This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

Sensor Data Collection Through Gateways in a Highly Mobile Mesh Network

Andrew Jenkins, Daniel Henkel, Timothy X Brown
Electrical and Computer Engineering
University of Colorado at Boulder
Email: {jenkinaj, henk, timxb}@colorado.edu

***Abstract*— Widely distributed sensors must discover paths back to data collection points possibly through sparsely connected and mobile networks. Current addressing and service discovery schemes in mobile networks are not well-suited to multihop disconnected networks. This paper describes an architecture and protocol for sensor data collection through highly mobile ad-hoc network (MANET) that may never experience end-to-end connectivity. Special gateway nodes are described which are responsible for intelligently routing messages to their intended destination(s). These gateway nodes qualify their links and announce their status to the MANET, a simple approach to service discovery that is effective in this implementation. The protocol is implemented and tested in a laboratory and outdoor environment.**

I. Introduction

Traditional data networks assume bi-directional, end-to-end connectivity for reliable message delivery. Mobile computing prompted development of protocols such as Mobile IP, which address mobility, but still rely on a well-connected network. This paper considers a very mobile ad-hoc sensor network. The goal of this network is to reliably deliver sensor data to sensor monitoring stations (SMS). Our network attempts to provide end-to-end services in spite of high node mobility and a lack of end-to-end connectivity at any instant.

The network under consideration is heterogeneous across node and link capabilities. Some nodes may be fixed, and some nodes are unmanned aircraft that have very high mobility. Sensor nodes are assumed to have limited network capabilities, while sensor monitoring stations may be much less limited, even having access to permanent mains power and multiple points of presence. The network links vary, and may include wireless LANs such as 802.11, wired ethernet (802.3), data over CDMA, satellite links, and well-connected backbone or global Internet IP networks. These may have varying data rates and reliability, and be only intermittently available.

There are two critical forms of communication over this network. *Event* communication consists of messages (such as sensor data) from the sensors to the data collection nodes, or SMS. *Command* communication consists of messages from the SMS to the sensors. Event and command communication must be reliable; however, necessary delays imposed by the partitioned nature of the network are acceptable. The structure of the network is presented in Fig. 1 which shows three distinct network domains: sensor-to-MANET, the MANET, and MANET-to-SMS. The network elements will be described in detail in the next section.

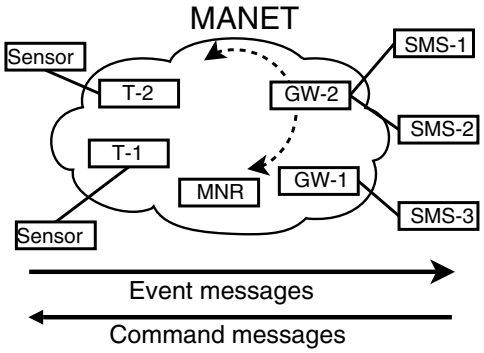

Fig. 1. Sensor data collection over a MANET, with gateways to other networks

This system poses three main challenges. First is sensor and SMS discovery. The sensors cannot be configured a priori with SMS addresses. In fact, the sensor-to-MANET, the MANET, and the MANET-to-SMS connectivity may all reside in incompatible network and address domains. Conversely, the SMS and MANET may not know a priori the number or location of the sensors. A second challenge is network dynamics. Links in the network can be dynamic. SMS may be connected through intermittent links. Nodes in the MANET may be mobile and never provide concurrent end-to-end connectivity. The SMS and sensors may dynamically join and leave the network. Third, the multicast function: A multicast tree must be formed across the different networks to deliver events to SMS. It must handle the network dynamics and the fact that neither sender nor receiver has access to the end-to-end network topology.

To address these challenges we develop a gateway concept that manages the sensor and SMS discovery, multicasting, and the network dynamics. The concept specifically enables these gateways to act as mobile data ferries [1] that can physically carry data between sensors and SMS. The gateway is supported by a reliable packet forwarding protocol which can fragment and deliver larger messages across intermittent network connectivity. We implement a hybrid between non-delay-tolerant wireless sensor networks [2] utilizing ad-hoc routing [3], and delay-tolerant networks [4], leveraging the mobility of gateway nodes to deliver messages over distances

1525-3511/07/$25.00 ©2007 IEEE

of several km.

