PGN3 – Store and Forward Approach to Dynamic Networks

William Raymond, Philip Sage, Raquel Castro, Brian Fisher
George Mason University
< wraymond, psage, rcastro2, bfisher2>@gmu.edu

Abstract

Continuous network coverage is becoming a requirement for many military, commercial, outer space, and humanitarian missions. Wireless networks are required where established infrastructures do not exist and where the environment may compromise communications - often times on purpose by external agents. This paper proposes an innovative approach of using data mules to allow for sparse network edge nodes to communicate with other nodes and the larger data network. In our protocol, mules will be used to collect and distribute information as nodes move in and out of a distressed network. Our proposed protocol will make it possible for edge nodes to receive additional data in less time and with less end-to-end delay.

1. Introduction

Wireless networks are becoming common for more and more applications including: connecting devices from any location to digital networks, global positioning systems, military applications, and scientific research. Unfortunately, many of these devices are working on the edge of the network where there is a limited exposure to the rest of the network. Many existing techniques for connecting edge nodes to a network do not take into account that the network is not fully connected due to an extreme degree of sparseness.

This paper addresses these problems by defining an innovative protocol for communicating among very sparse edge network nodes. This protocol will accomplish its goals by predicting where and when an edge node will reconnect to the network. It will also use store and forward techniques as well as data mules to route messages through the shortest available paths.

1.1 Motivating Example

Delivery of "heavy-weight" data such as video and imagery to personal video devices over a sparse, wireless, or distressed network, illustrates this problem. An example of such a device is the much-rumored Apple Tablet, which is reported to allow the user to download video on demand.

The problem exists because one of the goals of commercial companies, such as Apple, is to provide users with as much flexibility in using the device as possible. This includes using the device on the edge of the network where reliable communication is not continuous due to multi-path and fading effects of radio waves.

For small content, such as simple text messages, e-mails, and similar content, this is not a problem. But for heavyweight data such as video, where gigabytes of information must be downloaded, these devices no longer can reliably receive their information.

The simple solution, currently employed by most companies, is to require the device to connect to the network via Wi-Fi and/or other fairly large bandwidth networks. However, from a consumer's point of view, data should be reliably downloaded to the device from anywhere in the world.

Our solution to this problem is to consider using a store and forward model for content delivery, utilizing existing networks already available to the consumer. These networks include Low Bandwidth Networks (LBN) such as GPRS or Edge Networks, faster 3G networks, and High Bandwidth Networks (HBN) such as 4G or LTE networks or Wi-Fi networks.

2. Research Problem

2.1 Challenged Networks

The Internet provides process-to-process communications. These communications occur over networks supported by different types of router technology and protocols. In general, the networks comprising the Internet provide end-to-end delivery of message packets, reasonable round-trip time between any two nodes, and relatively small amounts of lost packets. Challenged networks, on the other hand, have been defined as networks that may suffer frequent, possibly unpredictable disconnection, high delay, high data rates, or asymmetric data rates between source and destination [1]. Slow transmission rates, frequent disconnections, excessive queuing, and nodes with limited memory and processing capabilities characterize challenged networks.

Examples of challenged networks include [2]:

- **Terrestrial Mobile Networks**: Mobility and changes in signal strength affect these networks.
- **Exotic Media Networks**: This type of network includes satellite communications and long distance radio or optical links.
- **Military Ad-Hoc Networks**: These networks exist in hostile environments and may be affected by mobility, environment, or intentional jamming.
- **Sensor/Actuator Networks**: These networks have limited end node power, memory, and CPU capability.

Early approaches to challenged networks included [2]:

- Applying Internet architectural concepts to challenged networks.
- Attempting to engineer problem links to act in a similar manner to typical Internet links.
- Attaching challenged networks only to the edge of the Internet by means of special proxy agents

Our research problem will define a protocol for communicating among a very sparse network of edge network nodes. This will be accomplished by predicting where an edge node will reconnect to the network, computing shortest paths, and through the use of store and forward techniques to transfer information between the edge nodes. Utilization of sensor network ideas such as data mules to preposition the information thus providing an acceptable quality of service (QoS) for the end-user will be employed.

3. Related Research

3.1 Delay Tolerant Network Architecture

Delay Tolerant Networking (DTN) is a network architecture designed to address the issues of challenged networks. DTN architecture provides the ability to deal with network disruptions through the use of store and forward message routing. DTN networks provide a service similar to electronic mail, however with enhanced name, routing, and security capabilities [3].

DTN nodes must support persistent storage, as they will be required to store data over long periods of time. In a challenged network, it must be assumed that links are unreliable and may be out of service for extended periods of time.

3.1.1 Bundle Layer and Protocol

DTN applications can send messages of various lengths. The bundle layer combines one or more of these messages into bundles. Bundles contain an originating timestamp, useful life indicator, a class of service designator, and a length [3]. DTN nodes for routing bundles use the information in the bundle header across the network.

The Bundle Protocol was designed to improve the transfer of data across distressed networks. With the Bundle Protocol, DTN end system applications do not have to transfer data in multiple packets. Using the Bundle Protocol, a DTN can consolidate data into bundles. These bundles can be stored at nodes whenever connectivity is interrupted. This eliminates the need for end-to-end connectivity as bundles can be stored at intermediate nodes and retransmitted when connectivity is restored.

