A Reliable Multicast Protocol for Wireless Mobile Multihop Ad Hoc Networks

Selected from CCNC 2004

Present by Lin Yu-Chen

9/17/2004

Outline

- Introduction
- Problem Definition and Motivation
- Proposed Solution
- Simulations and Performance Evaluation
- Conclusions and Discussions

Introduction

- In a typical ad hoc network , hosts work in groups to carry out a given task .
- Some multicast applications , such as audio/video conferencing , can tolerate packet error and/or loss .
- Other applications , such as one-to many file transfer and military applications do not .

Introduction

- The characteristics of wireless mobile multihop ad hoc networks make reliable multicasting extremely challenging .
  - Lack of network infrastructure
  - Dynamic network topology
  - Scarce bandwidth and variable link capacity
  - High error rates

Problem Definition and Motivation

- A reliable multicasting guarantees the eventual delivery of all the multicast data to all the multicast group members .
- In resent year , several reliable multicast protocol have been proposed , but they still have some problems .

Problem Definition and Motivation

- It is suitable to multicast over a dense multicast group , but inadequate for sparse group.
- It is inoperable in nonclustered ad hoc networks.
- It restricts that the underlying unicast routing protocol is reliable

Problem Definition and Motivation

- Observing that a certain class of application will be almost infeasible on ad hoc networks without reliable multicasting .
- We need to have a reliable multicast protocol that requires minimal support from underlying network protocols and does not depend on an underlying clustering protocol .

Proposed Solution

- ReMHoc is a distributed receiver-initiated NACK-based reliable multicast protocol .
  - REQUEST
  - REPAIR
  - HeartBeat(HB)

Proposed Solution (REQUEST)

- When a receiver detects a missing DATA packet , it should request its retransmission by multicasting a negative acknowledgement (REQUEST)
- Request implosion
  - A great number of REQUEST are generated to request the retransmission of the same DATA packet
- Request timers
  - Whenever a receiver detects a missing DATA packet , it sets a request timer for a random interval.

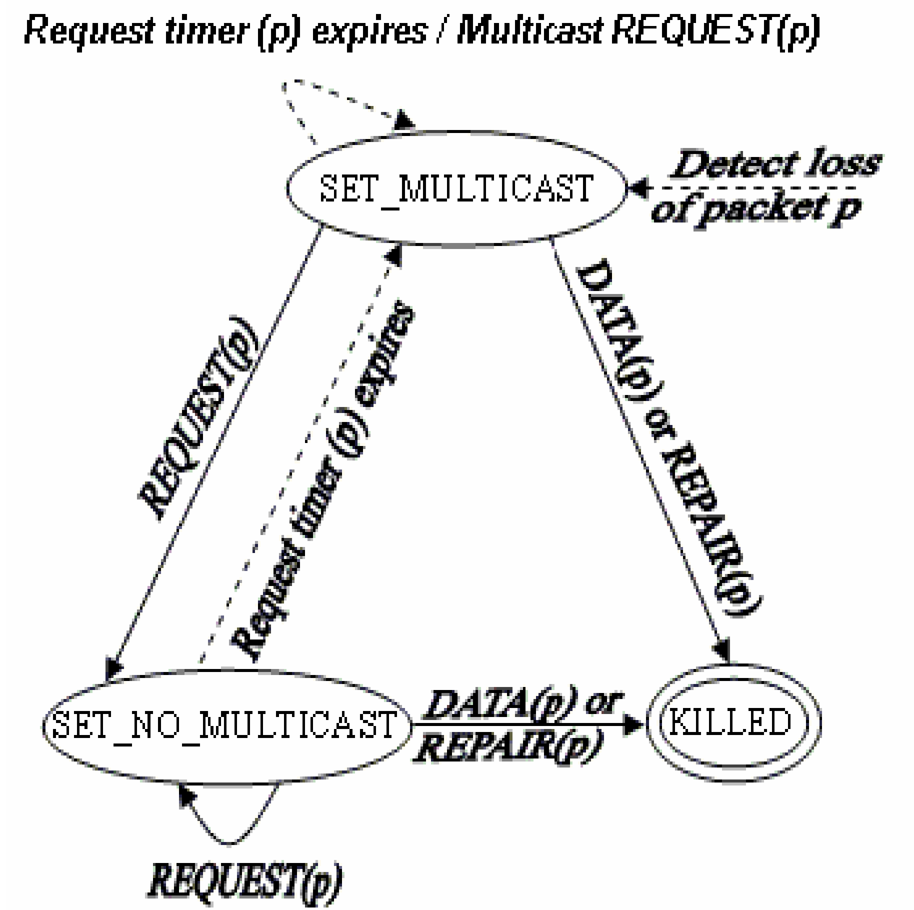
Proposed Solution (REPAIR)

