Analysis of Simple Counting Protocols for Delay-Tolerant Networks *

Brenton D. Walker, Joel K. Glenn, and T. Charles Clancy
Laboratory for Telecommunications Sciences
US Department of Defense
College Park, MD, USA
brenton@ltsnet.net, jglenn@ltsnet.net, clancy@ltsnet.net

ABSTRACT

Mobile Wireless Delay-Tolerant Networks (DTNs) are wireless networks that suffer from intermittent connectivity, but enjoy the benefit of mobile nodes that can store and forward packets or messages, and can act as relays, bringing packets and messages closer to their destination through a selective forwarding policy. Many DTN protocols compensate for the unpredictability of the network by distributing multiple message copies in the hopes that at least one will eventually be delivered. As the number of message carriers becomes large these schemes experience diminishing marginal benefits from the addition of more message carriers. We describe and analyze the **Simple Counting Protocol**, an extremely simple and robust method for limiting the fraction of nodes that carry a copy of a message. We examine the performance of this protocol in conjunction with several abstract mobility models and show that the protocol performs reasonably well in diverse circumstances. The Simple Counting Protocol does not assume much about node mobility, and therefore should be useful for applications where little is known about node encounter patterns. The simplicity of its implementation will hopefully make it a useful substitute for epidemic routing as a naive lower bound in protocol performance comparisons.

We also show how the same simple techniques and principles can be applied in conjunction with more complex *heuristic* DTN protocols to reduce network resource usage, a scheme we call **Intermediate Immunity**.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication; C.2.2 [**Network Protocols**]: Routing Protocols

*This work was sponsored by the Laboratory for Telecommunications Sciences, US Department of Defense. The opinions expressed in this paper reflect those of the authors and do not represent an the opinions of or an endorsement by the Department of Defense or US Federal Government.

CHANTS'07, September 14, 2007,Montréal,Québec,Canada.

General Terms

Algorithms, Design, Theory

1. INTRODUCTION

The key difference between Delay-Tolerant Networks (DTNs) and the Ad-Hoc Wireless Networks usually studied is that DTNs generally lack end-to-end connectivity. In the mobile wireless variety of DTNs that we focus on, nodes must rely on unpredictable mobile nodes to store and deliver packets or messages. Because of this unpredictability many DTN protocols allow nodes to distribute multiple copies of the messages they receive in the hopes that at least one copy will reach its destination. The extreme of this approach is the epidemic class of protocols [10, 12] whereby all nodes pass on copies of all their messages to every node they encounter until the message is delivered. The variants described in [10] differ in how they attempt to clean up once a copy of the message has been delivered.

More intelligent DTN protocols attempt to restrict the forwarding of messages while preserving network performance. One of the simplest approaches is to limit the number of copies of each message that are produced [10, 11]. In section 2.2 we will explain in more detail why this approach can be desirable. Let us refer to the ratio of the number of nodes who carry a message to the total number of nodes as the **carrier fraction**. Several possible distribution schemes for achieving a target carrier fraction are given in [10]. These approaches may be inadequate for some applications because they assume prior knowledge of the number of nodes in the system. The proposed "Spray and Wait" protocol [11] includes a clever technique for estimating the number of nodes in the system in order to compute the number of message copies that should be distributed. However this protocol could be complicated to implement and debug, and once a message has been released it will be unable to compensate for potential changes in the number of nodes. It also assumes that the nodes' mobility patterns are iid. In situations where node mobility is not iid it is possible that "relays" may never encounter the destination and that the "wait" phase could be indefinite.

In each of these approaches the nodes know more than they need to about the network. We describe a simple process wherein each node is ignorant of the number of nodes in the group, but still manages to achieve the same goal of limiting the carrier fraction. Our method will effectively adjust the number of carriers if the number of nodes in the system changes. It is only marginally more complex than epidemic

routing and requires a very small amount of memory which depends linearly on the number of messages carried.

Another complementary approach to optimizing the distribution of message copies is to use **heuristic protocols** which only pass message copies to nodes which are considered more likely to bring the message closer to its destination. The details of such protocols can become very complex. The simplest type of heuristic protocol is the **static hierarchy** where the nodes are manually organized into tiers based on how likely they are to deliver a message to some common destination. For example in [9] the nodes are organized into a hierarchy of slow-moving or stationary sensors and highly mobile "mules" which aggregate messages from the sensors and deliver the data to access points. We will elaborate on this in section 4.1.

We will show how the principles of the Simple Counting protocol can be used in conjunction with heuristic DTN protocols to effectively "clean-up" unnecessary copies of a message once that message has established itself in a population of nodes which are closer to the message's destination. This technique, which we call **Intermediate Immunity**, has aspects in common with the idea of custody transfer [1] but on a collective scale.

In section 2 we will state our assumptions and explain how Simple Counting works. In section 3 we model the behavior of Simple Counting and compare to simulation results under different mobility models. In section 4 we will describe how Intermediate Immunity works, and in section 5 we will give results from a variety of simulated example scenarios.

