

# Towards Distributed Network Classification for Mobile Ad hoc Networks

(Invited Paper)

Dimitrios Antonellis  
Electrical Engineering Dept  
University of Southern  
California  
dimitria@usc.edu

Ahmed Mansy  
College of Computing  
Georgia Institute of  
Technology  
amansy@cc.gatech.edu

Konstantinos Psounis  
Electrical Engineering Dept  
University of Southern  
California  
kpsounis@usc.edu

Mostafa H. Ammar  
College of Computing  
Georgia Institute of  
Technology  
ammar@cc.gatech.edu

## ABSTRACT

Mobile ad hoc networks range from traditional MANETs where end-to-end paths exist from sources to destinations, to DTNs where no contemporaneous end-to-end paths exist and communication is achieved by the store, carry, and forward model of routing. Hence, nodes of these networks need to identify the level of connectivity of the network they belong to and classify it as a MANET or a DTN, in order to properly select appropriate protocols to achieve end-to-end communication. What is more, since mobile ad hoc networks change over time and space, nodes need to periodically re-access their network classification to adapt to the always changing environment.

Recently, there has been an effort to classify the various types of mobile ad hoc networks assuming there is a centralized authority that has complete knowledge of the network and its dynamics. In this paper we design distributed mechanisms for nodes to perform the above classification on the fly, based only on local information that they collect as they move and encounter other nodes. The mechanisms take advantage of a combination of measurements and analytical techniques. We investigate the accuracy of our mechanisms by comparing the network classification of the centralized authority to that of a node using our schemes.

## Categories and Subject Descriptors

C.2.m [Computer-Communication Networks]: Miscellaneous

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

WICON'08 November 17-19, 2008, Maui, Hawaii, USA  
Copyright 2008 ACM ...\$5.00.

## General Terms

Theory, Design

## Keywords

classification, mobile ad hoc networks, intermittently connected networks, routing

## 1. INTRODUCTION

Mobile wireless networks have gained a lot of interest recently in the research community. However, because of the different node mobility patterns and density of nodes, such networks have different connectivity properties. In traditional Mobile Ad Hoc Networks (MANETs) it is assumed that end-to-end paths exist from any source to any destination most of the time. This does not occur in the case of Delay Tolerant Networks (DTNs), where the packets may need to get buffered while waiting for a link to come up. In this sense, in DTNs there exist space-time paths which are paths that connect the nodes over time.

If we apply a routing protocol designed for MANETs to a DTN, it is going to fail. On the other hand if we apply a routing protocol designed for DTNs to a MANET, even if the packets will eventually be received by the destination, the delivery delays will be really large. Motivated by this, the question that we address in this paper is how to decide the class that the mobile network belongs to, in order to use a corresponding routing protocol.

Recently, there has been an effort to classify the various types of mobile ad hoc networks assuming there is a centralized authority that has complete knowledge of the network and its dynamics [4]. The authors in this paper provide an algorithm that classifies mobile wireless networks into three classes: SPN (space-path networks), U-DTN (unassisted delay tolerant networks) and A-DTN (assistance-needed delay tolerant networks). This approach provides an interesting insight on the way that mobile networks operate but it is not practical because of the assumption of the centralized mechanism doing the classification. Another interesting work in this area is that of [8] in which the authors define formally a

number of mobile network classes as well as classes of routing mechanisms. Then, they prove a number of results about the network classes that each routing mechanism class can solve.

In this paper, we make the first step towards extending the work of [4] to determine the class that a mobile network belongs to in a distributed fashion and using only information that the nodes have up to that point in time. Since the mobile nodes usually have limited energy, computation capability and storage, it is preferable that the number of computations at each node is small. So, our approach is based on an analytical framework which is combined with some simple node observations in order to decide the class of the network. Note that in this paper we are interested only on the appropriate decision of the class that the mobile network belongs to, and it is up to the network designer to define the way that the nodes can then select the most appropriate routing protocol for this network class.

The rest of the paper is organized as follows. Section 2 presents the centralized algorithm that has been presented in the literature [4] for the classification of the wireless and mobile networks as well as the definition of the mobile network classes that we use. In Section 3, we present the distributed classification algorithm that uses a number of analytical formulas and simple node observations to decide the class that the network belongs to. Section 4 provides the derivations of the formulas that are used throughout the distributed classification algorithm. Then, in Section 5 we present the results of the distributed classification algorithm applied to a number of mobile networks and we compare the accuracy of the results with the ones that we get from the centralized classification algorithm. Finally, we conclude with some remarks and a brief discussion of future work directions.

## 2. CLASSIFICATION OF MOBILE AD HOC NETWORKS

In this section we define the three classes of mobile networks that we will use throughout the classification process and present the main aspects of the centralized classification algorithm presented in [4]. We will use the results from this algorithm to compare the accuracy of the proposed distributed classification algorithm.