The bundle layer exists at a layer above the transport layers of the networks on which it is hosted and below the application layer [3]. The bundle layer collects data into bundles and can transfer them across dissimilar networks. Bundling architecture requires that end applications “register” with bundle agents [9]. In this architecture, applications pass data to bundle agents. The agent combines the messages into bundles, and transmits them to the destination agent. The bundle agents take responsibility for the transfer across a DTN, but it is the responsibility of the end

application to guarantee that the data arrives uncorrupted and intact. The bundle architecture does not provide for error checking or checksums that could result in data corruption.

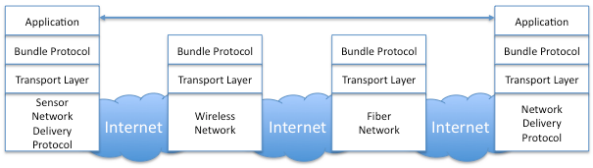

Figure 1 – Bundle protocol overlayed over transport layer to take responsibility for delivery.

In the DTN architecture, the bundle layer overlay network is expected to provide the following features [8]:

- The ability for a bundle agent to take full responsibility for the bundle reaching its final destination.
- The ability to deal with service interruptions and intermittent connectivity.
- The ability to cope with extended propagation delays.
- The ability to work with scheduled and opportunistic connectivity.
- The ability to utilize late binding of endpoint identifiers to network addresses.

Bundles are passed from node-to-node along a path to a destination node. When a bundle is passed to another node, the receiving node may accept custody [10] of the bundle, pass the bundle to the next node in the path, or reject the bundle. If the node accepts custody of a bundle, the node must be able to store the bundle until it can transfer it to the next node in the path. If a node rejects a bundle, custody of the bundle remains with the previous node.

Our proposed approach will bundle multiple data packets for storage on data mules. The mules will transfer the bundles from data mule to data mule until connectivity to the greater network is reestablished. Upon reestablishment of connectivity to the greater network, the data will be unbundled and transferred to its network destination.

3.1.2 Routing and Forwarding

DTN architecture is designed for networks that can not guarantee end–to-end routing. In a DTN architecture routes are made up of time-dependent contacts [2]. Contacts are defined by their start and end times, capacity, latency, endpoints and direction.

Our approach is envisioned to function with the following major types of contacts [3]:

- **Persistent Contacts:** This type of contact is characterized as always being available.
- **On-Demand Contacts:** Service for his type of contact must be initialized before each use. Once initialized, on-demand contacts behave like persistent contacts until service is terminated.
- **Intermittent – Scheduled Contacts:** A scheduled contact is available only at predefined times for a predefined duration.
- **Intermittent – Opportunistic Contacts:** An opportunistic contact is intermittently available but does not appear at a scheduled time and the connection duration is unknown.
- **Intermittent – Predicted Contacts:** A predicted contact does not occur at a predefined time or for a predefined duration. This type of contact is intermittent, but its availability can be predicted, to within a reasonable level of confidence.

The object of traditional routing algorithms is to choose the most efficient path from source to destination. In DTN networks the optimum path is rarely apparent and, at any given time, may not be available. For these reasons the primary objective of most routing algorithms over DTN networks is not to select the optimum path, but to select the path with the greatest probability of successfully delivering the message.

Even though the primary object of routing in a DTN network is to successfully deliver the message to its destination, routing algorithms also attempt to minimize the inevitable delays that occur in DTN networks. Most DTN routing algorithms fall into the general categories of source routing algorithms and per-hop routing algorithm [4]. With source routing, the complete path a message will follow is selected before the message leaves the source node. In DTN networks, this type of routing can fail when nodes are

not available. With per-hop routing, a decision is made at each router as to which is the next router in the path. Per-hop routing may improve performance over source routing, but it leads to loops due to each node's limited view of the network. Within this category, routing algorithms use two different general approaches, replication and knowledge [5].

With the replication approach, multiple copies of messages are sent through the network. The logic behind this approach is that by sending many copies of the same message, one increases the probability of at least one copy reaching its destination. The most dependable, but most expensive, method in this strategy is to forward a copy of the message to each node of the network. The only scenario in which this method fails is if every node in the network is unable to deliver the message to its final destination.

Our design is intended to follow the knowledge approach. Using the knowledge approach, nodes make routing decisions based upon knowledge of the state of the network. This class of routing algorithms can be divided into algorithms with zero knowledge, partial knowledge, and algorithms with complete knowledge of the network [4]. In the worst case, if a node has zero knowledge of the network it will follow static rules that can not adapt to the changing network conditions. In the best case, a node has complete knowledge of the network at all times and is always capable of selecting the most efficient path. Our approach seeks to use knowledge of connectivity to preposition data on data mules for delivery when shortest-path connectivity becomes available.

3.1.3 Routing Strategies

Routing strategies can generally be divided into two families: flooding strategies and forwarding strategies [5]. Flooding strategies are based on duplicating messages to enough nodes that the message eventually reaches its destination. These strategies, in general, follow the replication approach. On the other hand, the forwarding strategies we employed follow the knowledge approach and attempt to select the best path based upon information available on the current state of the network.

3.1.4 Forwarding Strategy

Forwarding strategy algorithms use knowledge of the state of the network to select the optimum path for a message to reach its destination. Following this strategy, paths can be selected using location-based routing, assigning metrics to nodes, or by assigning metrics to links [5]. In selecting paths, these algorithms usually consider queuing delay, transmission delay, and propagation delay. Algorithms using this strategy generally do not use replication; they send a single message to the destination.