2. THE SIMPLE COUNTING PROTOCOL

2.1 Assumptions

DTNs are often viewed as mostly connected mobile wireless networks that suffer from intermittent disruptions and segment isolation. We take another, not uncommon, view of the DTN as a sparse disconnected collection of nodes that may occasionally come together in pairs or small groups. We model this situation using discrete space and discrete time. In this model each node occupies a cell in the grid and can communicate only with other nodes in its cell. A model like this seems like a reasonable approximation for very sparse networks such as the Saami Network or Zebra Net [4, 3], or in a denser urban environment where the nodes may have short range and be generally surrounded by obstructions. We do, however assume that the network is dense enough that encounters will take place. The actual encounter rate will determine the time scale on which data propagates.

2.2 Why Limit the Carrier Fraction?

Given a collection of nodes and a message to be delivered to some destination, we expect the probability of a delivery taking place to be roughly doubled if there are two message carriers instead of one. Likewise, if there are N nodes in the system and $n \ll N$ of them carry a copy of the message, we expect the probability of delivery to be roughly n times what it would be for only one carrier. As n approaches the same order of magnitude as N, however, we expect to see diminishing benefits from each additional carrier. An example of this would be a geometric process with success probability p. The expected waiting time is $\frac{1}{p}$. If $p = \frac{n}{N}$ then changes in n make very little difference in the waiting time if n is close to N. Therefore if our nodes' buffer space is at a premium it makes sense to limit the number of carriers to only a fraction of the nodes. Cutting the carrier fraction in half will drastically reduce buffer usage in the system as a whole, but is unlikely to have a large effect on delivery time.

2.3 The Protocol

The Simple Counting Protocol works as follows for each message $\mathcal{M}$. We assume that nodes exchange some form of summary vector at the beginning of each meeting just as in the epidemic protocol [12].

- Each carrier of the message $\mathcal{M}$ keeps two counters

  $n_c = \#$ of consecutive non-carriers of $\mathcal{M}$ seen

  $n_d = \#$ of consecutive carriers of $\mathcal{M}$ seen

- If $n_c \geq C$ for some threshold $C \geq 0$ then hand off a copy of $\mathcal{M}$ and reset $n_c = 0$.

- If $n_d \geq D$ for some threshold $D > 0$ then drop your copy of $\mathcal{M}$ and reset $n_d = 0$.

Note that we always have either $n_c = 0$ or $n_d = 0$ so it isn't really necessary to keep two separate counters. In practice we just keep a single counter and represent n_c with positive values and n_d with negative values. Also we can assume that the copy threshold C and the drop threshold D will be less than 100, so it suffices for each node to keep a signed byte for each message it carries.

It is impossible for a message to be completely eradicated before it is delivered. When two nodes meet and exchange summary vectors, one of the nodes will always have to be the initial sender. The initial receiver must update its counters and delete the necessary messages before it offers its summary vector to the initial sender. If two carriers come together and the initial receiver decides to drop its copy of a message, the initial sender will perceive that node as a non-carrier of that message and will therefore be unable to drop its copy. Thus, only one copy of a message can be deleted in any encounter. Since a copy can only be deleted when two carriers meet, this can never result in a message being completely eradicated before it is delivered. Note that this reasoning does not apply to messages that are being dropped due to storage constraints. There are some small changes that can be made to this protocol to slow down message eradication in situations where buffers are full, but we choose to focus on the simplest version here.

One potential benefit of this technique is that the set of message carriers changes over time. The copy-amortization techniques in [10] are geared towards achieving more diverse spatial distribution of message copies, as is the "binary spray and wait" scheme in [11]. They make a special effort to make sure that the eventual message carriers are not just the first ones that the source sees. In our scheme the available copies are constantly *percolating* through the nodes. Even if the nodes' mobility patterns are not iid, this percolation automatically gives us greater spatial diversity. The drawback of this is that more energy is spent on transmissions than if we just picked a fixed subset of nodes to be carriers. We will quantify this in section 3.2.1.

Note that Simple Counting with $C = 0$, $D = \infty$ is equivalent to traditional epidemic routing.

3. ANALYSIS

First we study the steady-state properties of Simple Counting under uniform mobility. Then we will address other mobility models and some transient properties.

3.1 Mobility Models

In our analysis we will refer to a number of mobility models so we take a moment to summarize what they are. Our analysis and simulations use a discrete toroidal grid and discrete time. We often refer to grid points that can be occupied by nodes as "cells".

- Uniform (**UNI**)
  If the simulation space is a $K \times K$ grid, then at any time step each node has a uniform $1/K^2$ probability of being in any cell on the grid, independent of its previous position.

- Random Walk (**RAW**)
  At each step a node makes a list of its neighboring cells and chooses its next position uniformly from those neighbors.

- Random Direction (**RDIR**)
  Each node chooses a velocity uniformly from some interval $[v_{min}, v_{max}]$, a direction uniformly from the interval $[0, 2\pi)$ and a duration uniformly from some interval $[r_{min}, r_{max}]$. Then it travels in the chosen direction at the chosen velocity for the chosen amount of time [7].