### 2.1 Connected Networks (*Conn.*)

We use the term *Conn.* networks to refer to the traditional Mobile Ad hoc networks (MANETs) where it is assumed that the network is connected most of the time. This means that at each timeslot there is an end-to-end path (space path) that connects every pair of source and destination. Also, it is implicitly assumed that the links do not change that fast, which entails that the routes between sources and destinations do not alter that much. The most common routing protocols that are used in this class are AODV [1] and DSR [6].

### 2.2 Intermittently Connected Nets (*Int. Conn.*)

In the case of the *Int. Conn.* networks, no contemporaneous end-to-end paths exist most of the time and communication is achieved by the store, carry, and forward model of routing. The authors in [4] call such paths space-time paths to distinguish them from the contemporaneous space paths used in MANETs. Many routing protocols have also been

proposed for this class of mobile networks such as Epidemic Routing [13] and Spray and Wait [10].

### 2.3 Disconnected Networks (*Disconn.*)

Such networks are really sparse and the mobility of the nodes doesn't allow them to communicate even through space-time paths. Actually, the lengths of the space-time paths are too long. In this class of mobile networks it is preferable to use additional mobile nodes that move around the network area collecting messages and transferring them to the destination nodes. Such message ferrying approaches are presented in [7] and [14].

Let's now define a number of quantities that the centralized classification algorithm uses:

- $NCC_i$ : the number of connected graph components at timeslot  $i$ .
- Link Persistence: the average duration that a link spends from its inception to its outage.
- Journey: the sequence of edges traversed in a space-time path with the corresponding timeslots of the traversal.
- Foremost Journey from node  $a$  to node  $b$ : a journey starting at node  $a$  with destination node  $b$  that has the earliest arrival time.
- Longest Foremost Journey  $LFJ_i$ : the maximal duration that a foremost journey will take from any origin to any destination starting at timeslot  $i$ .

The algorithm also uses the following parameters that are given as input to it:

- $\gamma$ : maximum acceptable journey delay,
- $\delta$ : minimum acceptable link persistence,
- $\eta$ : minimum acceptable duration of time over which a network graph should consist of a single component for the network to be considered of class *Conn.*

### 2.4 The centralized classification algorithm

Given the values of the parameters  $\gamma$ ,  $\delta$ ,  $\eta$  the following classification rules are applied at each timeslot  $i$ :

- Any timeslot where  $NCC_i = 1$  is *Conn.*-eligible.
- A maximal succession of *Conn.*-eligible instants going from timeslot  $t_a$  to  $t_b$  constitutes a *Conn.* phase if it meets the following conditions: 1)  $t_a - t_b > \eta$  and 2) the link persistence over this time period is greater than  $\delta$ .
- Any timeslot that is not *Conn.*-eligible or is *Conn.*-eligible but not part of a *Conn.* phase, belongs to either a *Int. Conn.* phase or a *Disconn.* phase.
- The timeslot belongs to a *Int. Conn.* class if it meets either one of the following two conditions: 1) the timeslot  $i$  is more than  $\gamma$  timeslots before the last timeslot that we consider and  $LFJ_i < \gamma$ , or 2) the timeslot  $i$  is less than  $\gamma$  timeslots before the last timeslot and the predecessor timeslot maps onto the *Int. Conn.* class.
- Otherwise the timeslot is part of a *Disconn.* phase.

**Table 1: Notation**

$N$	network area
$K$	transmission range
$M$	number of nodes in the network
$EFJ$	expected foremost journey
$EM_{mm}$	expected meeting time under 'mm'
$E[\tau_{mm}]$	expected contact time under 'mm'
$L$	expected epoch length
$T$	expected epoch duration
$T_{stop}$	average pause time after an epoch
$\bar{v}$	average node speed
$\hat{v}_{mm}$	normalized relative speed under 'mm'

### 3. DISTRIBUTED CLASSIFICATION

#### 3.1 Assumptions and Definitions

In this section we present the distributed classification algorithm which is based on analytically calculated formulas and simple node observations. First, we present the assumptions and some definitions of quantities that are used throughout the algorithm and then we present the algorithm.

Throughout the classification algorithm, it is assumed that the mobile network progresses in timeslots. Also, we assume that the nodes know in advance the transmission range ( $K$ ) which is identical for all the nodes and they also know the size of the network area ( $N$ ). It will become clear later that the algorithm can work with any node mobility model as long as we have analytical formulas for some basic properties of the model. We have derived such formulas in our prior work for a wide range of mobility models such as the Random Direction, Random Waypoint, Random Walk and the Community-based model [12]. In this paper, without loss of generality we assume that the nodes move according to the Random Direction mobility model.

