Message Ferrying

A Report Submitted in Partial Fulfillment
of the Requirements for ENSC 427

Group 9
Dan Hendry (danh@sfu.ca), 301133878
Yazan Shehadeh (ysa5@sfu.ca), 301028275
Timbo Yuen (tty2@sfu.ca), 301023080

Simon Fraser University
School of Engineering Science

March 24, 2010.
Course Instructor: Professor Ljiljana Trajkovic

Abstract

Machine to machine communication has long been considered a feature of the next technological age. Networking options available today are either too expensive or cumbersome to justify the information they are able to provide (such as dedicated wired Ethernet or cellular data modems), or do not provide uniform connectivity (such as WiFi). An alternate solution is to use ad-hoc mesh networking. Such networks however, require every node to be connect to another and fail when the network is sparse or becomes partitioned. Message ferrying is a technique which uses physical mobile devices, known as message ferries, as data transport mechanisms between disconnected network nodes or partitioned subnetworks. This project will implement and simulate a message ferrying algorithm applicable to a specialized remote sensor network in which a central repository maintains current sensor state.

References for interim report: [1] [2] [3] [4] [5] [6] [7] [8]

Contents

List of Tables

List of Figures

1

Introduction

1.1 Background

1.1.1 Mobile Ad Hoc Networks

A mobile ad hoc network (MANET), is a self-configuring mesh network of mobile devices connected by wireless links. These mobile devices are free to move independently in any direction and it acts as a router, where it must forward traffic unrelated to its own use.

1.1.2 Partitioned Networks

Partitioned networks are networks with no single hop or multiple hop route between some or even all node pairs. In a partitioned network, nodes may remain fully disconnected or they may *cluster*, forming subnetworks in which all nodes are connected. All current used routing algorithms used in MANETs fail in the presence of partitioning [3].

1.1.3 Delay Tolerant Networks

A delay tolerant network is one in which routing strategies and applications must tolerate significant delays delivering packets. This delay may range from a few minutes up to hours or even days.

1.1.4 Message Ferrying & Store-Carry-Forward Routing

Message ferrying is a technique where mobile nodes in a MANET buffer data and physically carry it between nodes which are unable to communicate [1]. Store-carry-forward routing is a strategy which makes use of, typically, known or assigned trajectories of these mobile nodes, known as message ferries. Some messages are dropped if no route to the destination can be found.

1.2 Motivations and Potential Applications

With the increasing numbers of mobile devices today, such as smartphones, laptops, tablets, netbooks, and more, there are many devices which could be potentially used as message ferries. This project proposes one way to make use of the significant amount of technology we transport with us on a daily basis. A message ferrying network could potentially transport small amounts of data over large distances essentially for free. Beyond the use of message ferrying in remote sensor networks, discussed throughout this report, other applications of this technique might include tracking road traffic conditions, in-house utility management, automation for home devices, industrial monitoring, robot to robot communication and more.

1.3 Project Goals

Message ferrying has typically been examined within the context of improving throughput, reducing delay and increasing reliability within an ad hoc network. Due to the complexity of incorporating message ferrying into existing ad hoc and MANET routing algorithms, this project will focus on a network in which data is transported strictly using ferries. No clustering of network nodes and routing within subnetworks will be considered. Surprisingly, very little research has been found for a network with these characteristics. The goals of this project may be listed as follows.

- Design and implement a message ferrying algorithm.
- Simulate this algorithm in a highly partitioned network without node clustering or subnetworks.
- Evaluate the network considering topology and the message ferrying algorithm.
- Examine the impact of node density and ferry count on the network.

2

Project Premise & Model Design

Any application running over a message ferrying network must have the following characteristics.

Delay Tolerance: Since data is transported by a physical device, significant delays of minutes to hours must be expected.

Loss Tolerance: Given that ferries have limited memory, loss of data must be expected.

Small and Independent Messages: Following from the limited memory capacity of ferries and the high probability of packet loss, a reliable method for segmentation and reassembly of messages should not be expected. Applications should limit the size of messages such that the can be transmitted in their entirety using one network packet. (See section 4.2.1 for future work)

Given these criteria, a message ferrying network is unsuitable for many classic networking applications including web browsing, real-time voice or text communication and file transfer. As such, a very specialized 'state monitoring' network designed for non-critical monitoring of remote sensors is considered.

2.1 State Monitoring Network

The general premise for this project consists of a network containing numerous, uniquely identifiable source nodes. Each source node has a limited number of properties, in the form of key/value pairs, specifying a property name (the key) and its current value. Properties may change overtime and each change defines a new state for the source node. A temperature sensor for example, might support a 'temperature' property, the value of which is the current temperature updated every hour. Properties do not have to contain a single value and each may be as large as the payload limit of network packets.

The network and message ferrying algorithm is designed to synchronize a central repository with the current state of every source node. Only the most recent state (or most recent value) for each property is important, not the history of how that property has changed. This limits the number of packets which can exist in the network as only the most recent update must be reported. The message ferries collect data from source nodes when they are in range and transport it to the central repository. The central repository is assumed to be a server connected to the Internet. Ferries pass updates they have collected from source nodes to special gateway nodes. These gateway nodes are then responsible for using a reliable delivery mechanism over a standard IP network to update the central repository.