- Localized Random Walk (**LRAW**)
  Each node is assigned a fixed **home cell**. At each step each node makes a list of its neighboring cells and chooses one with multinomial probability depending on each cell's distance from the node's home cell. In particular

$$Prob(\text{choosing cell } i) \propto e^{-d_i/2\tau}$$

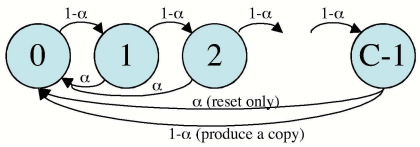
  where d_i is the taxicab distance from cell i to the home cell and τ is some positive parameter we call the "tightness parameter". It turns out that a node following the LRAW mobility model will have a double exponential (or Laplace) stationary distribution about the home cell. We will pull in a variety of properties of this mobility model, though a complete analysis is beyond the scope of this paper [13].

3.2 Steady-State Analysis

The only two parameters this scheme takes are the copy threshold, C, and the delete threshold, D. We would like to compute the expected carrier fraction achieved at equilibrium for parameters (C, D). We assume there are N nodes in the system and (for the present analysis) assume that at any time every node has an equal probability of encountering any other node. This is essentially assuming that the nodes have a uniform mobility model.

Let $\alpha \in [0, 1]$ be the carrier fraction. That is, there are αN carriers in a system with N nodes. Then for any particular node we can model its copy and drop counters, n_c and n_d, as Markov processes. Let $\pi_j(\alpha)$ be the probability that the copy counter $n_c = j$ given carrier fraction α. Similarly, let $\rho_j(\alpha)$ be the probability that the drop counter $n_d = j$.

First consider the copy counter, n_c, which tracks the number of consecutive non-carriers the node has seen. Upon seeing the C^{th} consecutive non-carrier the node makes a copy and resets n_c, so the counter can take any value from 0 to $C - 1$. In any encounter, the probability of seeing a non-carrier and increasing the counter is $(1-\alpha)$. The probability of seeing a carrier and resetting the counter to 0 is α.



This model is represented by the $C \times C$ matrix

$$A = \begin{pmatrix} \alpha & \alpha & \dots & \alpha & 1 \\ 1-\alpha & 0 & \dots & 0 & 0 \\ 0 & 1-\alpha & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1-\alpha & 0 \end{pmatrix} \tag{1}$$

This model is ergodic and has a stationary solution $A\mathbf{x} = \mathbf{x}$. The only state we are really interested in is the $(C-1)$th state which is occupied with probability

$$\pi_{C-1}(\alpha) = \frac{\alpha\,(1-\alpha)^{C-1}}{1-(1-\alpha)^C} \tag{2}$$

For any carrier fraction, α, $(\alpha\,N)(1-\alpha)\,\pi_{C-1}(\alpha)$ is the expected rate at which copies are being produced. Similar reasoning tells us that $(\alpha\,N)\alpha\,\rho_{D-1}(\alpha)$ is the expected rate at which copies are being dropped, where

$$\rho_{D-1}(\alpha) = \frac{(1-\alpha)\,\alpha^{D-1}}{1-\alpha^D} \tag{3}$$

is the probability of the delete counter, n_d, having value $D-1$.

To compute the steady state of the system we reason that the rate at which copies are being produced must equal the rate at which copies are being dropped. This is true when $(1-\alpha)\,\pi_{C-1}(\alpha) = \alpha\,\rho_{D-1}(\alpha)$. More explicitly, when

$$\frac{(1-\alpha)\,\alpha^D}{1-\alpha^D} = \frac{\alpha\,(1-\alpha)^C}{1-(1-\alpha)^C} \tag{4}$$

We do not know of a way to get a general closed-form solution for α as a function of C and D. But $\alpha\,\rho_{D-1}(\alpha)$ is monotonically increasing from 0 on $[0, 1]$, and $(1-\alpha)\,\pi_{C-1}(\alpha)$ is monotonically decreasing to 0 over the same interval. Therefore for any given C and D there exists a unique equilibrium point and it is simple enough to solve numerically for α. Figure 1 shows an example for $D = 3$ and $C = 5$.

We are interested in how accurately this protocol achieves the desired carrier fraction. This will depend on the mobility pattern of the nodes. In our analysis we assumed that probability of a carrier node encountering another carrier was equal to the global carrier fraction. This is the case if the nodes follow a uniform mobility model, and it turns out to be a good approximation for the other mobility models we consider. We make the following brief and intuitive argument for why our analysis will be applicable even for localized mobility models:

Definition: By a **neighborhood** of nodes we mean a collection of two or more nodes among which the encounter

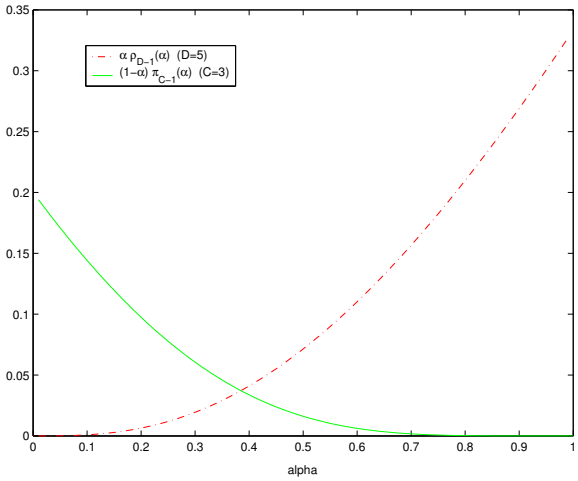


Figure 1: Plot of $(1-\alpha)\,\pi_{C-1}(\alpha)$ and $\alpha\,\rho_{D-1}(\alpha)$ as functions of the carrier fraction, α. The value of alpha where the two curves intersect is the equilibrium point, where the expected rate of copy creation is equal to the expected rate of copy deletion.