Let's now recall how the Random Direction model works. In particular, it progresses on epochs where at each epoch every node initially chooses a direction  $\theta$  uniformly in  $[0, 2\pi)$ . Then, it chooses a speed uniformly in  $[v_{min}, v_{max}]$  with  $v_{min} > 0$ ,  $v_{max} < \infty$  and average speed  $\bar{v}$ . Also, it chooses a duration  $T$  of movement from an exponential distribution with average  $\frac{T}{\bar{v}}$  and it moves towards  $\theta$  with the chosen speed for  $T$  time units. If the network boundary is reached, it re-enters from the opposite side of the network. After  $T$  time units it pauses for a random amount of time chosen from  $[0, T_{max}]$  with average pause time  $\bar{T}_{stop}$ . Then, a new epoch begins. For convenience, we summarize the notation used in the Random Direction model as well as throughout the paper in Table 1, where 'mm' denotes the corresponding mobility model that we use.

Notice that it is not assumed that the nodes know in advance the number of nodes in the network because as it will be presented, this value can be estimated at each node by simple observations. So, the proposed distributed algorithm can adapt to a network in which the number of nodes changes from time to time. This means that the nodes can run the distributed classification algorithm from time to time and update their decision about the class that the network belongs to, based on the new estimations.

In the following, we define some quantities that are used

in the distributed classification algorithm. The derivation of the analytical formulas that calculate these quantities is presented in the next section with the analytical framework. The following formally define the contact and meeting times of a mobility model 'mm'.

*Definition 1. (Contact Time)* Let nodes  $i$  and  $j$  move according to a mobility model 'mm' and assume they come within range of each other at time 0. The contact time ( $\tau_{mm}$ ) is defined as the time they remain in contact with each other before moving out of the range of each other.

*Definition 2. (Meeting Time)* Let nodes  $i$  and  $j$  move according to a mobility model 'mm' and start from their stationary distribution at time 0. The meeting time ( $M_{mm}$ ) between the two nodes is defined as the time it takes them to first come within range of each other.

#### 3.2 The distributed classification algorithm

The distributed classification algorithm uses a number of thresholds in order to apply the classification rules. These are  $T_{connected}$ ,  $T_{contact}$ ,  $T_{duration}$ ,  $T_{FJ}$  and we assume that they are given as input to the algorithm. We analyze the effect of these parameters in the simulation results section. According to the distributed classification algorithm, each node proceeds as follows:

- Estimates the number of nodes  $M$  in the network with the procedure presented in the following section.
- Calculates the probability that the network is connected  $Pr[\text{network is connected}]$  for the number of nodes that it estimated.

Then, every node at each timeslot  $t_i$  applies the following classification rules:

- If  $Pr[\text{network is connected}] \geq T_{connected}$  and the node has at least one neighbor at this timeslot, then the timeslot  $t_i$  is *Conn.*-eligible.
- If it has decided that the network is *Conn.*-eligible for the last  $T_{duration}$  timeslots and  $E[\tau_{rd}] > T_{contact}$ , then the timeslot  $t_i$  is classified as *Conn.*
- If  $EFJ < T_{FJ}$ , then the timeslot  $t_i$  is classified as *Int. Conn.*
- Otherwise, the timeslot  $t_i$  is classified as *Disconn.*

As we can see from the classification process, the threshold  $T_{duration}$  is related to the minimum duration of a *Conn.* network. This threshold is important because if the network is connected for a small number of consecutive timeslots and then it becomes disconnected, it does not make sense to apply a routing protocol designed for *Conn.* networks. Also, the classification to the *Conn.* class takes into account the average duration of a link, which is actually the expected contact time of the mobility model that we use. This occurs because if the network is connected at each timeslot but the links change too fast, then the use of a routing protocol for *Conn.* networks would fail because the routes between pairs of sources and destinations change also very fast. So, it is better to apply a routing protocol designed for *Int. Conn.* networks which in general is not based on routing tables. Finally, the threshold  $T_{FJ}$  is used in order to distinguish the

*Int. Conn.* from the *Disconn.* networks. So, if the value of the expected foremost journey (*EFJ*) is smaller than  $T_{FJ}$  then, we expect that all the nodes will communicate with each other (over space-time paths) within  $T_{FJ}$  timeslots in the future. Otherwise, the space-time paths are very long, so, it is better to classify the network as *Disconn.*

At this point, it is worth noting the correspondence between the parameters used in the centralized classification algorithm with the ones that are used in the distributed algorithm. Firstly, with the use of the threshold  $T_{connected}$  in the distributed algorithm, we are trying to set a threshold in the probability that the network is connected. Obviously, there is not such a threshold in the centralized approach since the classifier can deterministically decide if the network at the specific timeslot is connected or not. Then, the threshold  $T_{duration}$  is actually the  $\eta$  of the centralized approach since it is the minimum acceptable duration of a *Conn.* network and the same occurs for the  $T_{contact}$  and  $\delta$  which correspond to the minimum acceptable link persistence. As far as the  $T_{FJ}$  is concerned, we have to mention that its value is smaller than the value of  $\gamma$  from the centralized approach. This occurs because in the statement that distinguishes the *Int. Conn.* network class from the *Disconn.* one, the centralized algorithm uses the value of the longest foremost journey, while in the proposed distributed algorithm we use the expected value of the foremost journey. Thus, if only one journey has a big value and all the other journeys have small values, then the value of *EFJ* is not that big as the value of *LFJ*.