Popular forwarding strategies include: Gradient Routing and Link Metrics [5]. With Gradient Routing, each node is assigned a weight that represents its suitability to deliver a message to the given destination. In this approach, the node holding the message passes it to any connected node containing an improved destination weight. Link Metrics uses topology graphs, assigns weights to each link, and run a shortest path algorithm to find the best routes. For delay-tolerant networks, delivery ratio and delivery latency are common metrics.

The simplest forwarding strategy is Location-Based Routing [5]. In this strategy nodes forward messages to any node that is closer to the destination than the current node. Location-Based Routing strategies only require that a node know its own location, the destination location, and the location of all reachable nodes. With this information the node uses a function to determine where to forward the message.

Virtual Repository Routing is a form of Location-Based Routing [7]. We have chosen to use Virtual Repositories as our proposed approach, as they provide static locations to which, data destined for a given location may be routed. Whenever a destination is unknown or unreachable, the message will be forwarded to a Virtual Repository. When connectivity is reestablished, the Virtual Repository will then forward the message to its final destination.

3.1.5 Movement-Controlled Strategies

In Movement-Controlled Strategies, all data is moved from source node to a data collection area. The data is then forwarded from the collection area to the destination node. The entity our approach will use to assemble data is the Data mule [7]. Data mules collect data from nodes and pass it to other Data mules or to a central collection area. In our approach, Data mules are also used to transfer messages to their destination.

3.2 Routing Protocols

Routing protocols are used by networks to select the path information will follow to reach its destination. Routing tables are used to store information routers use to direct traffic from one router to the next.

3.2.1 Routing Algorithms

Routing algorithms populate routing tables with the information required to make efficient routing decisions. Routers use these routing tables to select optimal paths and decide when and where to forward data.

Our routing algorithm was designed to select an optimum path by considering the following metrics [30]:

- **Path length**: Based upon a cost assigned to each network link.
- **Reliability**: The dependability and availability of a network link.
- **Delay**: The time it takes information to pass from one link to another.
- **Bandwidth**: The capacity of a network link.
- **Load**: The amount of data a link is processing.
- **Communication Cost**: The cost of the network infrastructure.

Additionally, our routing algorithm was designed based upon selecting between six types of routing-algorithm differentiators [30].

- **Static versus dynamic**: Our algorithm is dynamic, as it must adapt to changing network connectivity.
- **Single-path versus multi-path**: Multi-path was chosen to increase the probability of information successfully reaching its destination.
- **Flat versus hierarchical**: We chose hierarchical so as to support different types of links with different capabilities.
- **Host-intelligent versus router-intelligent**: Our protocol is router-intelligent, as the routers in a DTN network must act as store and forward devices.
- **Intra-domain versus inter-domain**: We used an inter-domain protocol to function within a DTN network.
- **Link state versus distance vector**: Our distance vector algorithm is designed to share routing table information only with neighbors.

3.2.2 Knowledge-Based Routing

Data mules are used as mobile routers in DTN networks to receive, store, and forward information to and from other nodes on the network. Data mules are used in DTN networks that experience connectivity disruptions, because they are mobile and support store-and-forward algorithms.

Data mules use Hello-Response [30] techniques to locate approaching routers. Mules carrying data periodically broadcast a “Hello” message. When a neighboring router receives a “Hello” message, it responds with a “Response” message announcing its presence.

Mobile routers use knowledge-based routing algorithms and Hello-Response techniques to decide when and where to forward data based on location and mobility. There are five basic strategies used by data mules to decide when and where to forward data [30].

- **No-talk**: In this strategy data mules do not talk to each other. A mule will store data until it can communicate with the destination node.
- **Broadcast**: In this strategy data mules send data to every mule that responds to one of its hello messages.
- **Location-based**: With this strategy, a mobile router forwards a message to any other mule it contacts if that mule is closer to the final destination.
- **Motion-Vector**: Motion-Vector strategies seek to use the speed and direction of neighboring data mules to determine which mule will be closer to the destination in the future.
- **Move Look-ahead**: This strategy is similar to the Motion-Vector strategy except it seeks to predict future changes in direction that may be taken by destination mobile routers.

Our protocol is designed to combine features for all of the above knowledge-base routing strategies.

3.2.3 Store-Carry-Forward Protocols

Store-Carry-Forward [31] is the category of protocol used to overcome the inconsistent connectivity of DTN networks. Our solution proposes using a Store-Carry-Forward protocol in combination

with the use of data mules as a solution to DTN network communications.

Our solution utilizes characteristics and features from the following Store-Carry-Forward network routing protocol strategies:

- **Trajectory-based Protocols**: Trajectory-based protocols [31] make routing decisions based on the direction and speed of mobile routers. When selecting amount, the protocol selects the router that will move closest to the destination in the shortest period of time.

- **Epidemic Routing Protocols**: Epidemic routing protocols [32] are flooding based protocols where information is shared among neighboring mobile nodes. The strategy utilizes the movement of data mules to assist in moving data to its eventual destination during periods of lost connectivity.

- **Border Node Based Routing (BBR)**: Border Node Based Routing (BBR) [32] uses broadcast messages to gather information on all border nodes through the use of Hello-Response messages. Routing decisions are made based upon information received from the responses to these broadcast messages. As connectivity constantly changes in DTN networks utilizing data mules, new Hello-Response messages are broadcast before all routing decisions.

