Mobile Networks and Applications 10, 61–77, 2005

Regional Gossip Routing for Wireless Ad Hoc Networks

XIANG-YANG LI, KOUSHA MOAVENINEJAD and OPHIR FRIEDER
Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616, USA

Abstract. Many routing protocols have been proposed for wireless ad hoc networks, and most of them are based on some variants of flooding. Thus many routing messages are propagated through the network unnecessarily despite various optimizations. Gossip based routing method has been used and re-investigated to reduce the number of messages in both wired networks and wireless ad hoc networks. However, the global gossiping still generates many unnecessary messages in the area that could be far away from the line between sender node and receiver node. We propose a regional gossip approach, where only the nodes within some region forward a message with some probability, to reduce the overhead of the route discovery in the network. We show how to set the forwarding probability based on the region and the network density both by theoretical analysis and by extensive simulations. Our simulations show that the number of messages generated using this approach is much less than the simple global gossiping method, which already saves many messages compared with global flooding. We expect that the improvement should be even more significant in larger networks.

Keywords: gossip, fault tolerance, routing, wireless ad hoc networks

1. Introduction

Recent years saw a great amount of research in wireless networks, especially ad hoc wireless networks due to its potential applications in various situations such as battlefield, emergency relief, and so on. There are no wired infrastructures or cellular networks in *ad hoc* wireless network. Two nodes can communicate directly if they are within the transmission range of the other. Otherwise, they communicate through multi-hop wireless links by using intermediate nodes to relay the message. Consequently, each node in the wireless network also acts as a router, forwarding data packets for other nodes. In addition, we assume that each node has a low-power Global Position System (GPS) receiver, which provides the position information of the node itself. If GPS is not available, the distance between neighboring nodes can be estimated on the basis of incoming signal strengths and the direction of arrival. Relative co-ordinates of neighboring nodes can be obtained by exchanging such information between neighbors [1].

The devices in the wireless ad hoc networks are often powered by batteries only. Thus, the power supply is limited and it is often difficult to recharge the batteries, which motivates many researches in designing power efficient protocols for power assignment [2–7], topology control [8–14] and routing [15–17]. In addition, the bandwidth available is much less compared with the wired networks counterpart due to its unique transmission characteristics. Moreover, since nodes can be mobile, routes may constantly change. Thus, the designed routing protocols for wireless ad hoc networks should use as less messages as possible, which will reduce power consumption (thus enlong network life), and signal interference (thus increase the throughput).

One of the key challenges in the design of *ad hoc* networks is the development of dynamic routing protocols that can efficiently find routes between two communication nodes. In recent years, a variety of routing protocols [16,18–32], targeted specifically for *ad hoc* environment, have been developed. For the review of the state of the art of routing protocols, see surveys by Royer and Toh [33], by Ramanathan and Steenstrup [34], and by Mauve et al. [35]. Some routing protocols assume that the each node knows its own positions (e.g., equipped with GPS receivers). These category of protocols are called *Location-Aided Routing* (LAR) protocols in which the overhead of route discovery is decreased by utilizing location information. Some protocols do not rely on position information, and make use flooding (or some variants of flooding). Thus many routing messages are propagated through the network unnecessarily despite possible various optimizations. Gossip based routing method has been used and re-investigated to reduce the number of messages in both wired networks and wireless ad hoc networks. Whenever a node receives a message, it tosses a coin to decide whether to forward a message or not in order to reduce the total number of routing messages sent by all nodes. However, the global gossiping still generates many unnecessary messages in the area that could be far away from the line between sender node and receiver node. We propose a regional gossip approach, where only the nodes within some region forward a message with some probability, to reduce the overhead of route discovery in the network.

The key observation for all gossiping based routing methods is that the gossiping exhibits a bimodal behavior, which is well-known in the percolation theory [36,37]. This can be rephrased as follows. Let p be the uniform probability that a node will forward the routing message to its neighbors. Then, there is a threshold value p_0 such that, in sufficiently large random networks, the gossip message quickly dies out if $p < p_0$ (p is slightly less than p_0) and the gossip message spreads to all network nodes if $p > p_0$ (p is slightly greater than p_0). In other words, in almost all executions, either almost no node receives the message or almost all of them do. So ideally, we would set the gossiping probability to some value slightly larger than p_0 to reduce the routing messages propagated. When the network is sufficiently large, we can set p sufficiently close to p_0, thus save about $(1 - p_0)n$ messages overhead compared with the flooding, since about $p_0 n$ nodes will forward the message in gossiping based method compared with n nodes forwarding in flooding. Hass et al. [24] conducted extensive simulations to investigate the extent to which this gossiping probability can be lowered. They found that using gossiping probability between 0.6 and 0.8 suffices to ensure that almost every node gets the message in almost every routing. They report of up to 35% fewer messages than flooding (close to our previous explanation). Notice that their experimental setting of the network has some special configurations [24].

Although gossiping reduces the routing messages compared with flooding, it still produces lots of unnecessary messages in regions that are far from the line between sender node and receiver node. Notice that, the traditional gossip will propagate the message to the whole network. To further reduce the number of forwarding messages, we propose regional gossiping, in which essentially only nodes inside some region (derived from the source and target) will execute the gossiping protocol, and nodes outside the region will not participate in the gossiping at all. The region we select in our simulations are some ellipses using the source and target as foci. Notice that here we assume source node knows either the exact or the approximate location of the destination node, we will discuss this later in section 2 in detail. We also dynamically adjust the forwarding probability based on the node density estimated by the current node. Our results show that, by using appropriate optimization heuristics, we can save up to 94% messages even compared with the global flooding method.

The remaining of this paper is organized as follows. In section 2, we review some known location services techniques for wireless ad hoc networks. We study our regional gossip method in detail in section 3. We demonstrate its effectiveness by both theoretical study and extensive simulations in section 4 . We also study the effectiveness of the regional gossiping on constructing multiple paths for any pair of source and destination nodes in section 5. We conclude our paper and discuss possible future research directions in section 6.

2. Preliminaries

We consider a wireless ad hoc network (or sensor network) with all nodes distributed in a two-dimensional plane. Assume that all wireless nodes have distinctive identities and each static wireless node knows its position information[1] either through a low-power Global Position System (GPS) receiver or through some other way. For simplicity, we also assume that all wireless nodes have the same maximum transmission range and we normalize it to one unit. Throughout this paper, a *broadcast* by a node u means that u sends the message to all nodes within its transmission range. Notice that, in wireless ad hoc networks, the radio signal sent out by a node u can be received by all nodes within the transmission range of u. The main communication cost in wireless networks is to send out the signal while the receiving and processing costs of a message is neglected here.

[1] More specifically, it is enough for our protocol when each node knows the relative position of its one-hop neighbors. The relative position of neighbors can be estimated by the *direction of arrival* and the *strength of signal*.

2.1. Location service

Several proposed routing algorithms [18,22] assume that the source node knows the position information (or approximate position) of the destination node. Our regional gossip method also assumes that the source node knows the current position information of the target approximately. Notice that, for sensor networks collecting data, the destination node is often fixed, thus, location service is not needed in those applications. However, the help of a *location service* is needed in most application scenarios. Mobile nodes register their locations to the location service. When a source node does not know the position of the destination node, it queries the location service to get that information. In cellular networks, there are dedicated position severs. It will be difficult to implement the centralized approach of location services in wireless ad-hoc networks. First, for centralized approach, each node has to know the position of the node that provides the location services, which is a chicken-and-egg problem. Second, the dynamic nature of the wireless ad hoc networks makes it very unlikely that there is at least one location server available for each node. Thus, we will concentrate on distributed location services.

For the wireless ad hoc networks, the location service provided can be classified into four categorizes: *some-for-all*, *some-for-some*, *all-for-some*, *all-for-all*. Some-for-all service means that some wireless nodes provide location services for all wireless nodes. Other categorizations are defined similarly.

An example of all-for-all services is the location services provided in the Distance Routing Effect Algorithm for Mobility (DREAM) by Basagni et al. [38]. Each node stores a database of the position information for all other nodes in the wireless networks. Each node will regularly flood packets containing its position to all other nodes. A frequency of the flooding and the range of the flooding is used as a control of the cost of updating and the accuracy of the database.

Using the idea of *quorum* developed in the databases and distributed systems, Hass and Liang [39] and Stojmenovic [40] developed quorum based location services for wireless ad-hoc networks. Given a set of wireless nodes V, a quorum system is a set of subset $(Q_1, Q_2, \ldots, Q_k)$ of nodes whose union is V. These subsets could be mutually disjoint or often have equal number of intersections. When one of the nodes requires the information of the other, it suffices to query one node (called the representative node of Q_i) from each quorum Q_i. A virtual backbone is often constructed between the representative nodes using a non-position-based methods such as [41–44]. The updated information of a node v is sent to the representative node (or the nearest if there are many) of the quorum containing v. The difficulty of using quorum is that the mobility of the nodes requires the frequent updating of the quorums. The quorum based location service is often *some-for-some* type.