Other researchers have considered different elements to this problem. Mobile IP [5] requires a gateway-like home agent that forwards messages to nodes whose addresses may change. Dynamic DNS [6] provides dynamic service discovery of mobile nodes moving across networks. This requires an always-available home-agent like mobile IP and requires the end host to update routes if the mobile node address changes. Our approach has a more robust dynamic gateway approach where any one of several gateway nodes can be active at a time. Further, intermediate nodes are defined which can relieve end nodes of much of the burden of managing the network dynamics. The group-based service discovery (GSD) [7] for MANETs is similar to our gateway discovery protocol. However our protocol implements a reliable service redirect so that if a gateway no longer can provide a SMS forwarding service it can forward packets to alternative gateways.

Reliable UDP (RUDP) [8] implements many of the features of our reliable packet forwarding. However, RUDP requires nodes to first establish a connection before communicating messages while our approach either does not require a connection setup or allows the connection setup to arrive in any order along with message packets.

The Stream Control Transmission Protocol (SCTP) [9], [10] is designed for hosts with multiple network addresses. If the SMS are well connected to each other, SCTP would enable events delivered to any one SMS to be forwarded to the other SMS so that the SMS form a single SCTP endpoint in order to implement a type of reliable multicast. Unfortunately, different SMS may represent different data customers who may not necessarily cooperate or know of each other. SCTP also provides fragmentation and in order delivery of larger messages like our protocol but requires concurrent connectivity. The application of SCTP to mobile networks is explored in [11], but implementations of mobility characteristics (required to handle new, departed, or moving nodes) are limited.

Ouyang et al. [12] describe two reliable multicast protocols. These require that the multicast senders have knowledge of the receivers when they emit packets and membership is established by communicating join packets to the source. In our implementation the multicasting takes place at two levels internal to the network each level taking care of the local packet copying and forwarding without needing to know the full multicast tree.

The closest to our approach is the delay tolerant networking (DTN) architecture [4], [13]. This defines a notion of message bundles that are forwarded in stages between so-called custody transfer points. However, the general problem of routing through intermittent connectivity remains unsolved. Our approach focuses on a specific limited type of communication (events and commands) for this application that enables us to provide a robust routing solution. Also, our custody notion applies only to packets and not to entire bundles. As will be seen this enables more efficient packet forwarding. One downside to our approach is that we impose a specific three-stage topology and require certain prior knowledge (e.g. gateways must know the address of the SMS they serve) which limits its applicability. We also do not provide end-to-end reliability as a network primitive since acknowledgements to sensors by SMS may not always be desirable for a multicast connection. As a result there are scenarios (e.g. loss of a custody node) which could cause data to never be delivered. Future work will address these deficiencies.

Our network attempts to provide end-to-end services in spite of high node mobility and a lack of end-to-end connectivity at any instant. This research shows how a MANET can function to relay user messages over a large geographic area as a service to other networks; we insulate both the sensor and sensor monitoring nodes from the internals of the network, yet provide multicast and unicast services. Our network uses self-organizing gateway nodes that can physically relay messages when the network becomes partitioned. The next section describes the protocol architecture in detail. Section III describes our implementation and basic performance. Section IV explores how gateway nodes can physically relay messages. Section V describes how the network mitigates various types of faults. The paper concludes with a discussion and description of future work.

II. Architecture

The network consists of different classes of nodes, as shown in Fig. 1. The gateways may be highly mobile nodes, typically onboard unmanned aircraft (UA) or ground-based vehicles. They communicate with SMS through links external to the MANET that support IP. Ethernet, 802.11 wireless LAN, CDMA, and satellite connections have been tested within the architecture. Each gateway may have connectivity to any number of SMS, or none at all. As well, each SMS may be served by multiple gateways. This many-to-many association presents considerable challenges in promoting reliability.