- If the member receiving a REQUEST has a cached copy of the requested DATA packet , it can respond by multicasting the cached copy (REPAIR)
- Repair implosion
- Repair timers

Proposed Solution (HB)

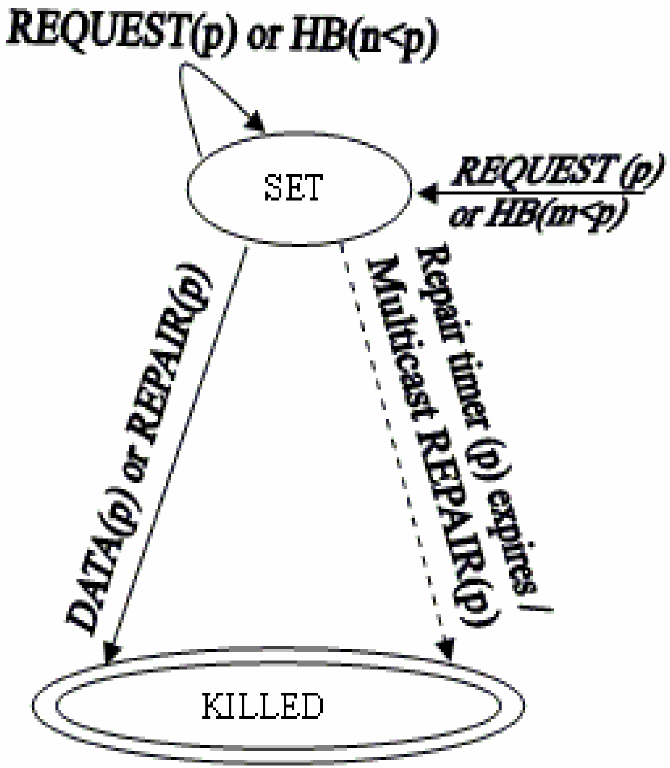
- As long as a receiver has not received that END packet , it sets a heartbeat timer. When the timer expires , a receiver multicasts a heartbeat (HB)
- Any group member which receives an HB packet , tries to find in its cache a copy of any DATA packet whose sequence number is higher than that indicated in HB packet.
- This mechanism ensures that all receivers will receive the last data packet and that receivers which do not receive any data packets for a “long” period can keep pace with other receivers.

State Transition Diagram (REQUEST)



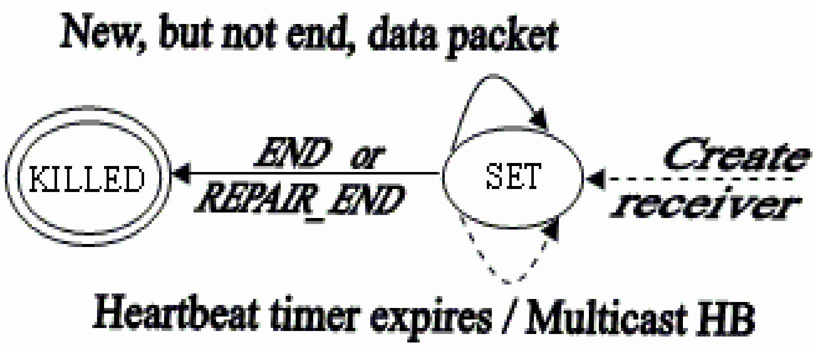
- The request timer interval is made dependent on the hop count between the receiver and source.
- It is desirable to make the request timer interval for a missing packet gets longer if the receiver has already sent REQUESTs for it before.

State Transition Diagram (REPAIR)



- It is desirable to make the one who is likely to time out first be the one that is closest to the requestor.