Insert image showing this process.

2.2 Network Elements

This section describes the elements present in the network.

2.2.1 Physical Network Entities

The network is comprised of four types of nodes.

Source Node: Static nodes in the network which have a set of properties (key/value pairs). After a property of a source changes, known as a state change, it attempt to notify the central repository by transferring update packets to message ferries. A source node could be, for example, a remote temperature sensor.

Message Ferry: A mobile node which collects updates from source nodes when they are in range. Message ferries store update packets from source nodes within a buffer. When in range, these update packets are forwarded to gateway nodes. A source node could be, for example, a specially equipt cell phone or a small computer attached to a vehicle.

Gateway: Gateway nodes download update packets form message ferries and forward them to the central repository over the Internet. Transfer of updates between the gateway and central repository is assumed to be reliable.

Central Repository: The central repository is a server and the final destination for updates sent from source nodes. It maintains a list of the current state of all source nodes. There is only one central repository node.

2.2.2 Inter-Node Communication

Nodes communicate with each other using the ZigBee (802.15.4) MAC layer at 2.4 GHz.

2.2.3 Packets

The ZigBee MAC layer uses special packets to establish and acknowledge inter-node communication. At the network layer, two types of packets are used as follows:

Update Packet: Update packets are generated by source nodes when a message ferry is in range and their state has changed. Source nodes will continue to generate update packets until the state change is acknowledged by the central repository (using an acknowledgment packet).

Acknowledgment Packet: Packets sent from the central repository to source nodes acknowledging that the state update has been recorded.

Interim Report: There is more to this which I have not included.

3

Simulation

Various scenarios are to be tested as discussed in section 3.1.

When simulating each scenario, the following metrics will be measured:

1. Time to update state in central repository (delay)

2. Variation in time to update state (delay jitter)

3. Number of state updates lost. Roughly corresponds to packet loss.

4. Memory utilization and packet dropping threshold within ferries.

3.1 Scenarios

3.1.1 Scenario Considerations

The following should be considered when designing scenarios

- Number of sources to ferries to gateways (various ratios)

- Speed and trajectories of ferries (random vs set path)

- Rate of source node state changes

- Buffer size of ferries and size of property values (affects packet sizes)

- Distances and distributions of ferries and gateways

3.1.2 Scenario 1: x

What is the scenario features. What specifically is it trying to test or measure.

List other scenarios

3.2 Results

Show result tables and graphs. Discuss implications.

4

Conclusion

4.1 Results

An general discussion of results. Include a discussion on the feasibility of a practical implementation and what adoption threshold would be needed.

4.2 Future Work

General discussion on future possibilities.

4.2.1 Fountain Codes

An assumption of the scenario was that application messages can be transmitted in a single network packet. A possible solution overcome this limitation without using an ARQ scheme (poorly suited to a delay tolerant network) is the use of fountain error correcting codes (aka rateless erasure codes). See `http://en.wikipedia.org/wiki/Fountain_codes`.

References

[1] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, College of Computing, Georgia Institute of Technology, Atlanta, Georgia 30332, 2004.

[2] Y. Chen, W. Zhao, M. Ammar, and E. Zegura, "Hybrid routing in clustered DTNs with message ferrying," in *Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, College of Computing, Georgia Institute of Technology, Atlanta, GA, 30332, 2007.

[3] R. Patra, K. Fall, and S. Jain, "Routing in a delay tolerant network," in *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, 2004.

[4] J. A. Davis, A. H. Fagg, and B. N. Levine, "Wearable Computers as Packet Transport Mechanisms in Highly-Partitioned Ad-Hoc Networks," in *In Proceedings of the 5th IEEE international Symposium on Wearable Computers*, IEEE Computer Society, Washington, DC, 141, October 2001.

[5] R. Patra, K. Fall, and S. Jain, "Comparison of QoS routing approaches for intermittently connected Mobile Ad Hoc Networks," in *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, 2004.

[6] R.C.Suganthe and P.Balasubramanie, "Efficient Routing For Intermittently Connected Mobile Ad hoc Network," in *IJCSNS International Journal of Com-*

puter Science and Network Security, 184 VOL.8 No.11, November 2008, 2008, http://paper.ijcsns.org/07_book/200811/20081126.pdf.

[7] T. Massey, "Message Ferry Architecture and Implementation," in *Georgia Institute of Technology*, 2004, http://www.cs.ucla.edu/~tmassey/MF_Master_Proj.pdf.

[8] S.-L. Wu and Y.-C. Tseng, *Wireless Ad Hoc Networking.* Auerbach Publications, 2007, pp. 439–459.

Appendix A

Code

A.1 Message Ferrying

Coding goes here Remember to comment code and explain process

A.2 MAC and other stuff here

Coding goes here Remember to comment code and explain process