- **Geographic Routing Protocols**: Geographic routing protocol [31] make routing decisions based upon the current location of nodes. These protocols use greedy heuristics to choose the neighbor node that is currently closest to the destination node.

3.3 Mesh Networks

Mesh networking is a type of networking where each node in the network may act as an independent router, regardless of whether it is connected to another network or not. It allows for continuous connections and reconfiguration around broken or blocked paths by "hopping" from node to node until the destination is reached. A mesh network whose nodes are all connected to each other is a fully connected network. Mesh networks differ from other networks in that the component parts can all connect to each other via multiple hops, and they generally are not mobile. Mesh networks can be seen as one type of ad hoc network. Mobile ad hoc networks (MANET) and mesh networks are therefore closely related, but MANET also have to deal with the problems introduced by the mobility of the nodes. Mesh networks are self-healing in that the network can still operate when one node breaks down or a connection goes bad. As a result, the network may typically be very reliable, as there is often more than one path between a source and a destination in the network. Although mostly used in wireless scenarios, this concept is also applicable to wired networks and software interaction [11].

One of the major benefits of a mesh network is its path diversity. The network can provide many routes to the main network in case one of the routers fails or its transmission path is temporary blocked or unavailable [11].

A wireless mesh network can be seen as a special type of wireless ad hoc network. It is often assumed that all nodes in a wireless mesh network are immobile, however, this is not the case all of the time. The mesh routers may be highly mobile. Often the mesh routers are not limited in terms of resources compared to other nodes in the network and thus can be exploited to perform more resource rigorous functions. In this way, the wireless mesh network differs from an ad-hoc network, since all of these nodes are often constrained by resources.

Most wireless network installations today involve a set of access points with overlapping coverage zones and each access point connected to a wired network tap. Mesh networks remove this strong connectivity requirement by having only a few of the access points connected to a wired network and allowing the others to forward packets over multiple wireless hops [18].

Mobile clients get network access through the mesh by connecting to wireless access points. As a mobile client moves away from an access point and closer to another, it switches its connectivity to the closest access point. This connectivity change involves a transition (handoff) before being able to route packets to and from the new access point. Ideally, the handoff should be completely transparent to mobile clients. There should be no interruption in network connectivity and the communication protocols involved should follow the standards deployed in regular wireless devices. A wireless network that offers such a service is called a seamless wireless mesh network [18].

3.3.1 The Mesh Topology

Figure 2 illustrates a notional wireless mesh providing ubiquitous coverage, demonstrating how the mesh routers are situated in a typical environment.

An advantage of a mesh network is its robustness because it is not dependent on the performance of just one of its nodes for its operation. In mesh network architecture, if the nearest AP (access point) is down or there is localized interference, the network will continue to operate as data will simply be routed along an alternate path [12].

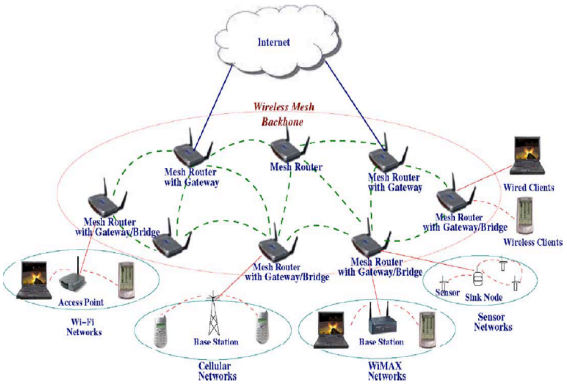

Figure 2 - Wireless mesh where each node may act as an independent router.

Higher bandwidth is also an advantage; the physics of wireless communication dictate that bandwidth is higher at shorter range, because of interference and other factors that contribute to loss of data as distance increases. One way to get more bandwidth out of the network is to transmit data across multiple short hops. That is what a mesh network does [12].

Our solution is dependent upon this heterogeneous network structure in that we will rely upon the different networks to provide both low bandwidth access and high bandwidth access allowing downloads to occur thoughout the network.

A wireless mesh network is a complex system with many inter-dependent factors that affect its behavior. The factors include networking protocols, traffic flows, hardware, software, different faults, and most importantly the interactions between them. Troubleshooting a multi-hop wireless network is even more difficult due to unreliable physical medium, fluctuating environmental conditions, complicated wireless interference, and limited network resources [13].

In research done by Microsoft, UC Berkeley and University of Texas, they applied their system to detect and diagnose performance problems arising from four faults: [13]

- **Packet dropping**. This may be intentional or may occur because of hardware and/or software failure in the networked nodes. It is important to notice persistent packet dropping.

- **Link congestion**. If the performance degradation is because of a lot of traffic on the link, this should be identified.

- **External noise sources**. RF (Radio Frequency) devices may disrupt on going network communications.

- **MAC misbehavior.** This may occur because of hardware or firmware bugs in the network adapter or where a node intentionally tries to use more than its share of the wireless medium.

These faults are more difficult to detect than fail-stop errors, which could happen when a node turns itself off because of power or battery outage. These faults can have fairly long-lasting impact on the network performance.