In our network, special *terminus* nodes are attached to sensors. A terminus participates in the MANET, and also presents a second interface to sensors. Terminus nodes are intermediary between the MANET and the sensors, because it is assumed that sensors may have limited networking ability and incapable of implementing our routing protocols. Sensors can communicate to terminus over links as simple as serial line IP (SLIP); wired ethernet is used in the implementation under consideration. More capable sensors can implement the terminus function directly.

As well, nodes called MNRs (mesh network radios) may be attached to the network. These participate in the MANET protocols along with the terminus and gateway, but are neither attached to sensors (as terminus are), or attached to SMS (as gateways are). These nodes can be used to extended coverage of the MANET. Though envisioned as a typical 802.11-based wireless MANET, the so-called MANET could be any distinct network domain.

With this overview of the network components, the rest of this section describes the protocol elements that comprise the reliable event and sensor communication.

A. SMS to Gateway Heartbeat Protocol

In order to permit characterization of the link between a gateway and an SMS, we implement a heartbeat protocol. This consists of a UDP/IP packet with specified source and destination port (P_{HB}). The datagram contains a 32-bit sequence number and a 16-bit valid time. The sequence number is used for identification purposes in case packets arrive out-of-order. The SMS responds with a similar UDP/IP packet (called heartbeat response) containing the same sequence number, and a valid time that is less than or equal to the valid time of the heartbeat datagram. When the gateway receives this response, it considers the link between gateway and SMS to be valid for the valid time. The valid time of both the original heartbeat packet and the heartbeat response packet are configurable; if an SMS is scheduled to be shut down, it can begin reducing the valid time of its heartbeat responses to gracefully deactivate the link.

The period at which heartbeats are emitted from the gateway is configurable, but should be more frequent than the valid time. This is because the links between gateway and SMS are unreliable, so only a fraction of the heartbeat and heartbeat response messages may be delivered. The valid time does not indicate the length of time for which a link is guaranteed to be valid; rather, it is an interval that satisfies the assumption that if the gateway has not received a response for the valid time, it is unlikely it will be connected to an SMS in the future, and nodes should attempt to deliver messages through another gateway.

If a gateway has received unexpired heartbeat responses from its potential SMS, it is considered to be a valid gateway. Valid gateways announce their status to the MANET, permitting nodes to route messages through these gateways.

Gateways that do not receive any heartbeat responses do not announce themselves as valid gateways to the MANET. However, like all MANET nodes, they still act as an ordinary ad hoc node, and can serve as an intermediate hop in a multihop path on the MANET, functioning as MANET range extenders.

B. Equivalence Classes

We define gateway *equivalence classes*. The equivalence relation asserts that each valid member in a class has connections to the same set of SMS. Equivalence classes partition the set of gateways. The terminus attempts to forward messages to one gateway in each equivalence class, and gateways can forward messages that they cannot deliver to another gateway in the same equivalence class to be delivered. Equivalence classes are defined as a configuration parameter on the gateways.

C. Advertisement Protocol

Once a gateway has established links to its SMS, it must advertise this status to the MANET. Remembering that the MANET has a rapidly changing topology in which no node can be considered a centralized and available server, we designed a decentralized advertisement protocol. At a configurable frequency, a gateway will check if it is valid, and if so, it will generate a gateway advertisement message. This packet contains its MANET network address, its equivalence class, a sequence number, and the length of time for which the advertisement is considered valid. This valid time is the least of all valid times for SMS, minus the interval from which those responses were received to the time at which it emits the advertisement. The SMS valid time should always be longer than the advertisement period unless the SMS is planning to shut down before the end of the advertisement period; otherwise, the advertisement will not be persistent even for links that may be persistently valid.