State Transition Diagram (HB)



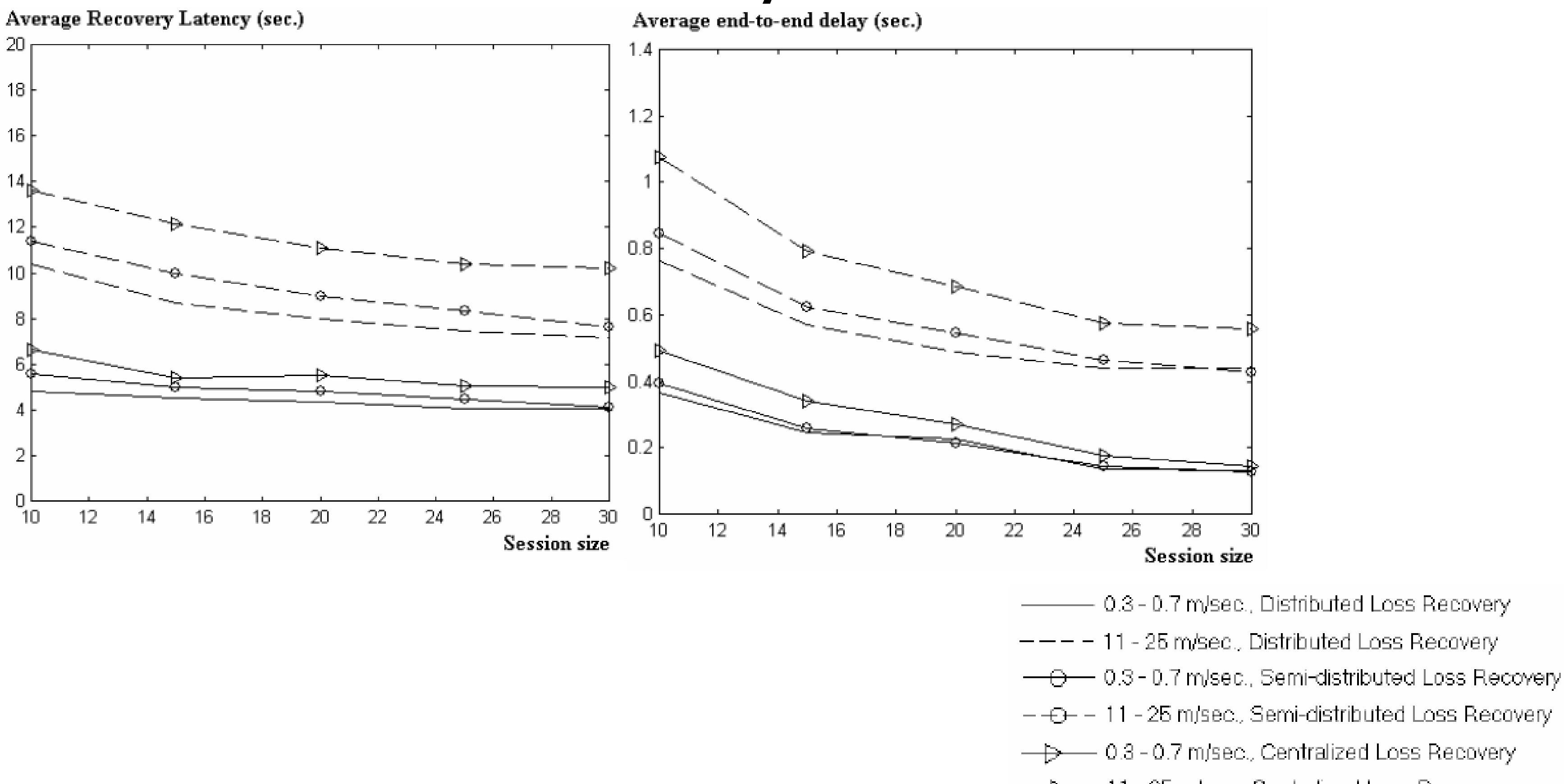
- A receiver may not receive new data packets for a long period because the rate of arrival of data packets is low.
- The ReMHoc make the heartbeat timer interval a multiple of the packet interarrival time.

