Architectural Framework for Large-Scale Multicast in Mobile Ad Hoc Networks

Ahmed Helmy
Electrical Engineering Department
University of Southern California (USC)
helmy@usc.edu
http://ceng.usc.edu/~helmy
Outline

• Motivation
• Architectural Design Requirements
• Architectural Components
  – Multicast Service Bootstrap/Rendezvous Model
  – Contact-based Small World Hierarchy Formation
• Future Work
Motivation

• Target Environment
  – *Infrastructure-less* mobile ad hoc networks (MANets)
  – MANets are self-organizing, *cooperative* networks
  – Expect common interests & sharing among nodes
  – Tens of thousands of mobile wireless nodes
  – Need scalable group-communication (multicast)

• Example applications:
  – Search and rescue (disaster relief)
  – Location-based service (tourist/visitor info, navigation)
  – Rapidly deployable remote reconnaissance and exploration missions (military, oceanography,…)
Architectural Design Requirements

• Functional Requirements (Multicast Service)
  – Dynamic creation of groups
  – Dynamic membership (nodes join/leave at will)
    • participants unknown a priori

• Robustness
  – Adaptive to link/node failure, and to mobility

• Scalability & Energy Efficiency
  – Avoids global flooding
  – Provides simple hierarchy formation
Architectural Components

• (1) Mechanisms for Bootstrapping of groups and Rendezvous of participants
• (2) Simple Hierarchy Formation Scheme
  – zone-based architecture
  – *small world* contact extensions
• (3) Multicast Routing (on-going/future work)
First Architectural Component: (1) Bootstrap and Rendezvous Mechanisms

- Problem Statement:
  - 1. Provide a *rendezvous* mechanism for group participants (senders and receivers) to *meet*
  - 2. Provide a *bootstrap* mechanism to initiate and discover new groups

- Approach:
  - View the problem as that of *resource discovery*
    - Design a scalable efficient scheme to *search* for and discover resources (information about groups and senders)
Alternative Multicast Rendezvous Designs

- Broadcast-and-prune (flooding of packets or sender advertisements)
- Expanding ring search (scoped flooding)
- Center-based tree (rendezvous-point)
- Hierarchical (e.g., domain-based, needs AS hierarchy)
The Multicast Model

• New Multicast Model: sender push, servers cache, receivers pull

• Where are the servers located? And how do participants find these servers?
  – Do we have to resort to global flooding?
  – Can we maintain this info. in well-known rendezvous node? (A node can move or fail, so bootstrapping will be hard)
  – Use multiple nodes within a known Rendezvous Region (RR) (Still need bootstrap)
Search and Resource Discovery Scheme

- The geographic topology is divided into *rendezvous regions* (RRs)
- The multicast space is divided into *group prefixes* (Gprefix)
- Mapping between $Gprefix \leftrightarrow RR$ is provided to all nodes (e.g., through a small table, or could also be algorithmic)
Assumptions

• Nodes know their (approximate location) [using GPS or GPS-less schemes]
• Nodes capable of geographic routing (as in location-aided routing (LAR) and geocast)
Get location \((x,y)\)

\[
\begin{array}{c|c}
RR_1 & Gprefix_1 \\
RR_2 & Gprefix_2 \\
\vdots & \vdots \\
RR_n & Gprefix_n \\
\end{array}
\]

\((x,y) \in RR_i \leftrightarrow Gprefix_i\)

\(SDS: \text{sender discovery server}\)
<table>
<thead>
<tr>
<th>RR1</th>
<th>RR2</th>
<th>RR3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><em>RRleave</em></td>
</tr>
<tr>
<td>RR4</td>
<td>RR5</td>
<td>RR6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RR7</td>
<td>RR8</td>
<td>RR9</td>
</tr>
</tbody>
</table>
Sender to group $G$

<table>
<thead>
<tr>
<th>$RR_1$</th>
<th>$RR_2$</th>
<th>$RR_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$RR_4$</td>
<td>$RR_5$</td>
<td>$RR_6$</td>
</tr>
<tr>
<td>$RR_7$</td>
<td>$RR_8$</td>
<td>$RR_9$</td>
</tr>
</tbody>
</table>

$G \in G_{prefix_i} \leftrightarrow RR_i$
geo-cast

RR1  RR2  RR3
RR4  RR5  RR6
RR7  RR8  RR9

contact

Ahmed Helmy, USC
Bootstrap Mechanism

• The goal:
  – To create sessions/groups dynamically
  – To announce these groups
  – To enable interested potential participants to obtain information about new groups

• Approach:
  – Designate a well-known group (WKG) address for the ‘session/group announcement group’ as a bootstrap group.
  – As any other group, this group is also supported by RR
Bootstrap (initiating, and knowing about, groups)

**WKG**: Well-known Group (a bootstrap group for initiating further groups)
Bootstrap (initiating, and knowing about, groups)

**GI: Group Initiator**
Bootstrap (initiating, and knowing about, groups)

\[
\begin{array}{c|c|c}
RR1 & RR2 & RR3 \\
\hline
RR4 & RR5 & RR6 \\
\hline
RR7 & RR8 & RR9 \\
\end{array}
\]

\[
\begin{array}{c|c}
RR_i & Gprefix_i \\
\hline
RR_1 & Gprefix_1 \\
RR_2 & Gprefix_2 \\
\ldots & \ldots \\
RR_n & Gprefix_n \\
\end{array}
\]

\[WKG \in Gprefix_i \leftrightarrow RR_i\]