The other promising location service is based on the quadtree partition of the two-dimensional space [45]. It divides the region containing the wireless network into hierarchy of squares. The partition of the space in [45] is uniform. However, we notice that the partition could be non-uniform if the density of the wireless nodes is not uniform for some applications. Each node v will have the position information of all nodes within the same *smallest* square containing v. This position information of v is also propagated to up-layer squares by storing it in the node with the nearest identity to v in each up-layer square containing v. Using the nearest identity over the smallest identity, we can avoid the overload of some nodes. The query is conducted accordingly. It is easy to show that it takes about $O(\log n)$ time to update the location of v and to query another node's position information.

If the location service is not provided, the nodes can cache the location information of some other nodes. When the source node wants to send a message to the target, it directly uses the region gossip if the target location is known. Otherwise, it will use flooding (with selective forwarding [46] to control the number of messages sent) to send the message to all nodes within k hops, where k is a parameter to be set. Then if a node within k hops knows the destination location, that node then starts the regional gossip to send message to the destination.

2.2. Random deployment and connectivity

Energy conservation is critical for the life of the wireless network. One approach to save energy is to use the minimum power to transmit the signal without disconnecting the network. The universal minimum power used by all wireless nodes, such that the induced network topology is connected, is called the *critical power*. Determining the critical power for static wireless ad hoc networks is well-studied [5,7,13]. It remains to study the critical power for connectivity for mobile wireless networks. As the wireless nodes move around, it is impossible to have a unanimous critical power to guarantee the connectivity for all instances of the network configuration. Thus, we need to find a critical power, if possible, at which each node has to transmit to guarantee the connectivity of the network almost surely, i.e., with high probability almost one.

The wireless nodes are randomly deployed in majority wireless ad hoc networks either due to its massive number, due to its emergency requirement, or due to harsh environment. For simplicity, we assume that the n wireless devices are distributed in a unit area square (or disk) according to some distribution function, e.g., random uniform distribution, denoted by $\mathcal{X}_n$, or Poisson process, denoted by $\mathcal{P}_n$.

Let $G(V, r)$ be the graph defined on V with edges $uv \in E$ if and only if $\|uv\| \leqslant r$ where $\|uv\|$ is the Euclidean distance between nodes u and v. Let $\mathcal{G}_\Omega(\mathcal{X}_n, r_n)$ be the set of graphs $G(V, r_n)$ for n nodes V that are uniformly and independently distributed in a two-dimensional region Ω. The problem considered by Gupta and Kumar [5] is then to determine the value of r_n such that a random graph in $\mathcal{G}_\Omega(\mathcal{X}_n, r_n)$ is asymptotically connected with probability one as n goes to infinity, when Ω is a unit disk. Specifically, they showed that $G(V, r_n)$ is connected almost surely if $n\pi r_n^2 \geqslant \ln n + c(n)$ for any $c(n)$ with $c(n) \to \infty$ as n goes to infinity, and $\mathcal{G}(\mathcal{X}_n, r_n)$ is asymptotically disconnected with positive probability if $n\pi r_n^2 = \ln n + c(n)$ and $\limsup_n c(n) < +\infty$. It is unknown whether the same result holds if the geometry domain in which the wireless nodes are distributed is a unit-area square instead of the unit-area disk.

Independently, Penrose [47] showed that the longest edge M_n of the minimum spanning tree of n points randomly and uniformly distributed in a unit area square $\mathcal{C}$ satisfies that

$$\lim_{n\to\infty} \Pr\big(n\pi M_n^2 - \ln n \leqslant \alpha\big) = \mathrm{e}^{-\mathrm{e}^{-\alpha}},$$

for any real number α. This result gives the probability of the network to be connected if the transmission radius is set as a positive real number r when n goes to infinity. For example, if we set $\alpha = \ln\ln n$, we have

$$\Pr\big(n\pi M_n^2 \leqslant \ln n + \ln\ln n\big) = \mathrm{e}^{-1/\ln n}.$$

It implies that the network is connected with probability at least $\mathrm{e}^{-1/\ln n}$ if the transmission radius r_n satisfies $n\pi r_n^2 = \ln n + \ln\ln n$. Notice that $\mathrm{e}^{-1/\ln n} > 1 - 1/\ln n$ from $\mathrm{e}^{-x} > 1 - x$ for $x > 0$. By setting $\alpha = \ln n$, the probability that the graph $G(V, r_n)$ is connected is at least $\mathrm{e}^{-1/n} > 1 - 1/n$, where $n\pi r_n^2 = 2\ln n$. Notice that the above probability is only true when n goes to infinity. When n is a finite number, then the probability of the graph being connected is smaller. In [48], Li et al. presented the experimental study of the probability of the graph $G(V, r_n)$ being connected for finite number n.

Gupta and Kumar [5] conjectured that if every node has probability p of being fault, then the transmission range for resulting a connected graph satisfies $p\pi r_n^2 = \log n/n$. This was recently confirmed by Wan et al. [49]. It is not difficult to see that whether the global gossip can deliver the packet is related to whether a set of randomly deployed nodes in a region form a connected graph when each node has a uniform faulting probability p. Consequently, given a wireless network with n nodes , each with transmission range r, the relay probability of a gossip routing protocol is $p = \log n/(\pi n r_n^2)$, when n goes to infinity. We conjecture that this is true for any non-flat convex region Ω.

2.3. Fault tolerance and security

Fault tolerance is one of the central challenges in designing the wireless ad hoc networks. To make fault tolerance possible, first of all, the underlying network topology must have multiple disjoint paths to connect any two given wireless devices. Here the path could be vertex disjoint or edge disjoint. Considering the communication nature of the wireless networks, the vertex disjoint multiple paths are often used in the literature. A graph is called k-vertex connected (k-connected for simplicity) if, for each pair of vertices, there are k mutually vertex disjoint paths (except end-vertices) connecting them. A k-connected wireless network can sustain the failure of $k-1$ nodes.

The connectivity of random graphs, especially the geometric graphs and its variations, have been considered in the random graph theory literature [50], in the stochastic geometry literature [47,51–54], and the wireless ad hoc network literature [2,5,55–61].

Penrose [53] showed that a graph of $G(\mathcal{X}_n, r)$ becomes k-connected almost surely at the moment it has minimum degree k. However, this does not mean to guarantee a graph over n points is k-connected almost surely, we only have to connect every node to its k nearest neighbors. Let V be a set of n points randomly and uniformly distributed in a unit square (or disk). Xue and Kumar [61] proved that, to guarantee that a geometry graph over V is connected, the number of nearest neighbors that every node has to connect must be asymptotically $\Theta(\ln n)$. Dette and Henze [51] studied the maximum length of the graph by connecting every node to its k nearest neighbors asymptotically. For the unit volume sphere, their result implies that, when $k > 2$,

$$\lim_{n\to\infty} \Pr\bigl(n\pi r_{n,k}^2 \leqslant \ln n + (2k-3)\ln\ln n - 2\ln(k-1)! \\ - 2(k-2)\ln 2 + \ln\pi + 2\alpha\bigr) = \mathrm{e}^{-\mathrm{e}^{-\alpha}}.$$

Li et al. [48] showed that, given n random points V over a unit-area square, to guarantee that a geometry graph over V is $(k+1)$-connected, the number of nearest neighbors that every node has to connect is asymptotically $\Theta(\ln n + (2k-1)\ln\ln n)$. Li et al. [48] derived a tighter bound on r_n for a set V of n two-dimensional points randomly and uniformly distributed in $\mathcal{C}$ such that the graph $G(V, r_n)$ is k-connected with high probability.

The theoretical value of the transmission ranges gives us insight on how to set the transmission radius to achieve the k-connectivity with certain probability. These results also apply to mobile networks when the moving of wireless nodes always generate randomly (or Poisson process) distributed node positions. Bettstetter [2] conducted the experiments to study the relations of the k-connectivity and the minimum node degree using toroidal model. Li et al. [48] also conducted experiments to study the probability that a graph has minimum degree k and has vertex connectivity k simultaneously using Euclidean model. Recently, Bahramgiri et al. [8] showed how to decide the minimum transmission range of each node such that the resulted directed communication graph is k-connected. Here it assumes that the unit disk graph by setting each node with the maximum transmission range is k-connected. Lukovszki [62] gave a method to construct a spanner that can sustain k nodes or k links failures.

