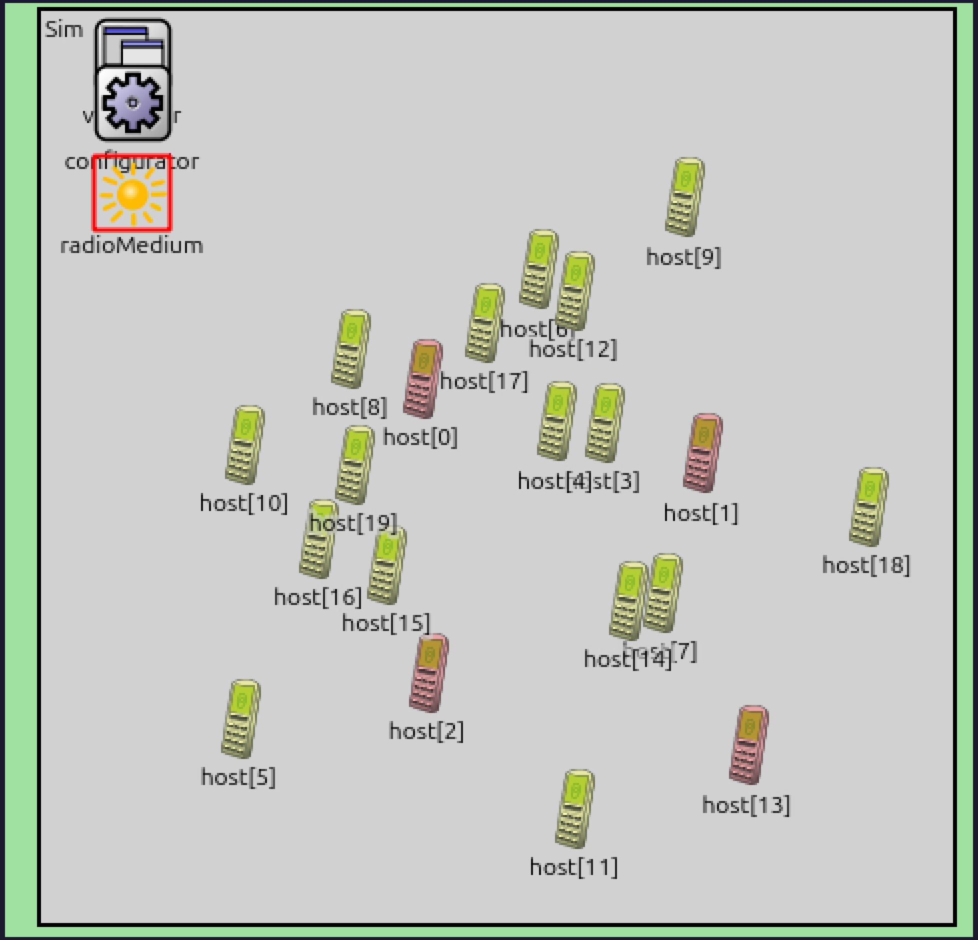
- **The Solution:** Partitioning the network into **Clusters** reduces the routing overhead.
- **The New Problem:** High Node Mobility.
- As nodes move, they constantly enter/leave transmission ranges.
- Standard algorithms trigger frequent re-elections (The **"Ripple Effect"**).
- **Project Goal:** To implement a protocol that prioritizes **Cluster Stability** over finding the mathematically "optimal" leader at every second.

The Proposed Solution: LCC Logic

- **Baseline Mechanism:** Uses the **"Lowest-ID"** criterion (Node with the lowest ID becomes the Cluster Head).
- **The LCC Innovation:** The **Hysteresis (Loyalty) Mechanism**.
- *LID Algorithm:* Switches immediately if a lower ID neighbor appears.
- *LCC Algorithm:* **Ignores** lower ID neighbors as long as the current Cluster Head is reachable.
- **Re-clustering Condition:** Only occurs when the link to the current Cluster Head is physically broken (Timeout).

Methodology & Implementation Details

- **Simulation Environment:** OMNeT++ v6.2.0 with INET Framework v4.5.4
- **Custom Development:**
  - **Application Layer:** C++ module inheriting from `inet::ApplicationBase`.
  - **Protocol Messages:** Custom `LccBeacon` (Control) and `LccData` (Payload) packets.
- **Network Calibration:**
  - **Tx Power:** Tuned to **2mW** (reduced from 20mW).
  - **Why?** To limit radio range, forcing a realistic **Multi-Hop Topology** within the 600m×600m area.



Results I - Scalability Analysis

- **Scenario:** Increasing Node Density (20→40→60 nodes).
- **Observation on Overhead:** Control traffic grows **linearly (O(N))**, proving the algorithm does not suffer from exponential message explosions.
- **Observation on Reliability (PDR):**
    - **20-40 Nodes:** High reliability (>91%).
    - **60 Nodes:** Drops to ~78% due to **MAC Layer Collisions** (Physical channel saturation, not algorithmic failure).

Results II - Mobility vs. Stability (Key Finding)

- **Scenario:** Extreme Mobility Stress Test (1m/s→25m/s).
- **The Stability Test:**
  - **Speed Increase:** 10× (from walking to driving speed).
  - **Role Change Increase:** Only ~3×.
- **Conclusion:** The **Hysteresis mechanism** successfully dampens instability. The network structure remains relatively intact even under high-speed conditions.

Results III - Physical Layer Realism

- **Comparison:** Ideal (Free Space) vs. Realistic (LogNormal Shadowing).
- **The Impact of Realism:**
- Packet Delivery Ratio (PDR) dropped to **74%** in shadowing scenarios.
- **Root Cause: "Link Flapping."**
  - Obstacles cause temporary signal drops.
  - The algorithm misinterprets these as "Link Breaks," triggering false timeouts and unnecessary re-elections.

Conclusion & Future Work

- **Conclusion:**
  - LCC effectively mitigates the "Broadcast Storm" through clustering.
  - The Hysteresis logic provides superior stability in high-mobility scenarios.
- **Future Work:**
  - **Adaptive Timers:** Adjusting "Hello Intervals" dynamically based on node velocity.
  - **Cross-Layer Design:** Making the clustering algorithm aware of MAC layer congestion.
- **Project Link:** github repo of the project