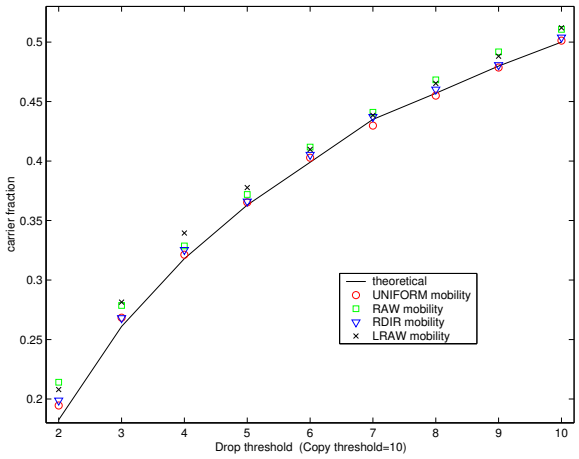


Figure 2: Predicted carrier fractions compared to carrier fractions observed in simulations for several mobility models. In all cases $C = 10$ and D is allowed to vary between 2 and 10. We omit the error bars to keep the plot legible. The experimental standard deviations were always in the range of 2-4%.

pattern is approximately uniform. That is, each node is approximately equally likely to encounter any other node in its neighborhood.

Assumption: If the expected carrier fraction for every neighborhood of nodes is α, then the expected global carrier fraction for all nodes in the system is α.

For example, consider a set of nodes with random walk mobility. If one draws a circle that only encloses a few nodes, the set of nodes enclosed by the circle would be a neighborhood of nodes. They are all approximately the same distance from each other, and therefore their encounter probabilities are approximately equal. Now, if the carrier fraction among the nodes in any such circle were about α, the carrier fraction of all nodes in the system would also be about α.

It is possible to draw node arrangements where this is not true, but such exceptions will be rare and short-lived. A lot more could be said about this, but we appeal to the reader's intuition and the consistency of our results. Figure 2 compares predicted equilibrium carrier fractions to simulation results for several mobility models. In all cases the predicted carrier fraction is fairly close to that observed. This highlights the flexibility and robustness of Simple Counting.

3.2.1 *The cost of message percolation*

The fact that the message carriers are constantly changing can be seen as a benefit. Under non-uniform and non-iid mobility models this *percolation* will usually ensure that all nodes will get a turn being message carriers. The cost of this percolation is that Simple Counting will consume energy on transmissions even after the desired carrier fraction has been reached. We can use the analysis in this section to quantify this.

Suppose the mean pair encounter rate is R and the equilibrium carrier fraction is α_0. Then the expected rate at which message copies are being transmitted at equilibrium is $R\alpha_0 N(1-\alpha_0)\pi_{C-1}(\alpha_0)$. This turns out to be fairly small in most situations, however it is an aspect that could be improved upon.

3.3 Transient Analysis

Even though different mobility models do not have a great impact on the steady-state properties of the Simple Counting protocol, they will have a great effect on its transient properties. That is, the speed and manner with which the system reaches equilibrium. One thing that is true in general is that the Simple Counting protocol will slow down the spread of messages relative to epidemic. Even when the carrier fraction is very small, a node must encounter C consecutive non-carriers before it is allowed to hand off its first copy. This could be viewed as benefit or a detriment. In some situations it might make sense to slowly increase the carrier fraction in the hopes that the destination is nearby and that the message will be delivered directly without wasting too much energy and storage. In other situations one may want to reach the equilibrium fraction as fast as possible. In this case a node might produce as many copies as possible immediately after the message is released and then let the Simple Counting protocol rein in the carrier fraction, analogously to TCP "slow-start".

Another small modification to Simple Counting is what we call "fast mode", where a message carrier does not reset its n_c counter after handing off a message copy. The analysis of these sorts of variations is almost exactly the same as for plain Simple Counting. We restrict our analysis here to the simplest protocol.