In recent years, mesh networks have been advocated as a cost-effective approach for providing high-speed last mile connectivity. In addition, mesh networks may also spur the growth of new neighborhood-specific applications, such as video sharing among community members that require high throughput. To build cost-effective mesh networks, it is desirable to operate the network in a multi-hop fashion using commodity wireless hardware. However, wireless medium is a shared resource, and the capacity of a multi-hop network quickly degrades as the node density and network diameter increase. As a result, there is a need to develop solutions to enhance the network capacity to meet the needs of mesh network applications. [15]

Existing commodity hardware like “Atheros Inc.” and “Maxim 2.4 GHz 802.11b Zero-IF Transceivers”, often allow users to control several physical layer parameters, such as transmission power, data rate, frequency of operation, etc. Such support can be utilized to improve the network capacity. Two examples of improving network capacity by using physical layer support are considering utilizing multiple frequency-separated channels often provisioned for in wireless standards to increase network capacity and considering an approach to improve the capacity of any given channel by introducing “spatial backoff” mechanisms. [15]

3.4 Store and Forward

Store and Forward is the act of buffering messages in the middle of a network for a known period of time until delivery to the final destination is possible. It is commonly used in networks with transient connectivity and reasonable expectation of eventual delivery. An example of this is when a link is temporarily unavailable due to being out of range as in the case of a wireless network. However there are also other benefits gained from storing a message before forwarding it. One such example is time-shifting, storing a non-critical message until off-peak communication rates are in effect.

3.4.1 Low Bandwidth Store and Forward

Store and Forward was created in the early 1970s to handle the low bandwidth network connections and unsophisticated receive buffers. In these systems it was common for the receive queue to be full before all of the information was received from the sender. Because of the amount of time that the receiving machine could take to process the information, the sender would often times need to hold onto the information before forwarding it on. This problem was further complicated by the fact that, for various reasons, the network connections often failed for substantial periods of time. Store and Forward is a technique used to send information between nodes where intermediate nodes may need to store the information for an unknown period of time, before forwarding the information on [19].

With the introduction of modern networking hardware and protocols, Store and Forward techniques were no longer needed. Current networks are more sophisticated and reliable making Store and Forward techniques obsolete [20]. However, using a Store and Forward approach with broadcasting allows us to overcome many of the problems associated with the limited time that a node will be able to receive information. This is especially true if information can be distributed to multiple sites by predicting where the download will occur.

3.4.2 Quality of Service (QoS) and Delay Guarantee

Quality of Service is an important aspect of any communication channel, especially one that can be used for future Internet and voice traffic. End-to-end delay is the QoS parameter that is focused on in both [21] and [22] because they claim it is one of the most important QoS parameters. End-to-end delay is the total time it takes from when the data is ready to be sent at the source to when the final byte of data is received at its destination. End-to-end delay guarantee research is looking into routing algorithms that take into account the available bandwidth of a link and how to fairly distribute that bandwidth among all of the available networks. This can either be a dynamic distribution of the bandwidth or can be a static distribution giving each connection equal amounts. No matter how the bandwidth is distributed the routing algorithm must be able to know when a link is congested and find the next best route to guarantee that the data will still be received within a specific delay boundary [21, 22].

3.4.3 Security

Wireless communications are very vulnerable to attacks due to their broadcast nature. It is our contention that wireless networks need to be able to secure themselves without the need for large amounts of time, computational power, or extra transmissions.

Current research is looking into efficient ways of securing wireless communication channels [23, 24]. Security can be addressed by encrypting the information being sent; however, there are several different ways of encrypting that data. The most versatile approach is to give each wireless device its own key and encrypting each packet for each individual receiver of that packet. This may be the most versatile approach, but it is extremely inefficient especially when packets are multicast. Therefore, another approach is to use group keys. The difficulty with group keys is that the keys need to be updated every time a user joins or leaves the network.

Note: while we did research some of the security issues, security was not a concern for our current research in this project.

4. Solutions / Analysis

A simplistic approach to the problem of delivering content is to broadcast the information to all nodes within the network as was shown in the flooding routing protocols. The main downside to an approach such as this is there does not exist unlimited bandwidth and storage space necessary to achieve desirable delay and delivery ratio attributes.

This leads us to our proposed solution to the wireless nodes on the edge of the network problem by examining and implementing a basic protocol that is split into three loosely coupled layers: the data-link /

mac layer, the network layer, and a minimal transport layer [25]. The layers are not completely uncoupled in order to maintain efficiency, which may be lost in completely separating the layers.

This utilizes modified Movement-Controlled strategy to provide methods to move information from source node to a data collection area [7] by utilizing the notion of a Data mule to transfer the messages from the collection area to their destination.

The devices are tracked within the network, allowing us to form a topology of the network for locations frequently visited by the target device. When a source has messages for a destination node, it decides the number of duplicate message required based on the frequently visited locations. This information is routed to the frequently visited destination nodes so that information is prepositioned for download.

4.1 Data-link / Mac Layer

The Data-link layer is in charge of obtaining required information to know approximately when any device will be in range of any portion of the connected network. It is also responsible for finding shortest paths; maintaining state information during a connection, dealing with collisions, dropped packets, timeouts, acknowledgments; and scheduling the sending of packets to initiate a connection.

4.1.1 Types of Nodes and Networks

Every node is designated a type in order to know what their purpose is in the network. For our purposes, the types of nodes we are currently considering are Low Bandwidth Node (LBN), High Bandwidth Mules (HBM), and the Edge-Node (EN). An example of a LBN's purpose is to relay small amounts of information between the network and the ENs with low latency but having very little available bandwidth, such as a GPRS or Edge type network. This will mainly be used for lightweight messages that are to be sent to the ENs as well as link state information from the EN. Our assumption is that this type of network is more prevalent than HBMs.