Simulations and Performance Evaluation

- Performance evaluation criteria can be stated as follows
  - Percentage of REQUESTs and HBs
  - Percentage of REPAIRs
  - Average recovery latency
  - Average end-to end delay
  - Overhead percentage

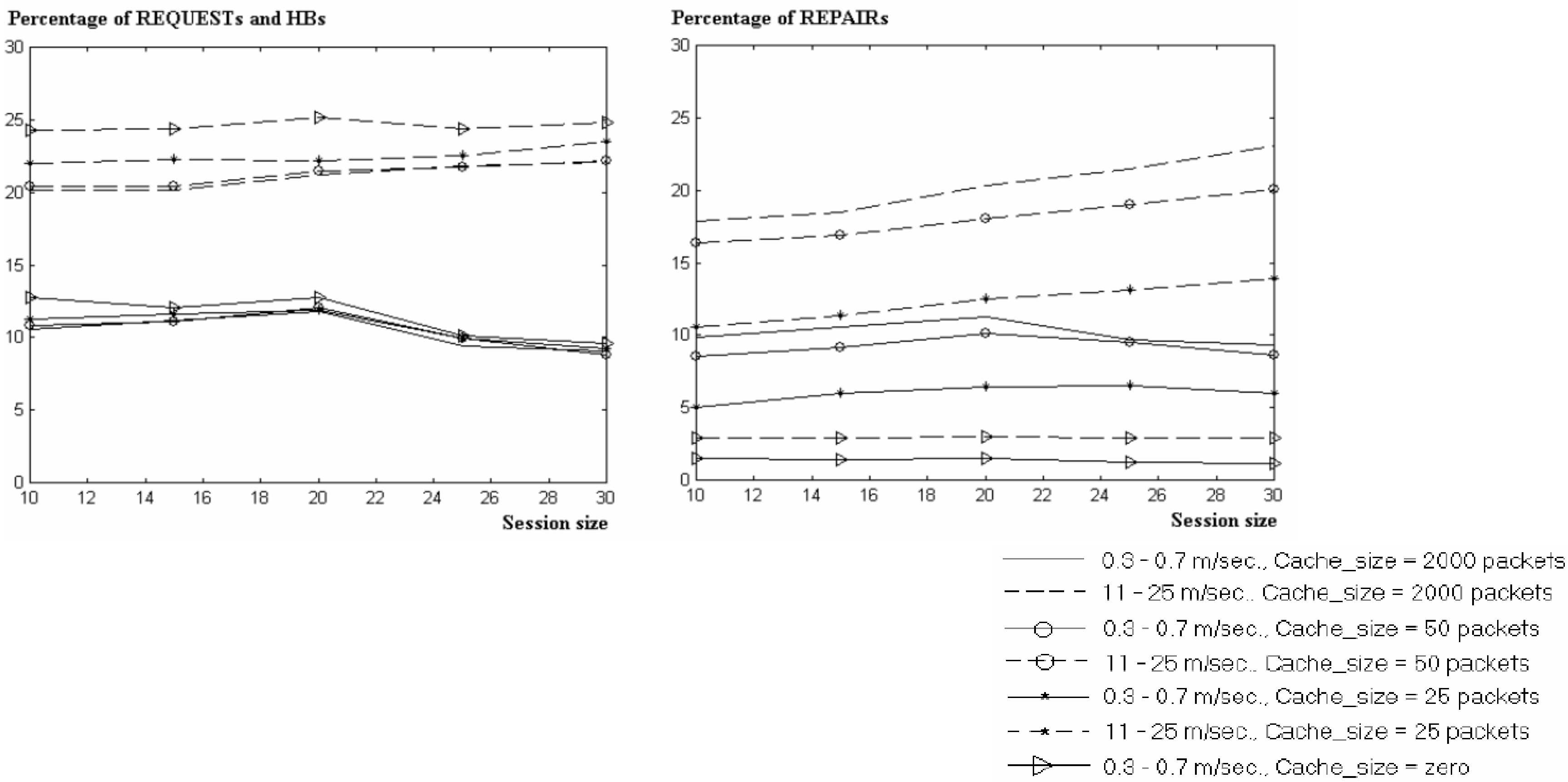
Simulations and Performance Evaluation

- Effect of mobility and session size



Simulations and Performance Evaluation

- Effect of the receiver's cache size



Conclusions and Discussions

- ReMHoc is scalable with the number of multicast group members (session size)
- The centralized loss recovery has higher overhead percentage , higher average recovery latency and higher average end-to-end delay than distributed loss recovery.
- Can this method extend to multi-channel ad hoc networks?