#### 4. ANALYTICAL FRAMEWORK

In this section we present the derivation of the formulas that are used in the distributed classification algorithm. Recall that Table 1 summarizes the notation used in the formulas. The results focus on the case that the nodes are moving according to the Random Direction mobility model as we assumed earlier.

The first quantity that we have to compute is the probability that the network is connected given that there are  $M$  nodes in the network and the size of the network area is  $N$ . This value is used in order to estimate how connected the network is and decide if it is possible that the particular timeslot can be classified as *Conn.*-eligible or not. The following Lemma calculates the probability that the network is connected.

LEMMA 1. *The probability that the network is connected at any time, assuming statistical independence between the position of the nodes, is given by*

$$Pr[\text{network is connected}] = (1 - e^{-\rho\pi K^2})^M,$$

where  $\rho$  is the density of the nodes in the network ( $\rho = \frac{M}{N}$ ),  $M$  is the number of nodes in the network,  $N$  is the network area and  $K$  is the transmission range of each node.

PROOF. As in [3], we define the distance of a node to its closest neighbor as its nearest neighbor distance  $\xi$ . Since the Random Direction mobility model in a torus generates a uniform stationary node distribution [2], we have a homogeneous Poisson point process in two dimensions so the probability distribution function (pdf) is given by

$$f(\xi) = 2\pi\rho\xi e^{-\rho\pi\xi^2} \text{ for } \xi > 0.$$

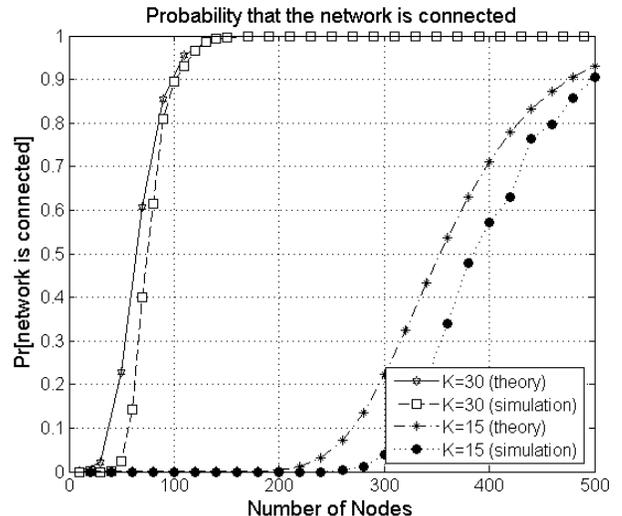


Figure 1: Comparison of the probability that the network is connected (theory and simulation) for network area equals to  $200^2$ , varying number of nodes and varying transmission range ( $K$ )

So, the probability that a specific node has a neighbor is actually the probability that the nearest neighbor distance  $\xi$  of this node is smaller than the transmission range  $K$ . This probability is given by

$$P(\xi \leq K) = \int_{\xi=0}^K f(\xi)d\xi = 1 - e^{-\rho\pi K^2}.$$

If we further assume statistical independence between the position of the nodes, the probability that the network is connected is

$$Pr[\text{network is connected}] = (1 - e^{-\rho\pi K^2})^M. \quad \square$$

Figure 1 presents the analytically calculated values as well as the simulation results of the probability that the network is connected for different values of transmission range and number of nodes. Notice that in the simulations, we used the fraction of time in which the nodes form one single connected component. As we can see, the theoretical value of the fraction of time that the network is connected is a good approximation of the real one. The small discrepancies in the figure result because in reality there is a correlation in the position of the nodes between consecutive timeslots. Thus, if we know that at timeslot  $t_i$  the network is connected then with high probability at the next timeslot  $t_{i+1}$ , the network will be also connected. However, these small discrepancies don't influence that much the classification results.

Let's now compute the expected value of the foremost journey that is used from the classification algorithm to distinguish the *Int. Conn.* from the *Disconn.* networks. From the definition of the foremost journey between two nodes  $a$  and  $b$ , we can conclude that it is actually the expected delay of a packet with source  $a$  and destination  $b$  when we use flooding without contention (infinite buffers and infinite bandwidth) as the routing mechanism in the mobile network. We have already calculated this value in our prior work [11] and the following Lemma presents the calculated formula.

LEMMA 2. *The expected value of the Foremost Journey (EFJ) when the nodes are moving according to a mobility model 'mm' is given by*

$$EFJ = EM_{mm} \frac{H_{M-1}}{M-1},$$

where  $EM_{mm}$  is the expected meeting time of the given mobility model 'mm',  $M$  is the number of nodes in the network and  $H_k$  is the  $k^{\text{th}}$  Harmonic Number, i.e.,  $H_k = \sum_{i=1}^k \frac{1}{i} = \Theta(\log k)$ .

PROOF. See our prior work [11], Theorem 4.2.  $\square$

As we can see from the above formula,  $EFJ$  is related to the expected meeting time of the mobility model in use. Since in our case we assume that the nodes are moving according to the Random Direction mobility model, in the following Lemma we present the analytically calculated expected meeting time for Random Direction.