3.3.1 *Transient properties with UNI*

The goal of this analysis is to derive a function $c(t)$ which approximates the expected number of copies of a message in circulation at time t. This process can be modeled as a differential equation. As a starting point consider the epidemic protocol (equivalent to the simple counting protocol with $C \leq 1$ and $D = \infty$). Uniform epidemic spread is modeled by:

$$\frac{d}{dt}c_{epidemic,UNI}(t) = NRc\left(1 - \frac{c}{N}\right) \qquad (5)$$

where R is the pair encounter rate of the nodes and N is the number of nodes in the system. Under the uniform mobility model on a $K \times K$ grid we have $R = 1/K^2$. The factor c is the number of carriers at time t and the factor $(1 - c/N)$ is the probability of encountering a non-carrier. This is solved by the logistic equation:

$$c_{epidemic,UNI}(t) = \frac{Ne^{NRt}}{N + (e^{NRt} - 1)} \tag{6}$$

The situation under the simple counting protocol is complicated by two factors:

- Not all carriers are eligible to hand off a copy.
- Message copies may be dropped.

Luckily eqn 2 provides the expected fraction of carriers that are eligible to create message copies as a function of total carrier fraction[1]. To compensate for the first issue we replace c in the logistic differential with $c\,\pi_{C-1}(c/N)$. We also add a term representing the rate at which message copies are being dropped. This is almost the same as the copy-creation term except we use $\rho_{D-1}(c/N)$ which represents the probability that an arbitrary carrier is eligible to drop its message copy, and we replace $(1 - c/N)$ with c/N. Combining the copy creation and copy deletion terms and factoring we get:

$$\frac{d}{dt}c_{SC,UNI}(t) = NR\,c\left(\left(1 - \frac{c}{N}\right)\pi_{C-1}(c/N) - \frac{c}{N}\,\rho_{D-1}(c/N)\right) \tag{7}$$

This will not have a closed form solution, but we can solve it numerically for reasonable initial conditions.

3.3.2 Transient properties with LRAW mobility

We summarized the Localized Random Walk mobility model in section 3.1. Part of the rationale for naively limiting carrier fraction, as the simple counting protocol does, is a scenario where the source belongs to a group of nodes that is spatially restricted, relying on highly mobile "MULE" nodes [9] to pass by and pick up message copies. In situations like this, limiting the carrier fraction can reduce buffer usage in the spatially restricted group of nodes by an order of magnitude while having little impact on the message pickup probability.

When several nodes obeying the LRAW mobility model share the same home cell we will refer to them as an "**LRAW cloud**". It turns out that the expected pair encounter rate in an LRAW cloud with tightness parameter τ can be computed exactly [13].

$$R = \frac{1}{16\tau^2} \tag{8}$$

To model SC in an LRAW cloud, the differential equation in the last section must also be modified to account for the fact that the nodes are not uniformly mixed. That is, as message copies spread, carriers are more likely to be near other carriers and non-carriers near other non-carriers. In the uniform case the probability of a carrier seeing a non-carrier was $(1 - c/N)$. In the case of an LRAW cloud it will be less. In particular we have found that replacing the linear $(1 - c/N)$ term with $2(1 - c/N)/(NRt + 1)$ gives a good approximation. So epidemic message spread in an LRAW cloud is approximated by the differential equation:

$$\frac{d}{dt}c_{epidemic,LRAW}(t) = \frac{2NR\,c}{NRt + 1}(1 - c/N) \tag{9}$$

which has a closed-form solution which is invertible to give t as a function of c. We will write $\tilde{t}(c)$ to denote the function obtained through this inversion.

$$c(t) = N\frac{(NRt + 1)^2}{N + (NRt + 1)^2 - 1} \tag{10}$$

$$\tilde{t}(c) = \frac{1}{NR}\sqrt{\frac{c\,(N - 1)}{N - c} - 1} \tag{11}$$

Just as in the uniform case we adjust the $c(t)$ differential equation to include both the copy-creation and the copy-deletion processes. The resulting differential equation is:

$$\frac{d}{dt}c_{SC,LRAW}(t) = \frac{2NR\,c}{NR\tilde{t}(c) + 1}\left(\left(1 - \frac{c}{N}\right)\pi_{C-1}(\frac{c}{N}) - \frac{c}{N}\,\rho_{D-1}(\frac{c}{N})\right) \tag{12}$$

We can solve this numerically. Figure 3 compares experimental $c(t)$ results to numerical solutions for the differential equations just given for UNIFORM and LRAW mobility models. The figure illustrates how the mobility model affects the transient properties of the message spread. The differential equations do a reasonable job of capturing the differences in the transient properties of the mobility models.

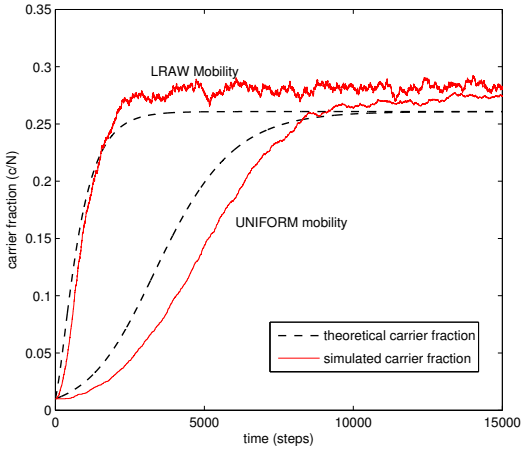


Figure 3: Plot of simulated and theoretical $c(t)$ curves for LRAW and UNIFORM mobility models using Simple Counting with $C = 10$, $D = 3$. All plots are for N=100 nodes. The uniform results are for a 100×100 grid and the LRAW results are for $\tau = 10.0$. The simulation results are averaged over 50 runs.