3. Regional gossip

Although gossiping reduces the routing messages compared with flooding, it still produces lots of unnecessary messages in regions that are far away from the line between the source and the target node. Notice that, the traditional gossip will propagate the message to the whole network. To further reduce the number of forwarding messages, we propose regional gossiping, in which essentially only nodes inside some region (derived from the source and target) will execute the gossiping protocol, and nodes outside the region will not participate the gossiping at all. The region we select in our simulations are some ellipses using the source and target as foci.

We now describe our regional gossiping routing method in detail. Assume that wireless mobile hosts are a set V of n points distributed in a two-dimensional space. Each node has a fixed transmission range r: all nodes within distance r to a node v can receive the signal sent by v. Thus, all mobile hosts define a communication graph $G(V, r)$ in which there is an edge uv iff $\|uv\| \leqslant r$. From now on, we also assume that the source node knows the position of the target node, the global ellipse factor ℓ, in addition to its own position. Every mobile host can get its own position through a low-cost GPS. In many applications such as data-centric sensor network, there is only a fixed number of destination nodes (called sink), which is often static, thus every node knows the positions of these possible target nodes. Otherwise, location service is needed to find the location of the destination node. The geometry information of the source node and the destination node and also the current route (i.e., the route from source to the sender of the message) is piggybacked along with the message packet. When a node, say v, receives a message, it retrieves the geometry position of the source node and the target node. Node v then checks if it is inside the ellipse defined by using the source point s and the destination point t as foci. Notice that, a node v is inside this ellipse iff

$$\|vs\| + \|vt\| \leqslant \ell\|st\|,$$

which can be checked trivially. When a node is not inside the ellipse, it will just simply discard this message. Otherwise, with a fixed probability p, the node forwards this message to all nodes within its transmission range. Hereafter, we call p the *relay probability* and ℓ the *ellipse factor* of our regional gossiping method. Obviously, the probability that the destination node receives the message depends on the relay probability p, the ellipse factor ℓ, the number of nodes n, and the transmission range r.

Gupta and Kumar [5] showed that a random graph $G(V, r)$ is connected whenever r is larger than some threshold value r_n. It is known that the global gossiping (by simply setting ℓ to ∞) exhibits some bimodal behavior: the destination node receives the message if and only if the relay probability is larger than some threshold value. We expect our regional gossiping method to have the similar transmission phenomena.

We then estimate the relay probability for a network of n nodes. It was shown in [49] that given n wireless nodes distributed in a unit square and each node has transmission range r_n and being off or fault with probability p, then the network is connected with high probability if $pn\pi r_n^2 \simeq 2\ln n$. Consider the network of n nodes distributed in a square region with side length a. Assume that the distance between the source and the target is d and the ellipse factor is ℓ. The number of nodes inside the ellipse is then about

$$N_d = \frac{n}{a^2} \cdot \frac{\pi \ell \sqrt{\ell^2 - 1}}{4} d^2.$$

Since each node inside the ellipse forwards the message with probability p after it receives the message, to let the target receive the message almost surely, the subnetwork composed of the nodes inside the ellipse with fault probability p must be connected. In other words, the relay probability in our regional gossiping is at least

$$p \geqslant \frac{\ln N_d + c(N_d)}{N_d \pi (r/a)^2}.$$

Here r is the transmission range of each wireless node and $c(N_d)$ is a number going to ∞ when N_d goes to ∞. The probability that the network (each node is chosen with probability p) is connected is $\mathrm{e}^{-\mathrm{e}^{-c(N_d)}}$. Substituting in N_d, we have

$$p \geqslant \frac{4a^4 \ln(n\pi\ell\sqrt{\ell^2-1}d^2/(4a^2))}{\pi^2 d^2 r^2 \ell\sqrt{\ell^2-1} \cdot n} = \frac{\ln(n\pi\tilde{\ell}^2\tilde{d}^2/4)}{n\pi^2\tilde{\ell}^2\tilde{d}^2\tilde{r}^2/4}.$$

Here $\tilde{\ell}^2 = \ell\sqrt{\ell^2-1}$, $\tilde{d} = d/a$, and $\tilde{r} = r/a$. Since for a random pair of source and target nodes, $d \leqslant \sqrt{2}a$, we have

$$p \simeq \frac{\ln(n\pi\tilde{\ell}^2/4)}{n\pi^2\tilde{\ell}^2\tilde{r}^2/4}.$$

For example, consider a network of $n = 1000$ nodes distributed in a square of side length $a = 15$, and each node has transmission range $r = 1$. For ellipse factor $\ell = 1.2$, we can calculate the relay probability p such that the regional gossiping routing can deliver the packets almost surely as

$$p \simeq \frac{\ln(n\pi\tilde{\ell}^2/4)}{n\pi^2\tilde{\ell}^2\tilde{r}^2/4} = 0.74.$$

The actual relay probability should be larger since we omit the number $c(N_d)$ here, which actually decides the success probability of the regional gossiping. The percentage of all vertices involved is at most

$$p \cdot N_d/n = \frac{\ln(n\pi\tilde{\ell}^2\tilde{d}^2/4)}{\pi\tilde{r}^2 \cdot n} \simeq 0.46.$$

Since the distance d between most pairs of source and target is small compared with a, the actual number of involved vertices is much smaller. Let P_d be the probability that a pair of source and target has distance d. The average percentage of number of vertices (for all source and target pairs) is actually $\int_{x=0}^{a} p \cdot N_x P_x/n \,\mathrm{d}x$. It is not difficult to show that the percentage of vertices involved in regional gossiping is at most $pN_d/2n = 0.23$. When the ellipse factor $\ell = \infty$, we can estimate the relay probability of the regional gossiping as

$$p \simeq \frac{\ln n}{n\pi\tilde{r}^2} = 0.495.$$

The actual relay probability should be larger, so do the percentage of vertices involved in global gossiping. The experiments discussed in the following sections verify the above study.

4. Experimental studies

4.1. Simulation environment

We conducted extensive simulations to study the performance of our region gossiping method. We model the network by unit disk graph and the mobile hosts are randomly placed in a square region. We tried unit disk graphs with different number of vertices that are randomly placed in a 15×15 square. Notice that the density of the graph must be above some threshold to see the effectiveness of the algorithm otherwise the properties would be hidden and cannot be seen. In other words, the algorithm works better for dense graphs than sparse graphs with the same parameters p and ℓ.

There are different parameters involved in our simulations, which are described as follows:

Number of vertices. We tried graphs with 1000, 1500 and 2000 vertices. For convenience, we use n to denote the number of vertices.

Ellipse factor. In each iteration of the simulation, the source vertex and the target vertex are the foci of an ellipse with ellipse factor ℓ chosen from 1.2, 1.4, 1.6, 1.8 and 2. We also consider the case where the ellipse factor ℓ is ∞ which is just the traditional global gossiping method. The smaller the ellipse factor is, the narrower the ellipse will be. Notice that ellipse factor must be greater than one.

Transmission range. Remember that to make the graph $G(V, r)$ connected, the transmission range has to be greater than some threshold value r_n. To study the effect of the graph density on the delivery rate, we tried different values of transmission range: 1, 1.5, 2, 2.5 and 3. From the result by Gupta and Kumar [5], given 1000 nodes in a 15×15 square, the transmission range should be at least about 0.7 to guarantee a connected network $G(V, r)$ theoretically.

Relay probability. In our simulation, we use different relay probabilities p. First, we use the relay probabilities p from 0.1 to 1.0 with step 0.1 and we find that, when the network is dense enough, the transmission phenomenon happens between two intervals of relay probabilities. To study this transmission phenomenon in detail, we further refine our relay probabilities. Specifically, we conduct further simulations using relay probabilities from 0.02 to 0.30 with step 0.02.

Beside the above parameters there are two more constant metrics used in our simulations as follows:

Source-target pairs. To compute the exact value of the average delivery rate, we have to try all possible pairs for each graph, which is $n \cdot (n-1)$, where n is the number of vertices. It is not feasible to test all pairs when n is large. Instead we randomly select 100 pairs for each graph and conduct regional gossiping based routing for each pair. Although we are not testing all possible pairs, choosing 100 random pairs would give the results close enough to exact values.

Number of try's. The *delivery probability* (called *delivery rate* also) of our regional gossiping method for a pair of nodes is defined as the probability that the destination node receives the message. To compute the delivery rate, we tried sending the message 1000 times for each pair and then the delivery rate is approximated by the total number of times that the message reached the target divided by the total number that the message is sent (which is 1000 in out simulations).

There are four different types of nodes in each iteration of our simulations:

(1) *Not in ellipse*. Nodes that are out of the ellipse region.

(2) *Blocked*. Nodes that receive the message and do not relay it.

(3) *Relayed*. Nodes that receive and relay the message.