LEMMA 3. *The expected meeting time under the Random Direction mobility model is given by*

$$EM_{rd} = \frac{(\frac{N}{2KL})(\frac{\bar{L}}{\bar{v}} + \bar{T}_{stop})}{p_m \hat{v}_{rd} + 2(1 - p_m)},$$

where  $\hat{v}_{rd} \approx 1.27$  is the normalized relative speed for the Random Direction model and  $p_m = \frac{T}{T + \bar{T}_{stop}}$  is the probability that a node is moving at any time.

PROOF. See our prior work [12], Theorems 3.1 and 3.2.  $\square$

So, if we combine the results from Lemma 2 and Lemma 3, we can compute the expected value of the foremost journey when the nodes are moving according to the Random Direction mobility model.

Also, in the distributed classification algorithm we need a formula for the expected contact time which is used to distinguish the *Conn.* from the *Int. Conn.* network classes. Lemma 4 presents the formula for the expected contact time when the nodes are moving according to the Random Direction mobility model.

LEMMA 4. *The expected contact time under the Random Direction mobility model is given by*

$$E[\tau_{rd}] = \frac{p_m^2}{p_m^2 + 2p_m(1 - p_m)} E[\tau_{rd}^1] + \frac{2p_m(1 - p_m)}{p_m^2 + 2p_m(1 - p_m)} E[\tau_{rd}^2],$$

where  $E[\tau_{rd}^1]$  is the expected contact time given both nodes were moving when they came within range of each other,  $E[\tau_{rd}^2]$  is the expected contact time given only one of the nodes was moving when they came within range and  $p_m = \frac{T}{T + \bar{T}_{stop}}$  is the probability that a node is moving at any time.

PROOF. See our prior work [9], Theorem 3.4.  $\square$

As it was stated in the description of the distributed classification algorithm, it is not assumed that we know the number of nodes in the network. This makes the classification algorithm more powerful and adaptive to network changes. So, in order for the nodes to get an estimation for the number of nodes that exist in the network, we use the approach presented in our prior work [10] (Section 4.2). For consistency, we also summarize this procedure here.

The main idea of this method is to estimate the number of nodes in the network  $M$  by taking advantage of the inter-meeting time statistics. Let's define  $T_1$  the time until a node (starting from the stationary distribution) encounters any other node. Then, it has been shown that  $T_1$  is exponentially distributed with average  $T_1 = \frac{EM_{mm}}{M-1}$ . Also, if we similarly define  $T_2$  the time until two different nodes are encountered, then  $T_2$  is also exponentially distributed with average  $EM_{mm}(\frac{1}{M-1} + \frac{1}{M-2})$ . So, if we cancel  $EM_{mm}$  from these two equations we get the following estimation for  $M$ :

$$\hat{M} = \frac{2T_2 - 3T_1}{T_2 - 2T_1}.$$

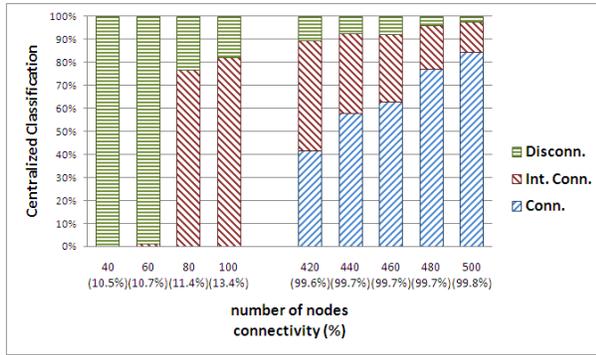
In [10] we also discuss some implications of this procedure and ways for bypassing them.

## 5. SIMULATION RESULTS

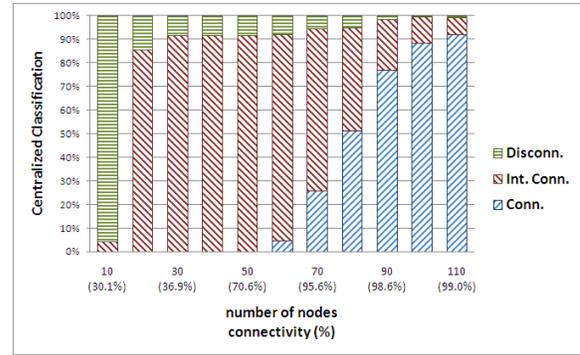
In this section we present the classification results from both the centralized classification algorithm and our distributed version in a number of scenarios. By presenting both results, we are evaluating how accurate the distributed classification algorithm is. Note that since the distributed algorithm is mainly based on analytical formulas, it produces only one single class as output for a given transmission range, number of nodes and size of the network area. On the other hand, the centralized algorithm provides the class percentage for each case. So, if we want to compare the results of the two algorithms, we can focus on the class with the bigger percentage for each case in the centralized algorithm and check the corresponding result from the distributed approach. For convenience, for each of the simulated scenario we provide a table that presents the classification from the distributed algorithm and the corresponding network class that has the majority in every case from the centralized algorithm.