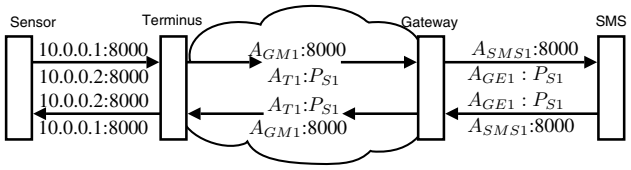

Fig. 2. Illustration of NAPT process in use. The upper and lower addresses are the destination and source addresses. All sensors can have identical addressing, and the sensor key port (P_{S_i}) is used to identify message source and destination.

Advertisements are flooded through the MANET by participating nodes, thus limiting their frequency is important for performance. Our implementation emits advertisements only a few times a minute.

D. Addressing and NAPT

Because event messages are broadcast to all available SMS, and sensors are assumed to be simple nodes, we implement a form of network address and port translation (NAPT) that shields non-MANET nodes such as sensors and SMS from the details of MANET operation. Each sensor addresses its events to one terminus network address and port. The sensors may all have the same network address if there is only one sensor connected to each terminus; sensors connected to the same terminus have different network addresses. This creates the opportunity to make the configuration process for sensors very simple and using very static network and data link stacks on sensor nodes. The process is outlined in Fig. 2.

The terminus nodes maintain a NAPT table that maps each of their connected sensor addresses to a MANET-unique UDP source port. When event packets are received by a terminus, the terminus assigns its MANET network address as the source address and this UDP port (called a sensor key port) as the source port, and attempts to route this new packet to gateways. Upon receiving these messages, gateways attempt to forward the messages to SMS over their gateway-to-SMS links, using their network address on this link as the new source address for the event message. Gateways can alter the source port for event messages, in the event that the sensor key port number conflicts with another port on the gateway-to-SMS link.

When an SMS receives an event, it also receives the sensor key port. This port can be used to address commands to a specific sensor. The SMS forwards commands to the gateway from which it received the event, and the gateway, using

its own NAPT table, determines which terminus corresponds to this sensor key port, and forwards the command to this terminus. The terminus then determines the network address of the corresponding sensor based on its NAPT table (performing the inverse of the NAPT mapping for events), and sends the packet to this sensor.

This architecture allows all sensors to address their events to an identical network address and UDP port, and themselves have identical network addresses and source UDP ports, but permit the SMS to address individual sensors with commands. As well, the SMS do not need to maintain any routing information to reach sensors with commands - they simply respond to the same address and port from which they received the event, and the gateway and terminus handle message forwarding. This isolates non-MANET nodes (sensor and SMS) from the details of the MANET, but these nodes still utilize the MANET for reliable message delivery.

III. Implementation

In our implementation, the nodes are embedded single-board computers using CompactFlash for permanent storage, and roughly equivalent to 486 personal computers in terms of processing power. They are battery powered, and have two 802.3 wired ethernet interfaces as well as 1 Watt 802.11a,b,g interfaces that operate in ad-hoc mode. Linux is used as the embedded operating system, and the protocols are implemented with the *Click Modular Router* [14] (referred to as Click) a packet routing and modification environment written for Linux in C++. A Click router is a connection of modular elements that each perform simple tasks (such as calculating checksums or filtering based on UDP port).

The MANET uses an implementation [15] of the Dynamic Source Routing (DSR) protocol, also written for Click. DSR permits MANET nodes to use the mesh network to reach destinations through hops that may not support our delay tolerant networking protocols, but do participate in DSR.

Fig. 3 outlines the logical interconnections of major elements in a gateway node. Source code and more details are available[1].

A. Reliable Packet Forwarding

Because links are intermittent and unreliable, reliable packet forwarding elements are implemented which provide a flow control mechanism with many similarities to TCP. These elements utilize a selective and cumulative acknowledgement scheme to determine when packets have been successfully delivered. Each packet to be reliably forwarded is assigned a sequence number and sent to the destination. When the destination node successfully receives it, the node generates a reply acknowledgement containing the sequence number of the packet it received (selective), as well as the last sequence number for which all packets have been received (cumulative).