(4) *Initial hops nodes*. The nodes within the initial hops always receive the message and from those, the ones whose distance from source is less than some fix initial hops parameter, always relay the message. Other nodes inside the ellipse relay the message with the given relay probability.

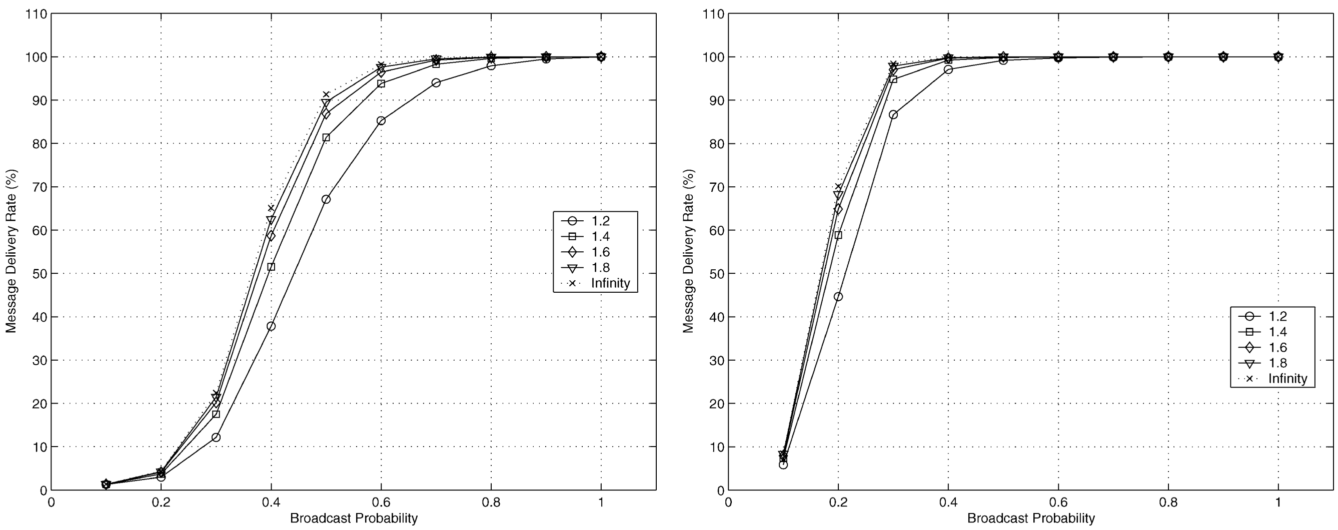
Here we want to involve as little nodes as possible. In other words, we want to minimize the number of nodes that relay the message. It is important because sending message consumes energy and energy is a bottleneck for wireless nodes.

In all the figures of this paper the $\mathbb{Y}$-axis is either the message delivery rate or the percentage of vertices that are involved in message delivery, and the $\mathbb{X}$-axis is one of the parameters with respect to another parameter which is shown in the legend and the remaining two parameters are fixed. For example, we can show message delivery rate as a function of relay probability p for different values of ellipse factor ℓ, while the transmission range r and the number of vertices n are fixed (see figure 1). Each point in each figure represents the *average* of the $100,000$ iterations since we will test 100 different source-target pairs, and each pair is tested 1000 times, when all four parameters are fixed.

We believe that the relay probability and the graph density are two major factors of message delivery rate. On the other hand, the ellipse factor and the relay probability are the major factors determining the number of vertices that are involved in message delivery. Here a node is said to be *involved* if it relays the message. In other words, when the $\mathbb{Y}$-axis is the message delivery rate and $\mathbb{X}$-axis is either relay probability, number of vertices or transmission range, we expect to see a *jump* in the figures. It means that when the $\mathbb{X}$-axis exceeds some threshold, then the regional gossiping method almost surely guarantees that the message arrives at the target. When the $\mathbb{X}$-axis is less than some threshold, the target almost never gets the message.

4.2. Message delivery rate as a function of relay probability

We first conduct extensive simulations to study the effect of the relay probability on the message delivery rate. Intuitively, if we increase the relay probability, the message delivery rate increases. Besides the relay probability, we vary either the ellipse factor ℓ, or the number of vertices n, or the transmission range r. Now we discuss them one by one as follows.



(a) (b)

Figure 1. Message delivery rate as a function of relay probability for different values of ellipse factor. Here transmission range is 1. (a) Number of vertices is 1000. (b) Number of vertices is 2000.

1. Message delivery rate as a function of relay probability for different values of ellipse factor. As can be seen in figure 1, when the probability exceeds some threshold the delivery rate jumps from near 0% to near 100%. In each figure, this threshold decreases as the ellipse factor increases because the bigger the ellipse factor is, the more vertices contribute in message delivery, and consequently, the probability of the message to reach the target, which is nothing but the message delivery rate, increases. For both figures the transmission range is fixed to 1 unit and the number of vertices is also fixed to 1000 and 2000, respectively.

From figure 1, we observe that when the graph becomes denser (more vertices in this case), the curve jumps earlier, and the reason is each time a vertex relays the message, more nodes get the message (due to more neighbors in dense graphs) so the probability that the message reaches the target increases.

One important observation is as follows: as we increase the ellipse factor, the message delivery rate does not increase proportionally. Surprisingly, when the ellipse factor is around 1.8, the message delivery rate is almost as good as the one using global gossiping (i.e., the ellipse factor constraint is relaxed to ∞). The reason is where a bigger ellipse factor is used we are actually considering the vertices that are less helpful than the vertices which are already considered. Intuitively, the vertices, which are far away from the line connecting the source and target, do not help improving the message delivery rate.

We also observe that, for a fixed relay probability, when the graph is dense, even a narrow ellipse could guarantee a good rate of message delivery. Achieving the same delivery rate using the same relay probability, for a sparser graph, might not be possible, even if the ellipse factor is relaxed to infinity. In other words, the ellipse factor does not compensate the description of the graph density. For example, in figure 1(b), when the relay probability is 0.3 with ellipse factor of 1.4, the delivery rate is about 95% for $n = 2000$, while we cannot achieve this rate when $n = 1000$ (see figure 1(a)).

2. Number of nodes involved in message delivery as a function of relay probability for different values of ellipse factor. So far, we have concentrated on the transition phenomena of the delivery rate over the relay probability. Not only the delivery rate is important for the network performance, but also the number of vertices involved in the message delivery is important for the network life since the wireless devices are often powered by the batteries only.

The challenge is to find an *ellipse factor* and a *relay probability* such that not only the delivery rate is high (close to 100%) but also the number of vertices involved in the message delivery is as small as possible. Actually the ellipse factor and the number of vertices involved in sending the message from source to target, work against each other. It means that if we choose a bigger ellipse factor, a higher delivery rate is achieved, on the other hand, lots of vertices will be involved in route discovery. In reverse, if we choose a small ellipse factor then fewer vertices will be involved but it may not have a good delivery rate.

As can be seen in figure 2, the relation between the number of vertices involved and the relay probability with respect to ellipse factors is close to linear. The bigger the relay probability, the more number of vertices will be involved in the message delivery. The exact relation between the number of vertices and relay probability is not simple. Clearly, the farther it is from the source, the less probability it will get the the message to relay.

In figure 2 when the ellipse factor is infinity, we are actually flooding the network with a uniform relay probability, and when this relay probability is 1, the network is completely

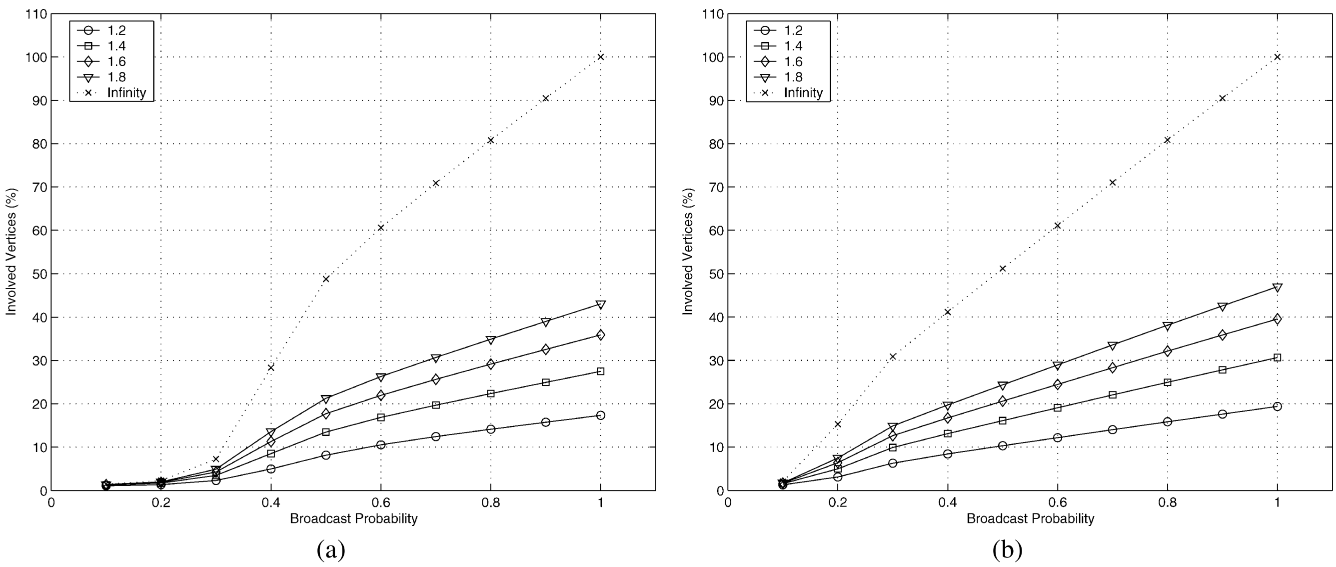


Figure 2. Number of nodes involved in message delivery as a function of relay probability for different values of ellipse factor. Here transmission range is 1. (a) Number of vertices is 1000. (b) Number sof vertices is 2000.