In order to understand how the classification changes, in the simulations we fix two values among the network area, number of nodes and transmission range and we vary the third one. Throughout the simulations, we assume that the nodes are moving according to the Random Direction mobility model where they choose their speed uniformly from (0.5, 1.5) and the time that they pause after each epoch is randomly chosen from (0, 400). Furthermore, we set the following values to the classification parameters:  $T_{connected} = 0.8$ ,  $T_{duration} = 10$ ,  $T_{contact} = 5$  and  $T_{FJ} = 100$ . As we have already shown, the corresponding values for the parameters of the centralized algorithm are  $\eta = 10$  and  $\delta = 5$ . Since there isn't an exact correspondence between the value of  $\gamma$  in the centralized algorithm with the value of the parameter  $T_{FJ}$  in the distributed classification algorithm, we applied a small modification into the centralized algorithm. More specifically, we changed the statement that distinguishes the *Int. Conn.* from the *Disconn.* network class from  $LFJ_i < \gamma$  to  $EFJ_i < \gamma$  in order to use the same value for  $\gamma$  and  $T_{FJ}$ .

Before we proceed, let us define a connectivity metric which indicates how connected the network is. There are a couple of different approaches in the literature [5] [10]. In [5] the authors assume that the network is connected enough such that a large connected component of nodes exists and are interested in the probability that a node belongs to the largest connected component. In the case of disconnected networks, the nodes are isolated and only a few links



(a) Network area =  $200^2$ , transmission range = 15 and varying number of nodes



(b) Network area =  $200^2$ , transmission range = 30 and varying number of nodes

**Figure 2: Centralized classification when the network area and the transmission range are constant and we vary the number of nodes**

**Table 2: Classification when network area =  $200^2$ , transmission range = 15 and varying number of nodes**

Number of nodes	40	60	80	...	420	440	460	480	500
Centralized	<i>Disconn.</i>	<i>Disconn.</i>	<i>Int. Conn.</i>	...	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>
Distributed	<i>Disconn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	...	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>

**Table 3: Classification when network area =  $200^2$ , transmission range = 30 and varying number of nodes**

Number of nodes	10	20	30	...	70	80	90	100	110
Centralized	<i>Disconn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	...	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>
Distributed	<i>Disconn.</i>	<i>Disconn.</i>	<i>Int. Conn.</i>	...	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>

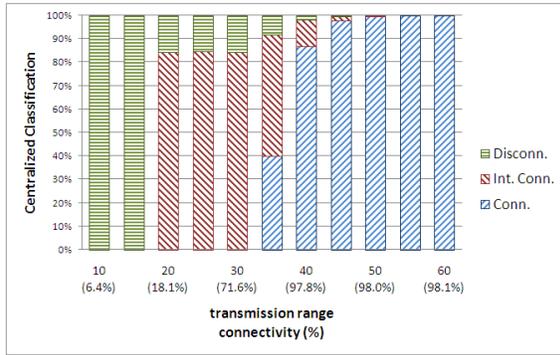
exist between them, so this notion of network connectivity is not very helpful.

With this in mind, in [10] we define the network connectivity to be the expected maximum cluster size which is the percentage of total nodes in the largest connected component. This definition has also some limitations since, for example, it cannot differentiate the case that there are many connected components in the network with the same size from the case that there is only one connected component with this size and the remaining nodes are totally isolated. However, even if such occasions may happen, our definition of the network connectivity gives a good estimate of how connected the network is, irrespectively of the size of the network area, the number of nodes, the transmission range or the node mobility model and can capture both connected and disconnected networks.

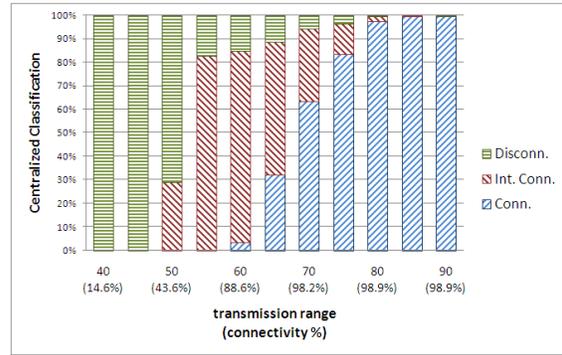
Figure 2 presents the classification from the centralized classification algorithm for a fixed network area equals to  $200^2$  and two cases for the transmission range equal to 15 and 30. In these scenarios we vary the number of nodes and observe the classification results. We also present the network connectivity for the different values of the number of nodes. As we can see from Figure 2(a), the transition from the *Disconn.* class to the *Int. Conn.* happens when there are approximately 70 nodes in the network, while the transition from the *Int. Conn.* to the *Conn.* happens when there are 440 nodes in the network. The corresponding values of the network connectivity as we defined it earlier are 11% and 99.7%.