The HBM is the data mule – a higher powered and larger wireless base-station in charge of transmitting large quantities of data such as images or video to ENs. It is assumed that the HBM has a longer period between connectivity with each node, but will stay generally stay in range for longer durations. The HBM will also have a higher bandwidth connection with the ENs because the range will be smaller and have less interference than from an EN to the network. An example of this is when a device enters into a Wi-Fi network. The range of the network is smaller, but the bandwidth is considerably larger.

An assumption is that HBM is connected to a fairly large network so that there is not a bottleneck in the system. Since the HBM is a larger base-station with more power and a larger antenna it will be able to transmit data down to the EN network with higher rates than the ENs themselves can.

The final type of node is the EN itself. The EN can both receive and transmit information. Its primary role is asymmetric by receiving potentially gigabytes of information and primarily sending up only Ks of information as request or status.

4.1.2 Link State Tables

In order to compute the shortest path and approximate when an EN will be close to either a LBM or HBM, some link state information must be collected. Link state tables are lists of information that describe the connectivity of a network.

Because this network is not fully connected the link state tables in this protocol store additional information. They not only store a list of connected nodes, but also when they connected and for how long. It is assumed that each node has a unique hardware address that must be registered on the network to differentiate between nodes and to know to what node a connection is related. The types of both of the nodes must be known so routing is possible depending on the data that needs to be delivered. This allows the protocol to differentiate between EN, LBN, and HBM so data can be sent through the appropriate channel. The time last seen is the time that the node last came within range of one another and started to communicate. It is used for calculating when the node will be in range again and where it was last seen.

The duration is used in conjunction with the bit rate to calculate the amount of data that could be transferred during a connection. The duration isn't simply saved as the time from the first contact to the time the node lost contact in the end. Instead the duration removes grey periods (when traveling between networks or coming into and out of networks) and takes into account the number of packets that are lost during the last transmission. This provides an estimate of how much data will be able to go through during the connection. The final piece of data is an estimate of the time until a node will again be within

range. This in combination with the time last seen will allow a protocol to compute the shortest path when reconnected. In effect, this allows the protocol to track where a node is located within the network so that information can be prepositioned.

4.1.3 EN Connection States

When going through a connection between an edge node and the network, each EN has several states it needs to go through in order to complete the connection and pass all the required data. The state information serves many purposes. It allows the EN to keep track of a connection between the EN and the network. This is complicated by the fact that the ENs will go in and out of range during a single "connection" due to the non-stationary nature of wireless networks.

The states also keep track of the connection and makes sure that it can differentiate between a grey period and a new connection to that EN. The states also make sure that the minimum amount of information is sent between the EN and network during every connection. This information is the updated link state information for other ENs in the network. The states can be extended in the future to contain other state information for a single connection such as more advanced rate of change for the time between connections, or information tracking missed connections due to a temporary problem or a bad estimation of position. The states can also be thought of as an ordered checklist that must be completed before deciding the connection is complete.

When the EN reaches the final state showing that a connection has completed sending all of the necessary information, the node with the smaller ID of the two begins sending "you there?" packets. These packets are analogous to two people standing in a dark room and one of them continuously asking at some interval, "Are you still there?" and the other one responding in the affirmative. This process is part of keeping track of the duration of connections.

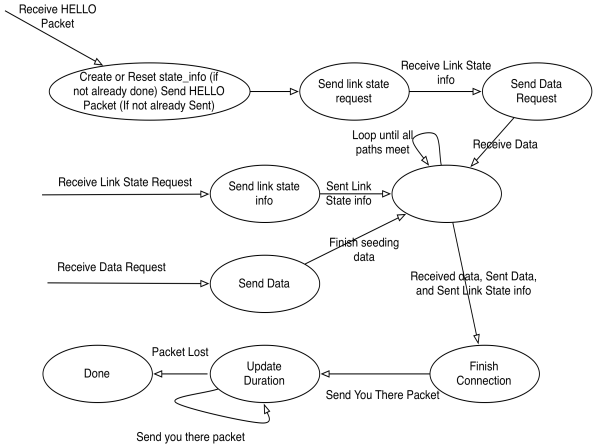

Figure 3. Proposed state transition for Edge Node entering in to network and broadcasting Hello packets until no longer in the network.

4.1.4 Tracking and Connection Attempts

When an EN is first turned on, it has no information on the network, (LBN or HBN), or, for that matter, when it will encounter any network. Before it is able to schedule connections to the network, it must discover and "learn" when it is in range of a particular network. The EN learns this information by starting out in "Discovery Mode". In this mode the EN sends out a broadcast HELLO packet every 60 seconds until it obtains enough information from each network to know approximately when it will be in range again.

One research issue is how long to remain in this discovery mode. We could envision a scenario where we would like to track the device for a month to determine normal patterns of behavior. Behavior we envision analyzing is being at home, or being within a HBN such as a home Wi-Fi network, or driving everyday to work and out into the field where the network is a LBN network. This allows us to form the topology of our network.

4.1.6 Link-State Protocol

Upon connection, nodes exchange summary vectors that list the link-state tables the nodes have received. Each table is tagged with a sequence number that permits the nodes to determine which ones are the most recent. The nodes exchange any missing updates so that they both have the same topology state. Then they re-compute their routing tables and finally can forward messages to the other node. Since we used per-contact routing, the metric for each link is temporarily set to zero to represent the fact that messages can be

immediately sent to the other node. A protocol state machine, shown in Figure 4 is maintained for each connection.