flooded, i.e., traditional flooding, so all nodes have the chance to contribute in message delivery.

Assume that we want to have the delivery rate more than 99%, first consider the case in which we have 1000 nodes, illustrated in figures 1(b) and 2(b).

We build the table 1 as follows: for each ellipse factor, we can find the needed relay probability to guarantee the message delivery above 99% from figure 1, and then by knowing the values of ellipse factor and the relay probability we can find the percentage of vertices that are involved from figure 2.

For example, to achieve this rate (above 99%) when ellipse factor is 1.2, the relay probability must be at least 0.9 (see figure 1). Then having these two values fixed, we can find the number of nodes that are involved from figure 2, which would be about 15% of all vertices. Doing the same thing for different values of ellipse factor, we get table 1.

The first column is the different ellipse factors we simulated, and the second column is the corresponding relay probability in our regional gossip method to guarantee this fixed delivery rate 99%, and the third column is the percentage of vertices that are involved in our regional gossiping (i.e., relaying the message).

Table 1 shows that we could involve only 15% of vertices to guarantee the message delivery rate above 99% when the ellipse factor is 1.2. If we do the same calculations where there are 2000 nodes then only 10% of vertices will be involved (see figures 1 and 2) by choosing ellipse factor 1.2 and relay probability 0.5.

Table 1
Percentage of the vertices involved in message delivery.

Ellipse factor	Relay probability	Vertices involved (%)
1.2	0.9	15
1.4	0.8	22
1.6	0.7	25
1.8	0.7	30
infinity	0.7	70

So far the transmission rang was fixed to 1. We were motivated to study the effect of transmission range as well. We then tried different values of transmission range. Obviously the larger the transmission range is, the denser the graph will be and as mentioned before that causes the jump to occur earlier.

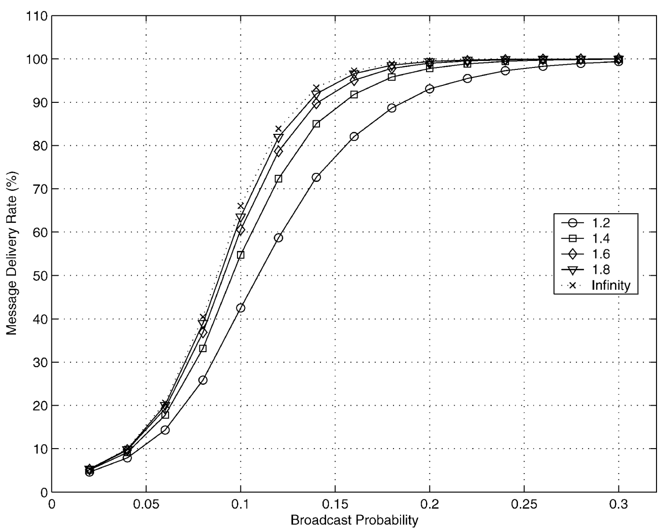
In figure 3 the transmission range is 2. See how similar figure 1 and figure 3 are, the only difference between these two figures is the probability at which the jump occurs for any fixed ellipse factor. Since in delivery rate happens earlier and quicker when the transmission range increases, we plot the figures using relay probability range [0, 0.3] for $r = 2$, instead of [0, 1] for $r = 1$.

Again assume that we want to have the delivery rate more than 99%. Consider the case in which we have 1000 nodes, but the transmission range is 2 (figures 3(a) and (b)).

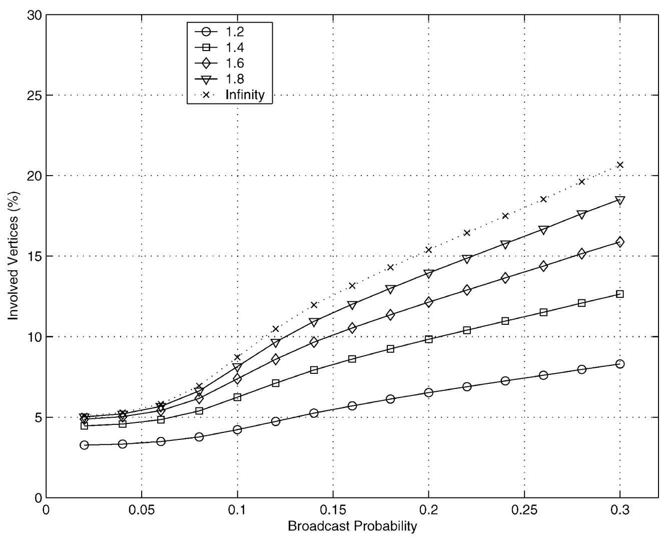
We build table 2 as we built table 1: for each ellipse factor. We can find the relay probability that guarantees the message delivery rate above 99% from figure 3(a), and then by knowing the values of ellipse factor and the relay probability we can find the percentage of vertices involved in message delivery from figure 3(b).

For example, to achieve this rate (above 99%) when ellipse factor is 1.2, the relay probability must be 0.3 (see figure 3(a)). Then having these two values fixed, we can find the number of nodes involved from figure 3(b), which would be about 8%. Doing the same thing for different values of ellipse factor, we get table 2.

Table 2 shows that we could involve only 8% of vertices to guarantee the message delivery rate above 99% for networks of 1000 nodes and with transmission range equal to 2. If we do the same calculations for networks of 2000 nodes



(a)



(b)

Figure 3. (a) Message delivery rate as a function of relay probability for different values of ellipse factor. Here number of vertices is 1000 and transmission range is 2. (b) Number of nodes involved in message delivery as a function of relay probability for different values of ellipse factor. Here number of vertices is 1000 and transmission range is 2.

with transmission range equal to 2, then only 6% of vertices will be involved (figures are not shown here).

3. Message delivery rate as a function of relay probability for different values of transmission range. So far we plotted the message delivery rate as a function of relay probability for different values of ellipse factor. Let us replace the ellipse factor parameter with transmission range and see how the graph behaves.

As you can see in figure 4, transmission range plays a very important role in message delivery (see how far the curves are from each other). As the transmission range is increased, the delivery rate improves significantly as opposed to the situation we had earlier with ellipse factor. The reason is when the transmission range is bigger then each node will be connected to more nodes, in other words the graph density increases. Thus, each time a node relays the message, more nodes will get it and the probability that the message dies out becomes smaller. Here in figure 4, the ellipse factor is fixed to 1.6.

We built table 3 as follows: for each transmission range, we can find the relay probability that guarantees the message delivery rate above 99% from figure 4(a), and then by knowing the values of transmission range and the relay probability we can find the percentage of vertices involved from figure 4(b).

For example, to achieve this rate (above 99%) when transmission range is 1, the relay probability must be at least 0.8, (see figure 4(a)). Then having these two values fixed, we can find the percentage of vertices involved from figure 4(b), which would be about 30%. We get table 3 by doing the same calculation for different values of transmission range.

Table 3 illustrates the number of vertices involved in the regional gossip routing to guarantee a fixed delivery rate 99% for networks of 1000 nodes with ellipse factor 1.6. Observe that, all these curves intersect in a common point when the relay probability is 1. Because the ellipse factor is fixed, changing the transmission range does not change the number of nodes that are *inside* ellipse, which is total number of vertices involved in message delivery when the relay probability is 1. Actually it is possible to have a node in the ellipse which does not contribute in message delivery even when the relay probability is 1, but that is very unlikely. It happens only when a node in the ellipse doesn't have any neighbor inside the ellipse. In our simulations this scenario happened 2 times out of 180,000,000 iterations.