4. INTERMEDIATE IMMUNITY

In this section we explain how the same principles that make the Simple Counting protocol work can be used in conjunction with a heuristic protocol to reduce the resource usage in a DTN. The idea is to "clean up" message copies

[1]This isn't completely accurate for the protocol we have described. The actual number of carriers eligible to make copies lags the instantaneous carrier fraction because all new carriers are initialized with $n_C = 0$, $n_D = 1$.

from nodes which are considered further away from the destination. These more distant nodes are made immune to the message before it is actually delivered. The judgment regarding when such an immunity is justified is made based on a criterion very similar to that used in Simple Counting.

4.1 A Note on Heuristic Protocols

By "heuristic protocol" we mean a protocol where nodes have some notion of their "distance" from the destination of a message. That is, some scalar quantity that reflects the likelihood that giving that node a message copy will lead to the message being delivered. Designing such protocols is a broad and difficult problem. Solutions are generally tuned to a particular type of scenario. Some examples would be

- A mobile wireless network where the nodes have GPS information and they can compute their physical distance from any particular destination.
- A *static hierarchy* where certain classes of nodes are pre-programmed to be considered more likely to deliver a message to a common destination. This might be the case in a data-gathering sensor network such as that described in [9] or [14].
- A logical metric based on past encounters such as that described in [4].
- More complicated heuristics such as [6] or [2]

For our experiments we use a multi-tiered static hierarchy. All nodes will be pre-programmed with a routing table accurately reflecting their distance from the destination. We are really assuming the best case scenario for these experiments. We have also done experiments with heuristics derived from encounter histories, but the amount of analysis possible with such protocols is beyond the scope of this paper.

4.2 Measuring Protocol Efficiency

One way to measure the efficiency of a DTN protocol is to look at the total amount of buffer space a message takes up in the system as a whole (the size of the message times the number of copies in circulation), and the amount of time it occupies that memory for.

As before we denote the number of copies of the message in circulation at time t by $c(t)$. Then if we treat time as discrete (as we do in our simulations) the **Time-Weighted Network Storage** (TNS) [8] of a message of size s is

$$T = \sum_{t=t_0}^{\infty} s\, c(t) \tag{13}$$

We believe that minimizing time-weighted network storage is a reasonable measure of the efficiency of a delay-tolerant protocol. On one hand it induces us to keep the number of copies low, thereby conserving buffer space. On the other hand it induces us to keep latency reasonable, since even if there is only one copy of a message in circulation, its impact on the network could become very large if it persists in the system for a very long time.

4.2.1 Controlling TNS

If we want the time-weighted network storage of a message to be finite, we must institute some mechanism that will eventually eliminate all copies of the message from all nodes' buffers. The simplest approach is to use a Time-to-Live (TTL), whereby each message is stamped with a creation time and all carriers of a message drop their copies after a certain time limit. This has the benefit of being relatively easy to implement and possibly easier to analyze, since (13) has finite endpoints. The main drawbacks are that the TTL method may drop the message before it is delivered and the TTL value must be carefully chosen based on assumptions about the network. If the latency distribution is multimodal or has a high variance there may be no reasonable way to choose a TTL value. Therefore TTL-type protocols will lack flexibility and robustness.

We prefer the commonly used idea of a vaccine, or anti-packet as described in [10]. The idea is that when a message is delivered, the destination releases a piece of meta-data called an **anti-packet** for that message. Any node which receives the anti-packet drops its copy of the message. It is assumed that the anti-packets spread in an epidemic manner. Realistically the anti-packets would have to eventually expire. Otherwise the network capacity would be eaten up by the overhead of exchanging longer and longer lists of anti-packets. We will ignore that issue and assume that anti-packets persist at least until all message copies are totally eradicated.

When such a vaccination system is instituted in conjunction with epidemic message spread we expect the number of message copies to increase initially, then decrease more or less symmetrically once the message is delivered and the anti-packet is released. Figure 6 depicts this process for a population of nodes with LRAW mobility and uniformly randomly placed home cells. The TNS of this experiment would be the area under the curve.

4.3 Details of Intermediate Immunity

The idea behind Intermediate Immunity is that the nodes can be logically divided into **levels** based on their heuristic distance from the message destination. Specifically, we will say that two nodes belong to the same level relative to some destination if their heuristic distances from that destination are the same or very close. The destination is always assumed to be at level 0 relative to itself, and nodes belonging to higher levels are less likely to meet the destination. Within a node's own level it manages the number of copies using Simple Counting. That is, only nodes in the same level affect a node's carrier counter, and a node distributes copies within its own level only when its carrier counter reaches the copy threshold, C. The nodes *always* pass message copies to nodes in levels is closer to the destination, but *never* to nodes which belong to levels further from the destination.