Let us now focus on the accuracy of the results that we get from the distributed classification algorithm which is the main contribution of this paper. As we can see from Table 2, the distributed classification algorithm is in agreement with the centralized one except from a small difference in the transition from the *Disconn.* to the *Int. Conn.* class. This occurs because the proposed algorithm is mainly based on analytically calculated formulas for the quantities that the centralized algorithm calculates deterministically. Thus, for example, the value of the expected foremost journey is analytically calculated slightly less than  $T_{FJ}$ , so the network is classified as *Int. Conn.*, while in the centralized classification it is calculated deterministically slightly more than  $T_{FJ}$  so the network is classified as *Disconn.*

Furthermore, we get similar results for the scenario in which the network area equals  $200^2$ , the transmission range equals 30 and we vary the number of nodes (Figure 2(b)). In this case the network is classified as *Disconn.* till there are 10 nodes which corresponds to a network connectivity of 30%. Also, for number of nodes from 20 to 70, it is classified as *Int. Conn.* while for number of nodes more than 80, the network is classified as *Conn.*. The network connectivity when there are 80 nodes in the network is approximately 98.6% which is quite high. For this scenario, we present the accuracy of the distributed algorithm in Table 3. As we can see, there are some differences only in the boundaries of the network classes for the reasons that we described earlier. Other than that, our proposed distributed classification algorithm works like the centralized one, meaning that it de-



(a) Network area =  $200^2$ , number of nodes = 50 and varying transmission range



(b) Network area =  $500^2$ , number of nodes = 100 and varying transmission range

**Figure 3: Centralized classification when the network area and the number of nodes are constant and we vary the transmission range**

**Table 4: Classification when network area =  $200^2$ , number of nodes = 50 and varying transmission range**

Transmission range	10	15	20	25	30	35	40	45	50
Centralized	<i>Disconn.</i>	<i>Disconn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>
Distributed	<i>Disconn.</i>	<i>Disconn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>

**Table 5: Classification when network area =  $500^2$ , number of nodes = 100 and varying transmission range**

Transmission range	40	45	50	55	60	65	70	75	80
Centralized	<i>Disconn.</i>	<i>Disconn.</i>	<i>Disconn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>
Distributed	<i>Disconn.</i>	<i>Disconn.</i>	<i>Disconn.</i>	<i>Disconn.</i>	<i>Int. Conn.</i>	<i>Int. Conn.</i>	<i>Conn.</i>	<i>Conn.</i>	<i>Conn.</i>

cides the same class of the network for each case and also, it can be implemented in practice since it doesn't assume a centralized authority in order to make a decision.

We also present some other scenarios in which we vary the transmission range (Figure 3) and the network area (Figure 4). The corresponding classification results are shown in Tables 4 to 7. In all the cases, the classification results from the two algorithms are pretty much the same.

As we can see from the figures, there is a correlation between the connectivity and the class that the network belongs to. Even if the value of the network connectivity is not identical in the boundaries of the classes for the different presented scenarios, we can extract useful knowledge from this experimentation. For example, a network connectivity between 40% and 70% implies that the network is not *Disconn.* or *Conn.* To this end, the network connectivity can give us a rough estimate of the network class without even knowing the details of the network like the node mobility model or the transmission range of the nodes.

## 6. CONCLUSIONS AND FUTURE WORK

In this paper, we have considered the problem of how to determine the class that a mobile network belongs to, in a distributed fashion. Our approach is based on a combination of measurements and analytical techniques. At every timeslot, each node has an estimate of the class that the network belong to. Based on this estimate, it can then select a corresponding routing protocol to send its potential packets to the destination inside the mobile network. We checked

the accuracy of our distributed classification algorithm with a centralized one that has been proposed in the literature and we concluded that the results are quite accurate.

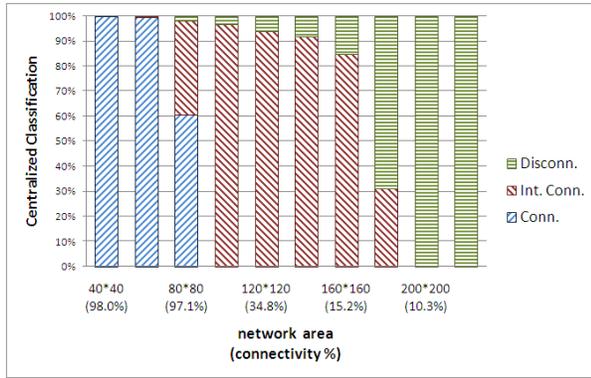
In the future, we plan to apply the distributed classification scheme to a different number of mobility models such as the Community-based model [12] or group mobility models. In such cases, depending on the positions of the nodes, their estimations about the class of the mobile network can be totally different. Also, when a group mobility model is applied, one node could have different class estimations for different groups of nodes and not for the whole network. Thus, for example it can decide that with some nodes it is continuously connected, so it can use a *Conn.* routing protocol to send packets to them, while it can decide that for some other nodes its connectivity is like the one in *Disconn.* networks. Another point to consider as future work could be the selection of the most appropriate routing protocol within each class of mobile networks. Also, in the case of the *Int. Conn.* networks, the classification algorithm could choose the optimal number of copies that would be disseminated to the network in order to minimize the transmission delay.