Another observation is that we get different curves for different transmission ranges. Typically, when the transmission range is larger, more nodes inside this ellipse will be involved in the message delivery.

4. Message delivery rate as a function of relay probability for different number of nodes. In our simulations we studied networks with different densities in two different ways.

Table 2
Percentage of the vertices involved in message delivery.

Ellipse factor	Relay probability	Vertices involved (%)
1.2	0.3	8
1.4	0.24	11
1.6	0.22	13
1.8	0.20	14
infinity	0.20	15

Table 3
Percentage of the vertices involved in message delivery.

Transmission range	Probability	Vertices involved (%)
1.0	0.8	30
1.5	0.5	20
2.0	0.3	14
2.5	0.14	12
3.0	0.11	11.71

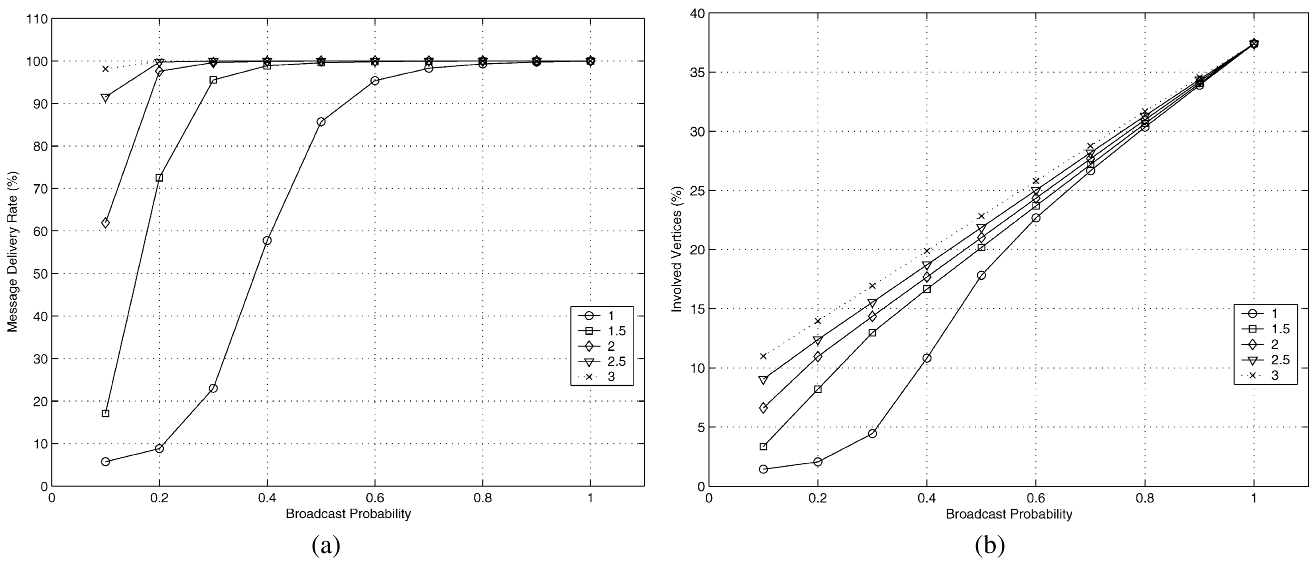
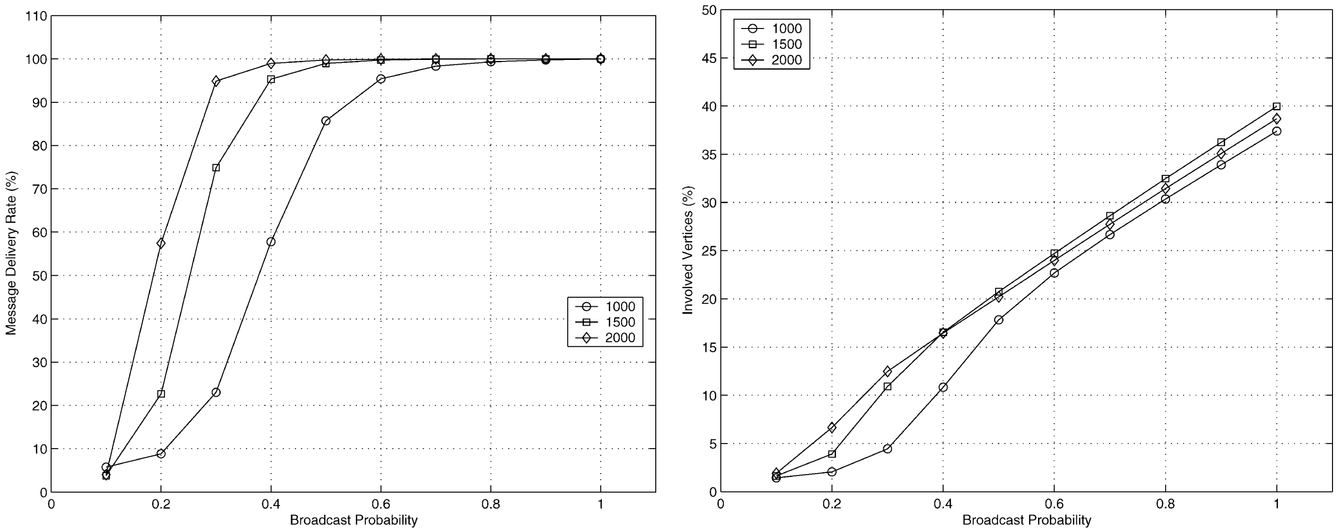


Figure 4. (a) Message delivery rate as a function of relay probability for different values of transmission range. Here number of vertices is 1000 and the ellipse factor is 1.6. (b) Number of nodes involved in message delivery as a function of relay probability for different values of transmission range. Here number of vertices is 1000 and ellipse factor is 1.6.



(a) (b)

Figure 5. (a) Message delivery rate as a function of relay probability for different number of nodes. Here ellipse factor is 1.6 and transmission range is 1. (b) Number of nodes involved as a function of relay probability for different number of nodes. Here ellipse factor is 1.6 and transmission range is 1.

First, as described in the previous section, we studied networks with fixed number of vertices and different transmission ranges. Now we study networks with fixed transmission range and different number of vertices placed in a 15×15 square. In both cases we expect the similar results if the network densities are similar.

As you can see in figure 5, the number of vertices plays an important role in message delivery (see how far the curves are from each other). Here we have the same reasoning as the previous section. As the number of vertices is increased, the delivery rate improves significantly. The reason is when there are more vertices in the same area, the graph becomes denser. Thus, each time a node relays the message more nodes will get it and the probability that the message dies out becomes smaller.

Now let us look at the percentage of nodes that are involved in message delivery as a function of relay probability for different number of nodes (see figure 5). Remember that in this case ellipse factor and transmission range are fixed. Here we have the same ellipse with different number of vertices inside them. When there are more vertices in the same area the message is delivered with higher probability since more nodes will relay the message. Notice that, given a fixed relay probability, when the node density exceeds some threshold (depending on the relay probability) almost *all* nodes inside the ellipse will receive the message, thus, have the chance to relay the massage. In other words, if the relay probability is low, high message delivery rate still can be achieved if the graph is dense enough and if the graph is sparse, high message delivery rate still can be achieved by increasing the relay probability. On the other hand, larger relay probability will involve more nodes in message delivery (the number of nodes involved is almost linear to the relay probability as shown in right figure of figure 5).

4.3. Message delivery rate as a function of ellipse factor

We can look at the problem from a totally different point of view. So far we have concentrated on the transition phenomena of the delivery rate over the relay probability. In other words, in all figures the X-axis was the relay probability. Now let us see how the network behaves if we use different ellipse factors while some other parameters are fixed. We found that, regardless of the network density and relay probability, increasing the ellipse factor does not improve the message delivery rate significantly.

1. Message delivery rate as a function of ellipse factor for different values of transmission range. First let us fix the relay probability and the number of vertices. Remember that to change the message delivery rate dramatically we can either increase the relay probability or increase the network density. As can be seen in figure 6 there is no jump. In other words, increasing the ellipse factor does not improve the message delivery rate dramatically.

Figure 6 shows when the relay probability is fixed, regardless of the value of ellipse factor, the graph density must be above some threshold to guarantee a high message delivery. As you can see in figure 6(a) when the transmission range is less than 1.5 then the delivery rate is always below 20% even if the ellipse factor constraint is relaxed (the case where ellipse factor constraint is relaxed and not shown in figure 6).

As it is expected if we set the relay probability to a higher value then the delivery rate would be higher. This is illustrated in figure 6: if we increase the value of the relay probability (from figure 6(a) to figure 6(b)) all curves will be shifted up.