To institute Intermediate Immunity we introduce the **immunity threshold**, I. When a node encounters I consecutive carriers of a message within its own level it releases an "intermediate anti-packet" for some more distant level. If a node's heuristic distance to the destination is l, it can release an anti-packet for all nodes of distance at least $l + g$ where we will call g the **immunity lag**. Under Intermediate Immunity an anti-packet must contain at least some identifier for the message that it immunizes against and a scalar indicating what level the immunity is for. When two nodes meet and exchange anti-packets, the immunity for the lower (closer) level always supersedes immunity for the higher (more distant) level.

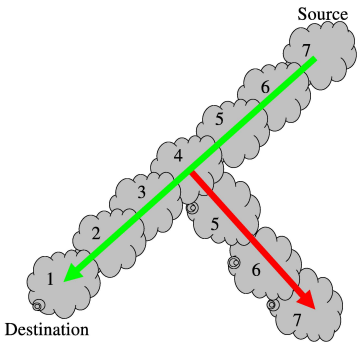


Figure 4: The branched chain scenario. Each cloud in the diagram represents a collection of about 50 LRAW nodes sharing a common home cell.

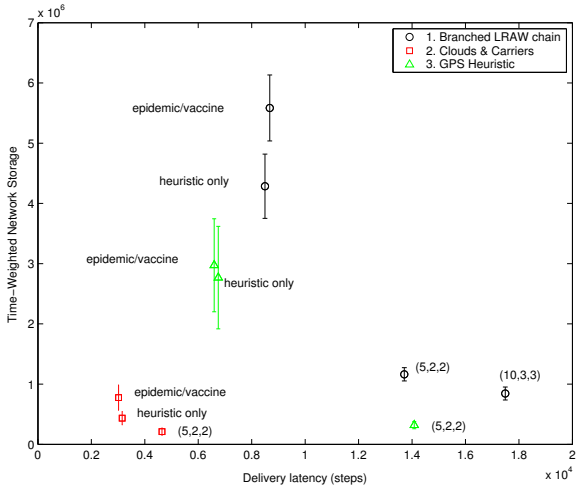


Figure 5: The TNS of epidemic routing, heuristic protocol alone, and Simple Counting + Intermediate Immunity (SC+II) for the three scenarios described here. Data points are averaged over 50 runs and error bars are one standard deviation. The SC+II data points are labeled with their counting thresholds: (C, D, I)

5. EXPERIMENTS AND RESULTS

5.1 Branched Chain

Consider the simple scenario of a branched chain of LRAW clouds (Fig 4). This is a chain in the sense that the home cells of the LRAW clouds are close enough that the nodes from adjacent clouds will occasionally meet and propagate the message from one cloud to the next. The message source and destination are at opposite ends of the chain. The branch exists as a potential wrong turn that the message could take were there no routing heuristic in place. The nodes are pre-programed with heuristic distances to give a 8-tiered static hierarchy. The destination node is always at level 0 relative to itself. The nodes in the same cloud as the destination are assigned level 1. The nodes in the next cloud are assigned level 2, and so on until we reach the ends of the chain.

The message is initially released by a node in the source cloud and the propagation of the message within cloud 7 is controlled by Simple Counting. When a message carrier in cloud 7 meets a node from cloud 6, however, it always hands off a message copy. Similarly from cloud 6 to cloud 5. In this example we set the immunity lag $g = 2$, so when a carrier in cloud 5 encounters enough consecutive carriers within its own cloud, it can release an anti-packet for levels 7 and higher. This anti-packet spreads backwards to cloud 7 and eradicates all message copies from those nodes.

Since the anti-packet tends to spread more quickly than the message itself we expect that message copies will only reside in 2 to 3 clouds at a time, and in those clouds the carrier fraction is limited by Simple Counting. One would expect that this would reduce the time-weighted network storage relative to epidemic or even a heuristic protocol with standard immunity. It is conceivable, though, that the added latency of simple counting will outweigh the gains made by reducing the carrier fraction. We find experimentally that this is not the case. Simple Counting and Intermediate Immunity reduce the TNS by a factor of two relative to the heuristic protocol alone. Figure 5 shows a comparison of TNS and latency for these two possibilities and also epidemic message routing with anti-packets.

5.2 Clouds and Carriers

Suppose we have four LRAW clouds, all far enough apart that the probability of their nodes meeting is effectively 0. The message source will reside in one cloud and the message destination in another cloud. Suppose also that there are some number of highly mobile "carrier" nodes which we will model as having Random Direction (RDIR) mobility. Again, the destination has level 0 relative to itself and the other nodes in its cloud have level 1. All RDIR carrier nodes are assigned level 2, and the non-destination clouds are all assigned level 3.

Once the message is released in the source cloud the carrier fraction is controlled by Simple Counting. We have observed that reducing the carrier fraction in the source cloud saves buffer space without increasing the pick-up latency by very much. Similarly, once a copy of the message is passed by an RDIR carrier to the destination cloud, the carrier fraction in the destination cloud is controlled by Simple Counting, reducing buffer usage without delaying message delivery by very much.

Since there are effectively only three "levels" in this scenario the gains we see come mainly from the effects of Simple Counting and are not as dramatic as the other two scenarios.