Multicast Member
Bootstrap (initiating, and knowing about, groups)
Performance Discussion

• What is the extent of the search for servers?
• We expect group initiators to choose groups that map to nearby RRs
• We expect popularity of groups in certain locations (e.g., for location-based services)
• These factors increase the chances of localizing the search for servers
• More evaluation needed (on-going work)
Architectural Components

• (1) Mechanisms for Bootstrapping of groups and Rendezvous of participants

• (2) Simple Hierarchy Formation Scheme
  – zone-based architecture
  – *small world* contact extensions
Second Architectural Component: (2) Simple Hierarchy formation

• Problem Statement:
  – To discover group/sender servers with reduced overhead (than in flooding or pure geocast)

• Approach:
  – Design a simple hierarchical architecture
    • Complex mechanisms incur lots of overhead with mobility and network dynamics
Alternative Hierarchical Designs

• Cluster-based hierarchy using cluster heads
  – Cluster heads are elected dynamically
  – Traffic funnels through cluster head

• Landmark Hierarchy
  – Landmarks advertised and used for mgmt/addressing
  – Re-configuration needed if landmarks move
    • Needs adoption/promotion/demotion mechanisms

• Zone-based Routing
  – Each node has its own neighboring zone
  – Hybrid routing:
    • pro-active within zone, re-active out of zone
Contact-based ‘*small world*’ Hierarchy

- Neighboring zone knowledge
  - A node knows routes to neighbors up to $R$ hops away ($R$ is the zone radius) pro-actively
  - A node maintains contacts to increase its view of the network

- Simplicity & Efficiency
  - No elections of cluster head or landmarks
  - Mobility does not lead to major re-configuration
  - Uses the concept of *Small World Graphs* (*six degrees of separation*)
How to Create a Small World

[Watts 98]
Clustering

- Clustering Coefficient \((C)\) captures how many of a node’s neighbors are also neighbors for each other

Node \(i\) has \(n\) neighbors

\[
\text{Clustering} = \frac{n(n-1)}{2} = 6
\]

Clustering Coefficient for node \(i\) \((Ci)\) = \(\frac{3}{6} = 0.5\)

(a) Given network

(b) Fully connected network

Ahmed Helmy, USC
Small World Characteristics

Desirable Region

probability of re-wiring (p)

Clustering (C)
Path Length

[Watts 98]

Ahmed Helmy, USC
Taking Advantage of Mobility

• Small world graphs
  – Adding a small number of random short-cuts drastically reduces the average path length of graph
  – For our case: choosing random nodes leads to unpredictable overhead.

• Choosing contacts:
  • Choose nodes whose characteristics you know (your neighbors).
  • When they move, they will have a network view with less overlap.
  • The resulting graph may tend to a small world graph.
• Example of zoning:
  - Zone for center node $C$ is shown (with radius $R$). Border nodes are numbered (1-7). Nodes 1, 3 and 6 are moving/drifting out of zone.
• \( C \) stays in \textit{contact} with the drifting nodes using low overhead, which enables it to obtain better network coverage.
Choosing Contacts

• A node chooses its contacts with probability $p$ proportional to
  – Energy estimates $E_{est}$ of the node and the contact,
  – Their relative stability $S_{est}$, and
  – Activity level of the node $A_{est}$ measured as rate of discovery requests.

• Also, $p$ is inversely proportional to the number of zone contacts $Z_{est}$.  
  \[ p \propto E_{est} S_{est} A_{est} / Z_{est} \]
Choosing Contacts (contd.)

• $E_{est}$: Energy estimate for the lifetime of the node itself and the contact node
  – $E_{left}$: Energy left
  – $\Delta E$: Energy drainage rate
  – $E_{est} \propto (E_{left} / \Delta E)_{node} (E_{left} / \Delta E)_{contact}$

• $S_{est}$: Relative stability (possible presentation)
  – $(\alpha, t)$ model: estimates probability that nodes within range at $t_0$ will be in range at $t$

  – Received Power model:
    • $RxdPwr/TxdPwr = 1/d^n$; where $n=2$ or $4$
    • $RxdPwr_{new}/RxdPwr_{old} << 1$ then nodes moving away
Conclusion

• Designed architectural framework to support multicast service in large-scale Ad Hoc networks

• Introduced the concept of *rendezvous regions (RRs)* as a base for bootstrapping and rendezvous mechanisms

• Introduced a new contact-based zone hierarchy based on small world concepts
On-going and Future Work

• Detail and evaluate contact selection and maintenance mechanisms

• Investigate contact-based hierarchy parameters
  – zone radius (R), number of contacts, max contact distance, depth of contact query
  – performance evaluation in terms of degrees of separation, selection/maintenance overhead, query response delay
Future Work (contd.)

- Study characteristics of the graphs resulting from the contact-based hierarchy
- Evaluate the efficiency of the search for group/sender servers under various scenarios
- Design the multicast routing component
  - Mesh-based with on-demand activation of alternative paths
Popularity overhead
Popularity-based optimization: Local SDS election based on group popularity