If packets are not successfully delivered within a certain time frame, the sender element can return the packet to other

[1]http://pecowiki.colorado.edu/doku.php?id=dsr-cu

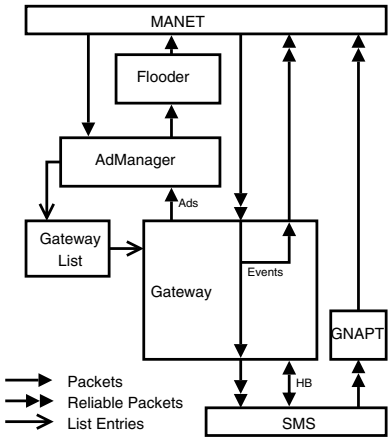

Fig. 3. Gateway Element

elements. In this case, the terminus or gateway will select another gateway to forward the packet to, using the same reliable packet forwarding architecture.

A reliable file transfer protocol similar to a DTN bundle protocol is built on top of the packet forwarding. Sequence number and transfer identification in each packet allows reassembly by the receiver. A setup packet provides file name and size information. The setup packet can be received at any time before, during, or after the file data is transferred. Unlike TCP, packets can be received and processed out-of-order. This property has implications for relaying discussed later.

B. Reliable Packet Forwarding Adapter

Although sensor and SMS nodes are insulated from the architecture of the MANET, their links to terminus or gateway nodes may be unreliable. A reliable packet forwarding adapter was implemented using the Java programming language to permit a variety of systems to utilize our reliable packet forwarding protocol. Any system supporting Java network sockets is capable of using this adapter.

IV. Gateway as Ferry

To illustrate the protocols' operation we consider a gateway ferry scenario. In this scenario, the terminus cannot communicate with the gateway at the same time as the gateway communicates with the SMS. However, our network can still support the end-to-end communication, with delay times that depend on the mobility characteristics of the gateway. In this case, the gateway would establish a heartbeat connection with the SMS when it is nearby the SMS, and begin emitting advertisements. Then, it would move away from the SMS, towards the terminus, but continue emitting advertisements for some period after it stopped receiving heartbeats. If the heartbeat lifetime is sufficiently long, then once the gateway is in the proximity of the terminus, the terminus will receive an advertisement from that gateway, and send pending event packets to the gateway, which will accept them and attempt to forward them to the SMS. But it will not be able to reach the SMS, because it is out of range. The gateway

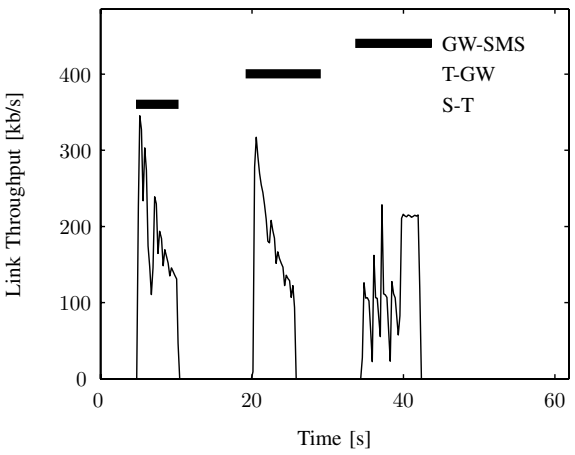

Fig. 4. Staged relaying: The solid bars indicate periods of link availability, and the lines beneath indicate the throughput over those links. Data is delivered end-to-end over the disconnected network.

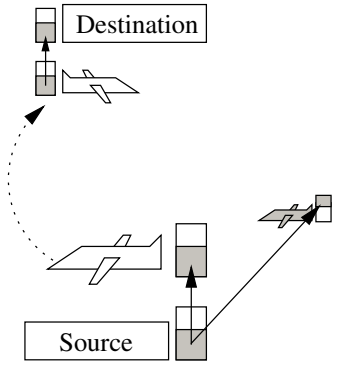

Fig. 5. Incremental Message Delivery: An orbiting relay node can receive part of a message and forward it to the destination before receiving the complete message. A second relay (shown in grey) can arrive later, receive the remainder and deliver. Thus, the destination receives the message after one orbit delay, rather than two.