5.3 Uniformly Placed LRAW Nodes

Suppose we have a collection of nodes with LRAW mobility and home cells placed uniformly randomly on a grid. We will assume that each node has some rough GPS ability in that it knows the Cartesian distance from its home cell to the destination's home cell. In this example this Cartesian distance will act as the routing heuristic. Nodes will treat other nodes as belonging to their same level if their Cartesian distance to the destination is within $\delta = 20$ of their own. In effect the levels relative to the destination are arranged in concentric circles about the destination node. The "cleaning up" effect of intermediate immunity is particularly effective in this scenario because of the large size of the more distant levels.

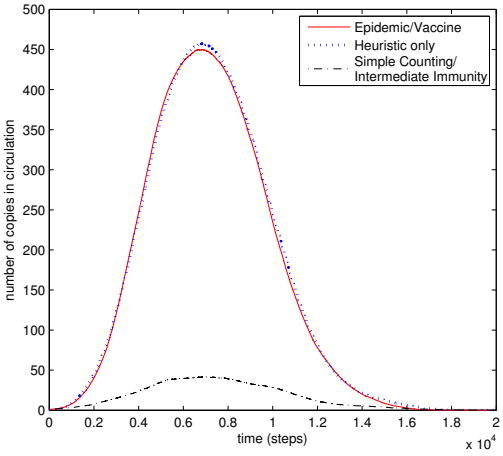


Figure 6: The number of message copies in circulation as a function of time for the uniformly placed LRAW nodes with GPS scenario. Data points are averaged over 50 runs.

Figure 6 shows the number of copies in circulation as a function of time for this scenario. The plot provides a visually dramatic illustration of the resource conservation of Simple Counting + Intermediate Immunity relative to epidemic routing or the heuristic protocol alone.

6. CONCLUSIONS AND FUTURE WORK

Simple Counting provides a robust method for controlling the carrier fraction with little more complexity than epidemic routing and negligible memory. Controlling the carrier fraction is desirable because the expected increase in latency is relatively small compared to the reduction in resource usage gained by reducing the number of message copies in circulation.

Intermediate Immunity is a general technique that can be combined with a heuristic protocol to effectively "clean up" message copies from more distant nodes as the message approaches its destination. Though the design and implementation of heuristic protocols is a difficult and complex problem, the Intermediate Immunity component is a simple addition and is based on the same principles as Simple Counting, and can therefore be analyzed in much the same way.

Some ares of future research include:

- Modeling of latency and reliability.
- Variations on Simple Counting. Using only two thresholds, C and D, restricts the possible carrier fractions attainable with Simple Counting. Other possibilities can be achieved by alternating the thresholds or performing hand-offs and drops according to a stochastic function of the carrier counters.
- Experimentation with more realistic or graph-based mobility models. We used only fairly abstract mobility models. More realistic models have proposed and implemented in simulations [5].
- Experimentation with other heuristic protocols. The results we presented here are for the simplest possible heuristic protocol. We have done some experiments with heuristic protocols in which the nodes try to deduce their logical distance from the destination based on past experience. The huge variability of such experiments make this a separate research effort.

7. ACKNOWLEDGMENTS

We thank Matt Seligman of LTS for reading a draft of this paper and sharing his thoughtful advice and DTN expertise.

8. REFERENCES

[1] K. Fall. A delay-tolerant network architecture for challenged internets. In *ACM SIGCOMM 2003*, 2003.

[2] S. Jain, M. Demmer, R. Patra, and K. Fall. Using redundancy to cope with failures in a delay tolerant network. In *SIGCOMM '05*, 2005.

[3] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet. In *ASPLOS, San Jose, CA*, Oct. 2002.

[4] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):19–20, 2003.

[5] M. Musolesi, S. Hailes, and C. Mascolo. An ad hoc mobility model founded on social network theory. In *MSWiM '04*, 2004.

[6] M. Musolesi, S. Hailes, and C. Mascolo. Adaptive routing for intermittently connected mobile ad hoc networks. In *IEEE WOWMOM '05*, 2005.

[7] P. Nain, D. Towsley, B. Liu, and Z. Liu. Properties of random direction models. In *INFOCOM 2005*, 2005.

[8] M. Seligman, K. Fall, and P. Mundur. Storage routing for dtn congestion control. *Wireless Communications and Mobile Computing*, 7(5), 2007.

[9] R. C. Shah, S. Roy, S. Jain, and W. Brunette. Data mules: Modeling a three-tier architecture for sparse sensor networks. In *Proceedings of the 2003 IEEE Workshop on Sensor Network Protocols and Applications*, 2003.

[10] T. Small and Z. J. Haas. Resource and performance tradeoffs in delay-tolerant wireless networks. In *ACM SIGCOMM WDTN '05*, 2005.

[11] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *ACM SIGCOMM WDTN '05*, 2005.

[12] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, 2000.

[13] B. Walker, T. C. Clancy, and J. Glenn. Using localized random walks to model delay-tolerant protocols. In submission.

[14] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *MobiHoc '04*, 2004.