2. Message delivery rate as a function of ellipse factor for different number of vertices. As mentioned earlier, the network density can be increased either by increasing the transmission range or by increasing the number of vertices. Now

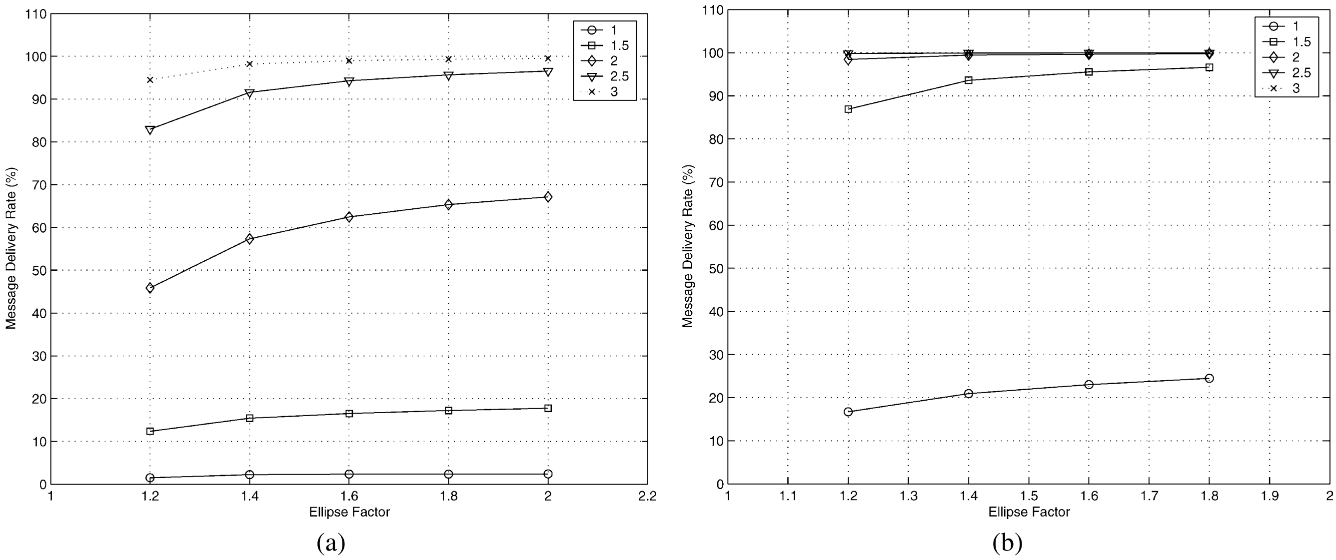


Figure 6. Message delivery rate as a function of relay probability for different values of transmission range. Here number of vertices is 1000. Relay probability is (a) 0.1, (b) 0.3.

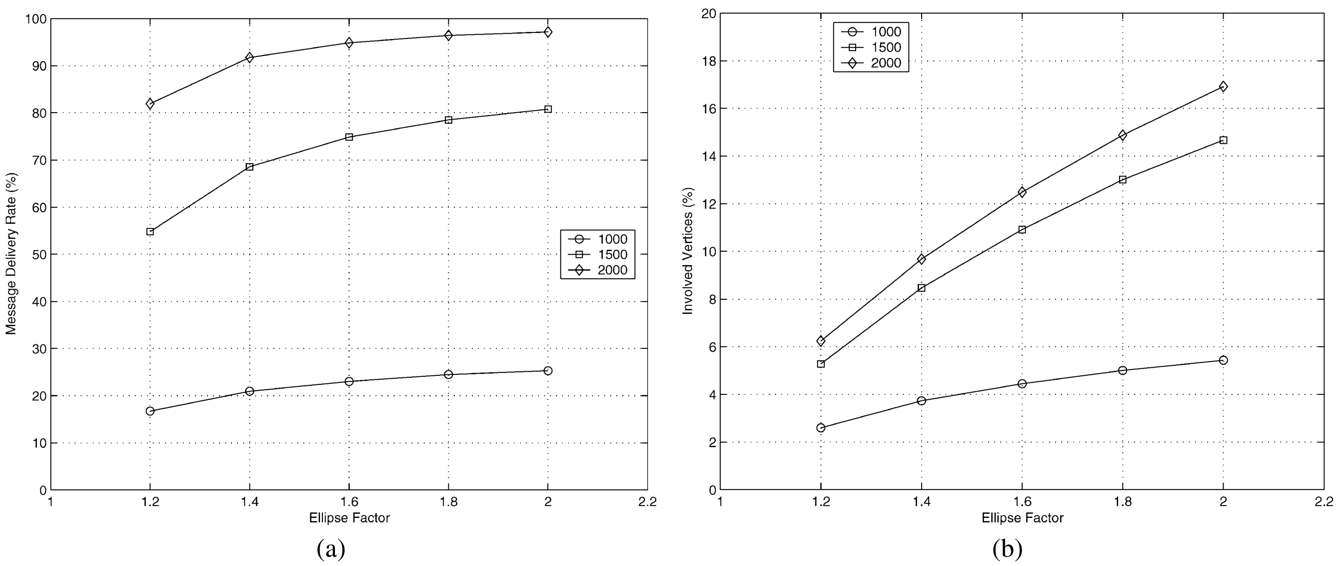


Figure 7. (a) Message delivery rate as a function of ellipse factor for different number of vertices. Here transmission range is 1 and relay probability is 0.3. (b) Number of nodes involved in message delivery as a function of ellipse factor for different number of vertices. Here transmission range is 1 and relay probability is 0.3.

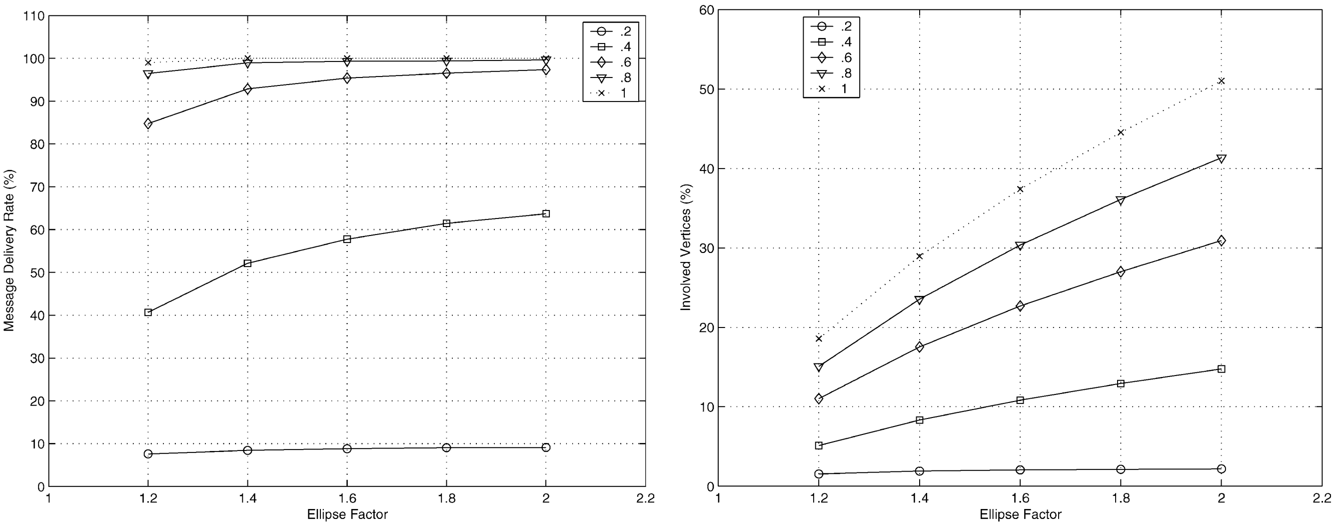
we replace the transmission range of the previous section with number of vertices and we expect similar results. In other words, let us fix the relay probability and the transmission range to see the delivery rate as a function of ellipse factor for different number of vertices.

Again, as can be seen in figure 7 there is no jump. In other words, increasing the ellipse factor does not improve the message delivery rate dramatically.

3. Message delivery rate as a function of ellipse factor for different values of relay probability. In the previous two sections, we studied the effect of ellipse factor in networks with different densities, in this section instead of changing the network density, we change the relay probability. Thus, in this section, the network density is fixed. Specifically, we study the message delivery rate (as a function of ellipse factor for different values of relay probability) by fixing the number of nodes and the transmission range.