## 7. ACKNOWLEDGMENTS

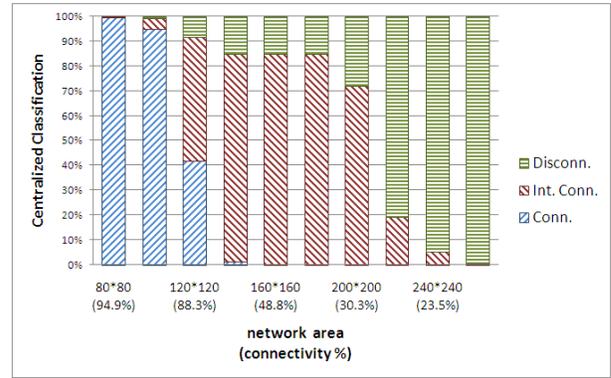
Parts of this work have been supported by a USC Provost's Fellowship and a METRANS Transportation Center Grant.

## 8. REFERENCES

- [1] E. M. Belding-Royer, C. E. Perkins, and S. R. Das.



(a) Number of nodes = 50, transmission range = 15 and varying network area



(b) Number of nodes = 20, transmission range = 30 and varying network area

**Figure 4: Centralized classification when the number of nodes and the transmission range are constant and we vary the network area**

**Table 6: Classification when number of nodes = 50, transmission range = 15 and varying network area**

Network area	$40^2$	$60^2$	$80^2$	$100^2$	...	$160^2$	$180^2$	$200^2$	$220^2$
Centralized	Conn.	Conn.	Conn.	Int. Conn.	...	Int. Conn.	Disconn.	Disconn.	Disconn.
Distributed	Conn.	Conn.	Conn.	Int. Conn.	...	Int. Conn.	Int. Conn.	Disconn.	Disconn.

**Table 7: Classification when number of nodes = 20, transmission range = 30 and varying network area**

Network area	$80^2$	$100^2$	$120^2$	$140^2$	$160^2$	$180^2$	$200^2$	$220^2$	$240^2$
Centralized	Conn.	Conn.	Int. Conn.	Int. Conn.	Int. Conn.	Int. Conn.	Int. Conn.	Disconn.	Disconn.
Distributed	Conn.	Conn.	Int. Conn.	Int. Conn.	Int. Conn.	Int. Conn.	Disconn.	Disconn.	Disconn.

Ad-hoc on-demand distance vector routing. *IETF MANET DRAFT*, 2002.

- [2] C. Bettstetter. Mobility modeling in wireless networks: categorization, smooth movement, and border effects. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(3):55–66, 2001.
- [3] C. Bettstetter. On the minimum node degree and connectivity of a wireless multihop network. In *MobiHoc '02*, pages 80–91, New York, NY, USA, 2002. ACM.
- [4] V. Borrel, M. H. Ammar, and E. W. Zegura. Understanding the wireless and mobile network space: a routing-centered classification. In *CHANTS '07*, pages 11–18, New York, NY, USA, 2007. ACM.
- [5] O. Dousse and P. Thiran. Connectivity vs capacity in dense ad hoc networks. In *Proc. IEEE Infocom, Hong Kong*, 2004.
- [6] D. B. Johnson, D. A. Maltz, and J. Broch. *DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks*, pages 139–172. Addison-Wesley, 2001.
- [7] Q. Li and D. Rus. Communication in disconnected ad hoc networks using message relay. *Parallel Distrib. Comput.*, 63(1):75–86, 2003.
- [8] R. Ramanathan, P. Basu, and R. Krishnan. Towards a formalism for routing in challenged networks. In *CHANTS '07*, pages 3–10, New York, NY, USA, 2007. ACM.
- [9] T. Spyropoulos, A. Jindal, and K. Psounis. An analytical study of fundamental mobility properties for encounter-based protocols. *Int. J. Auton. Adapt. Commun. Syst.*, 1(1):4–40, 2008.
- [10] T. Spyropoulos, K. Psounis, and C. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *Proceedings of ACM SIGCOMM workshop on Delay Tolerant Networking (WDTN-05)*, pages 252–259, 2005.
- [11] T. Spyropoulos, K. Psounis, and C. Raghavendra. Efficient routing in intermittently connected mobile networks: The single-copy case. *IEEE/ACM Transactions on Networking*, 16(1):63–76, 2008.
- [12] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Performance analysis of mobility-assisted routing. In *MobiHoc '06*, pages 49–60, New York, NY, USA, 2006. ACM.
- [13] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical report, Duke University, CS-200006, 2000.
- [14] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *MobiHoc '04*, pages 187–198, New York, NY, USA, 2004. ACM.