4.1.7 Sending of Packets

Packets scheduled to be sent are added to an outgoing queue. Packets are generally added to the end of the queue, but can be added to the front of the queue in order to expedite delivery. Acknowledgments and resent packets currently are added to the beginning of the queue instead of the end. The MAC protocol will send packets up to a maximum window size. Once acknowledgments are received for one or more packets the MAC protocol will allow more packets to be sent until the maximum window size is reached again [23].

4.1.8 Acknowledgments

Multiple acknowledgments are sent in a single packet in order to limit the amount of packets. An ACK packet is sent immediately once the maximum window size, described above, has been received. In addition, there is a maximum time between acknowledgments to allow for the possibility that the maximum window size is not reached. This maximum time is set to one fourth of the timeout time for a packet allowing ample time for the acknowledgments to be received so packets aren't unnecessarily resent.

4.1.9 Delivery

When a delivery route is not available at the time a message is ready to be sent, the node will store the message in its storage area, waiting for a while to see if possible routes appear. The node, which has messages to send, needs the destination node location as well as its neighboring node locations to make proper routing decisions. Neighboring nodes location information is obtained by asking, and waiting for all replies. In this process, new messages may be received, but they can only be stored at the end of the message queue, waiting their turn to be transmitted. A message is discarded when the sender receives an acknowledgement from the receiver that the message is complete. This message is then broadcast to all data mules to discard the message.

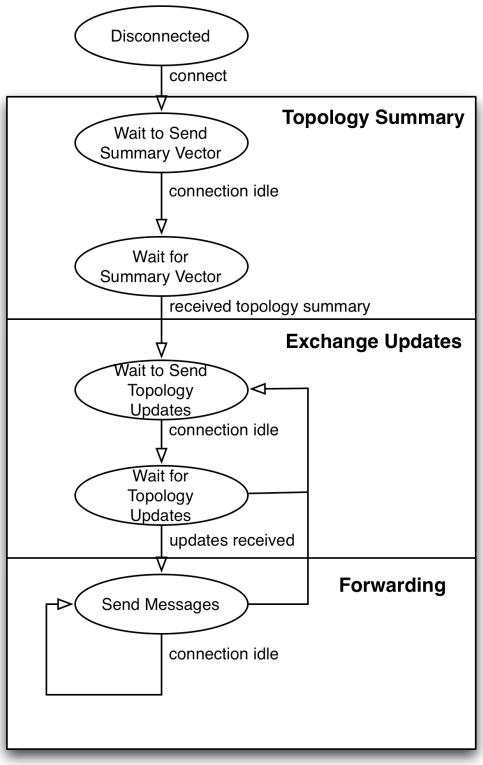

Figure 4 – Link-State Protocol

4.2 Network / Transport Layers

4.2.1 Data Transfer

In the transport layer, each node that can receive data has its own outgoing data queue. The data that is stored in this queue is simply a group of packets that have already gone through the routing process. An assumption is made that each EN will come into contact with either a HBM or a LBN several times. If we can predict this, we will be able to preposition information so we can transmit data. If we predict this incorrectly, we run the chance of congesting a node.

When an EN comes into contact with a HBM, we initialize the transfer of data until either the EN has received all of the information and/or the EN is no longer in the HBN range.

4.3 Theoretical Results

This section shows the theoretical results of a downloaded file using our approach. The ability to position the data at a forward node allows information to be downloaded with minimal overhead on the file.

The traditional method would require initiating the download at each site from the beginning.

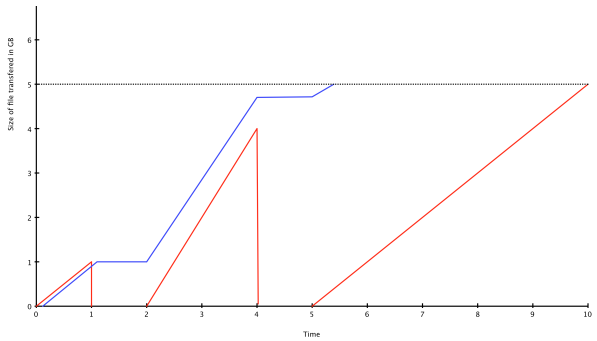

Figure 5 – Demonstrating notional transfer time to transfer a 5 GB file.

The red line represents the traditional file download that times out during the periods from 1 to 2, 4 to 5, etc. until it is at a node long enough to completely transfer the file. Our proposed approach will allow the information to be prepositioned at multiple nodes. There is overhead information that is broadcast when an EN enters into the LBN or HBM so that the file transfer can be initiated from the point where it left off.

A downside to this approach is that there is a need to store multiple copies of a requested file, prepositioned at different HBNs where an EN travels. Akamai [27] has demonstrated that this is a viable approach for positioning content that is frequently downloaded or streamed.

5. Summary and Future Work

We have proposed a protocol for communicating among a very sparse edge network nodes for delivery of heavy-weight content by predicting where and when an edge node will reconnect to the network. Data mules are used to receive messages from a Store and Forward technique that allow us to deliver the content in parts.

Future work in this area would require developing a simulation [28] to quantify the benefits of this approach and to compare some of the methods we researched for Delay Tolerant Networks and Mesh Networks.