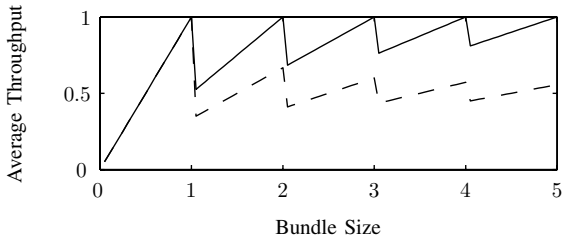

Fig. 6. Incremental Messages: Normalized average throughput for delivery of a message with an intermediate orbiting relay. Incremental (solid) uses orbits more efficiently than staged (dashed).

can move towards the SMS, and once in range, can deliver the event data and also pick up any commands from the SMS, destined for a sensor-terminus pair. This delay tolerance permits communication over large geographic areas; the use of small UA means sensor data may not be received in real time, but can be received with a delay time of the time it takes the UA to follow the terminus to SMS loop.

Fig. 4 shows relaying in a testbed environment. In this case, wireless links were simulated with 10Mbps ethernet so that the links can be opened or closed by a remote test monitor. The black bars represent the periods during which a corresponding link was opened. The flow of data from one relaying stage to the next can be observed beneath the links.

Our architecture has also been tested in an outdoor environment. Sensor and terminus nodes were separated from an SMS node over a distance of approximately 1 kilometer, and it was ensured that there was not a direct connection between the sensors or terminus and the SMS. A gateway node was attached to the roof of a car and driven between the two disconnected MANET subnetworks, and it was verified that the gateway reliably relays events and commands.

Our approach to out-of-order and atomic forwarding permits a staged delivery scheme. A common delay-tolerant scheme is to transfer an entire bundle as the smallest unit over a stream protocol such as TCP [13]. This is of limited use in the scenario depicted in Fig. 5, where a relaying gateway node (pictured as a UA) is not within range long enough to receive the entire bundle. In this case, a relay may have to orbit several times to receive the complete bundle, and then relay several more times to deliver the complete bundle; even this requires modification so that the TCP stream does not expire. In our network, the relay can receive some portion of the bundle during the first orbit, and deliver most or all of that before receiving more data on the next orbit.

Fig. 6 shows an analysis of this delivery scenario. The dashed lines represent staged bundle delivery schemes, such as FTP. In this case, the bundle must be completely delivered to the relay before the relay can deliver it to the destination. If the bundle can be entirely transmitted from the source to the relay in one opportunity (normalized bundle size is less than 1), then there is no advantage to our scheme. However, as the bundle size increases, a staged relay will perform multiple orbits during which it is only receiving, then multiple orbits where it is only transmitting. Our incremental scheme transmits and receives alternately during each orbit, which will achieve higher throughput.

In a further extension, if there are multiple relays, our approach can divide a bundle between the relays and make incremental delivery progress, instead of attempting to deliver the complete bundle to only one relay.

V. Fault Tolerance

This section considers two network fault scenarios to see the capabilities and limits of the architecture.

A. Gateway Cannot Reach SMS

In this scenario, the gateway cannot reach the SMS after many retries, perhaps because it has moved permanently out of range of the SMS, or the SMS is unavailable. We envision this scenario to occur when UA must be grounded to refuel and no longer have line-of-sight visibility of an SMS, or weather prevents satellite connectivity to an SMS. After a configurable number of retries, the gateway will check its

received advertisement list to see if another gateway in its equivalence class has advertised an SMS connection recently. If present, there is another gateway whose SMS connection is functioning. The original gateway will then attempt to forward the message to this new gateway. If successful (i.e. an ACK is received), the new gateway has taken custody of the packet and responsibility for delivering it. If unsuccessful in forwarding, the original gateway will retry delivering the packet to the SMS for which it is intended. This allows the MANET to dynamically adjust to changing gateway-to-SMS connectivity without requiring any communication with the message source.

B. Lost Node

A common problem facing delay-tolerant networks is how to handle the loss of a node. In our network, the loss of any node that is not a sensor, terminus, gateway, or SMS will simply result in a retransmission. However, the loss of a terminus or gateway will result in the loss of any carried packets. If those messages were fragments in larger bundles, the other fragments are correctly delivered and assembled properly.

VI. Conclusion

This paper describes an architecture for the reliable communication between a dynamic set of widely spaced sensors and sensor monitoring stations. At the heart was the definition of network services that reside in special gateway nodes. It solves the problems of sensors and SMS discovering connection paths between each other; non-concurrent and intermittent connectivity; and efficient multicasting of sensor data to all SMS. It was shown that the design choices allow for more efficient and reliable operation than existing protocols. The architecture was implemented and tested and shown to provide reliable performance in laboratory and outdoor environments.

Future work will continue to improve upon the protocol. A network level end-to-end acknowledgment would mitigate packets lost with lost terminus and gateway nodes. A more automated scheme for configuring gateways would remove the current manual equivalence class configuration. The sensor is currently connected to a single terminus. A protocol similar to the SMS heartbeat protocol would allow multiple potential terminus to serve a single sensor. Currently there is no direct interaction between node mobility and communication. Nodes just blindly attempt to forward packets over each stage. The gateway protocols could provide more support for mobility, for instance, to have a mobile node loiter longer when there is more data to be transferred.

We view the architecture as a special case of DTN, however sensor data collection is likely to be an important application in sparsely connected networks such as vehicle-to-vehicle communication, animal migration monitoring, and interplanetary exploration.

Acknowledgment

This work was supported by NSF award CNS 0428887, AFOSR grant no. FA9550-06-1-0205, and a contract from L3-ComCept Corporation.

References

[1] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *MobiHoc'04*, May 24-26 2004, pp. 187–198.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, pp. 393–422, March 2002.

[3] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," in *Elsevier Ad Hoc Network Journal*, 2004, in press.

[4] K. Fall, "A delay-tolerant network architecture for challenged internets," in *SIGCOMM '01*, 2003, pp. 27–34.

[5] Y. Chen and T. Boult, "Dynamic home agent reassignment in mobile IP," *IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 44–48, 2002.

[6] E. P. Vixie, S. Thomson, Y. Rekjter, and J. Bound, "Dynamic updates in the domain name system (DNS UPDATE)," RFC 2136, IETF, April 1997.

[7] D. Chakraborty, A. Joshi, Y. Yesha, and T. Finin, "GSD: A novel group-based service discovery protocol for manets," in *4th IEEE Workshop on Mobile and Wireless Communications Networks MWCN'02*, Sept. 2002, pp. 140–144.

[8] T. Bova and T. Krivoruchka, *Reliable UDP Protocol*, IETF Internet Draft, 1999.

[9] R. R. Steway, Q. Xie, K. Morneault, C. Sharp, H. J. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson, "Stream control transmission protocol," *RFC 2960, IETF*, October 2000.

[10] L. Ong and J. Yoakum, "An introduction to SCTP," RFC 3286, IETF, May 2002.

[11] P. T. Conrad, G. J. Heinz, A. L. C. Jr., P. D. Amer, and J. Fiore, "SCTP in battlefield networks," in *MILCOM '01*, 2001, pp. 289–295.

[12] B. Ouyang, X. Hong, and Y. Yi, "A comparison of reliable multicast protocols for mobile ad hoc networks," *IEEE SoutheastCon 05*, April 2005.

[13] V. G. Cerf, S. C. Burleigh, R. C. Durst, K. Fall, A. J. Hooke, K. L. Scott, L. Torgerson, and H. S. Weiss, *Delay-Tolerant Network Architecture*, IETF Internet Draft, March 2006.

[14] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The click modular router," *ACM Transactions on Computer Systems*, vol. 18, no. 3, pp. 263–297, August 2000.

[15] S. Doshi, S. Bhandare, and T. X. Brown, "An on-demand minimum energy routing protocol for a wireless ad-hoc network," *Mobile Computing and Communications Review*, vol. 6, pp. 50–66, 2002.