6. References

[1] Susan Symington, Robert C. Durst, and Keith Scott, “Custodial Multicast in Delay Tolerant Networks”, 06_100.pdf, 2006

[2] Kevin Fall, “A Delay-Tolerant Network Architecture for Challenged Internets”, p27-fall.pdf, August 2003

[3] V. Cerf, et al, “Delay-Tolerant Networking Architecture”, Delay-Tolerant Networking Architecture.doc, April 2007

[4] Sushant Jain, Kevin Fall, and Rabin Patra, “Routing in a Delay-Tolerant Network”, p299-jain111111.pdf, September 2004

[5] Evan P. C. Jones, “Practical Routing in Delay-Tolerant Networks”, thesis.pdf, 2006

[6] Yong Liao, Kun Tan, Zhensheng Zhang, and Lixin Gao, “Modeling Redundancy-based Routing in Delay Tolerant Networks”, ccnc07-conference.pdf

[7] PJ Dillon and Taieb Znati, “Virtual Repositories in Delay Tolerant Networks”, VirtualRepositories.pdf,

[8] W. Ivancic, W. M. Eddy, D. Stewart, L. Wood, J. Northam, and C. Jackson, “Experience with Delay-Tolerant Networking from Orbit”, ijscn-asms-bundle-paper-submitted.pdf

[9] Lloyd Wood, Wesley M. Eddy, and Peter Holliday, “A Bundle of Problems”, wood-ieee-aerospace-2009-bundle-problems.pdf , December 2008

[10] Susan Symington, Robert C. Durst, and Keith Scott, “Custodial Multicast in Delay Tolerant Networks”, 06_1000.pdf, 2006

[11] “Wireless Mesh Network Definition from PC Magazine Encyclopedia”, http://www.pcmag.com/encyclopedia_term/0,2542,t=wireless+mesh+network&i=54776,00.asp

[12] Ermanno Pietrosemoli, “Mesh Network Lecture” 2004 http://wireless.ictp.it/school_2004/lectures/ermanno/mesh.pdf

[13] Lili Qiu, Paramvir Bahl, Ananth Rao, and Lidong Zhou, “Troubleshooting Wireless Mesh Network.”

[14] Jakob Eriksson, Sharad Agarwal, Paramvir Bahl, and Jitendra Padhye, “Feasibility Study of Mesh Networks for All-Wireless Offices”.

[15] Pradeep Kyasanur, Xue Yang, and Nitin H. Vaidya, “Mesh Networking Protocols to Exploit Physical Layer Capabilities”.

[16] Website, “Atheros Inc.”, http://www.atheros.com.

[17] “Maxim 2.4 GHz 802.11b Zero-IF Transceivers,” http://pdfserv.maximic.com/en/ds/MAX2820-MAX2821.pdf.

[18] Yair Amir, Claudiu Danilov, Michael Hilsdale, Raluca Mus_aloiu, Elefteri, and Nilo Rivera, “Fast Handoff for Seamless Wireless Mesh Networks” Johns Hopkins

University, Department of Computer Science, Baltimore, MD 21218

[19] J. Wu and F. Dai, "Mobility-sensitive topology control in mobile ad hoc networks," in Proc. of IEEE IPDPS, 2004.

[20] J. Wu and F. Dai, "Mobility management and its applications in efficient broadcasting in mobile ad hoc networks," in Proc. of IEEE INFOCOM, 2004.

[21] Qijie Huang, Boon Sain Yeo, and Peng-Yong Kong, "A routing algorithm to provide end-to-end delay guarantee in low earth orbit satellite networks", In Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th, volume 5, pages 2911–2915 Vol.5, 2004.

[22] Qijie Huang, Boon Sain Yeo, and Peng-Yong Kong. "An enhanced QoS routing algorithm for provision of end-to-end delay guarantee in low earth orbit satellite networks", In Wireless Communications and Networking Conference, 2005 IEEE, volume 3, pages 1485–1490 Vol. 3, 2005.

[23] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad-hoc wireless networks," in Proc. of IEEE INFOCOM, 2001.

[24] P. Guptaand and P. R. Kumar, "The capacity of wireless networks," IEEE Transactionson Information Theory, vol. 46, no. 2, pp. 388–404, 2000.

[25] GENSO Project. GENSO. http://www.genso.org/, 2009.

[26] S.D. Shapiro, "Generalizations of random store and forward communication networks", Proceedings of the IEEE, 55(12):2191–2192, 1967.

[27] Website, Akami, www.akamai.com

[28] Network Simulator, The network simulator - ns-3, http://nsnam.org.

[29] Website, myipaddressinfo, www.myipaddressinfo.com/routingprotocols.com

[30] Jason LeBrun, Chen-Nee Chuah, Dipak Ghosal, Michael Zhang, "Knowledge Based Opportunistic Forwarding in Vehicular Wireless Ad Hoc Networks," IEEE VTC Spring 2005, www.ece.ucdavis.edu/~chuah/paper/2005/vtc05-move.pdf

[31] Victor Cabrera, Francisco J. Ros, Pedro M. Ruiz, "Simulation-based Study of Common Issues in VANET Routing Protocols," Spring 2009, Routing-VTC2009spring-cready.pdf

[32] Mingliu Zhang and Richard S. Wolff, "Routing Protocols for Vehicular Ad Hoc Networks in Rural Areas," www.coe.montana.edu/ee/rwolff/Papers/Rural_VANETsv5_figures_at%20end.pdf