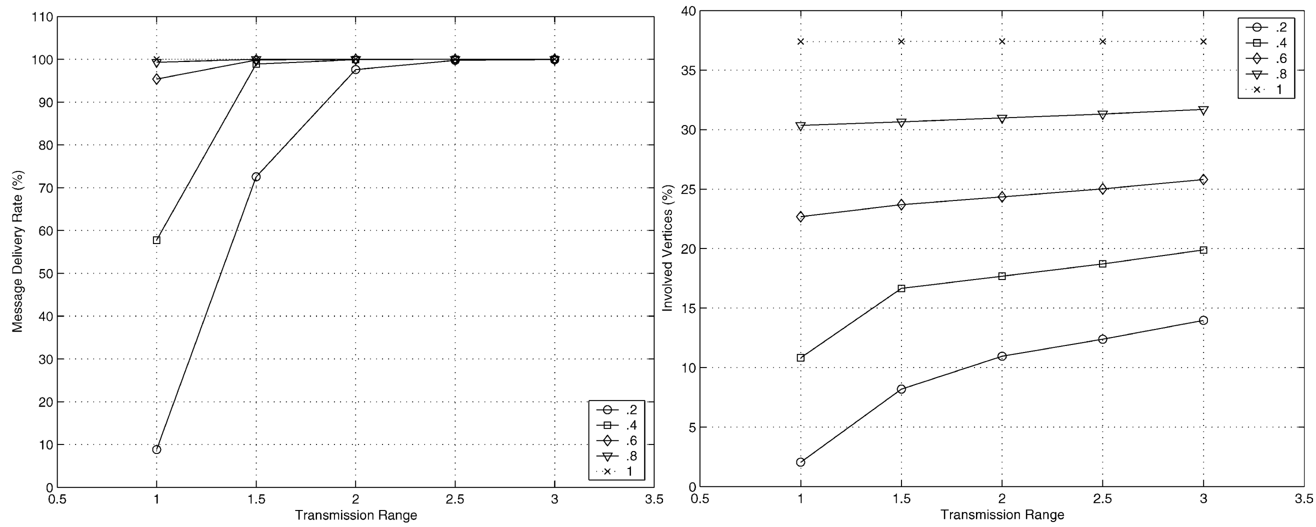
In figure 8 when the relay probability is below some threshold, a high delivery rate cannot be achieved even when the ellipse factor constraint is relaxed. Figure 8 is similar to figures 6 and 7 due to the fact that a high relay probability can compensate the sparseness of the network and vice versa.

Intuitively, all the discussions of the two previous sections apply to this section too. For example, when the network density is larger than some threshold, the number of vertices involved is almost linear to the ellipse factor, see figures 7 and 8.



(a) (b)

Figure 8. (a) Message delivery rate as a function of ellipse factor for different values of relay probability. Here transmission range is 1 and number of vertices is 1000. (b) Number of nodes involved in message delivery as a function of ellipse factor for different values of relay probability. Here transmission range is 1 and number of vertices is 1000.



(a) (b)

Figure 9. (a) Message delivery rate as a function of transmission range for different values of relay probability. Here ellipse factor is 1.6 and number of vertices is 1000. (b) Number of nodes involved in message delivery as a function of transmission range for different values of relay probability. Here ellipse factor is 1.6 and number of vertices is 1000.

4.4. Message delivery rate as a function of transmission range

We can look at the problem from a totally different point of view. So far the X-axis was the relay probability or the ellipse factor. Thus, for each curve in figures discussed in previous sections, the network density was fixed. But if we choose the transmission range or number of vertices as the X-axis then the graph density changes for each curve. We first study the case where the X-axis is the transmission range and in the next section we study the case where the X-axis is the the number of vertices.

1. Message delivery rate as a function of transmission range for different values of relay probability. First let us fix the ellipse factor and the number of vertices. We expect to see jump because in each curve the graph density changes and also we expect to see curves that are far from each other due to the fact that for each curve the relay probability is fixed.

As you can see in figure 9 when the relay probability is bigger the jump occurs earlier. This figure is similar to figure 4 due to the fact that the relay probability and transmission range both improve the message delivery rate significantly.

2. Message delivery rate as a function of transmission range for different values of ellipse factor. Let us fix the number of vertices and the relay probability to see the delivery rate as a function of transmission range for different values of ellipse factor. As you can see in figure 10, like figure 1, as we in-

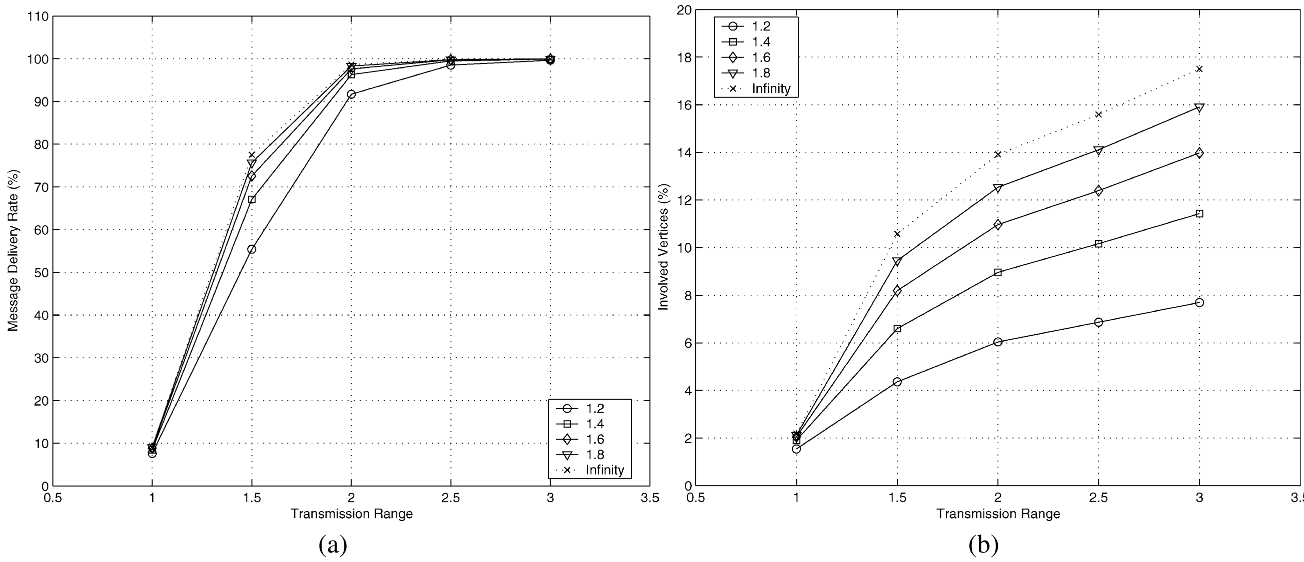


Figure 10. (a) Message delivery rate as a function of transmission range for different values of ellipse factor. Here number of vertices is 1000 and relay probability is 0.2. (b) Number of nodes involved in message delivery as a function of transmission range for different values of ellipse factor. Here number of vertices is 1000 and relay probability is 0.2.

crease the ellipse factor, the message delivery rate does not increase proportionally. The only difference between figure 10 and figure 1 is: in figure 10 the network density changes in each curve but in figure 1 the relay probability changes in each curve. Since increasing either the relay probability or transmission range improves the message delivery, exchanging those will lead to similar results. Observe that when the ellipse factor is 1.8, the delivery rate is almost the same as the global gossiping.

Observe that, in figure 10, the number of vertices involved in message delivery is almost linear after the transmission range is larger than some threshold (almost 2). When the transmission range is small, the number of nodes involved is small since the message quickly dies out (the relay probability is 0.2 here).

3. Message delivery rate as a function of transmission range for different number of vertices. Now let us fix the ellipse factor and the relay probability to study the message delivery rate (as a function of transmission range for different number of vertices). Since the transmission range and the number of vertices are factors that affect the network density, not only the network density changes in each curve, but also the network density is different for each curve.

In figure 11, not only the jump occurs (due to the change of graph density), but also the curves are far from each other (again due to the change of graph density).

Observe that, the number of vertices involved in the message delivery increases almost proportionally to the transmission range when the relay probability is set to 0.2 (see figure 11(a)). However, when the relay probability increases, say 0.7, the percentage of the number of vertices involved is almost constant, see figure 11(b).

4.5. Message delivery rate as a function of number of vertices

The last parameter is the number of vertices. Since both transmission range and number of vertices affect the network density, we expect similar results like the previous section.

1. Message delivery rate as a function of number of vertices for different values of relay probability. Now let us fix the ellipse factor and the transmission range to see delivery rate as a function of number of vertices for different values of relay probability. As shown in figure 12, if we use a big enough relay probability, a high delivery rate is guaranteed. But when the relay probability is small then we need a large number of vertices to compensate this small relay probability to guarantee a high delivery rate.

2. Message delivery rate as a function of number of vertices for different values of ellipse factor. Now let us fix the relay probability and the transmission range to see delivery rate as a function of number of vertices for different values of ellipse factor. Illustrated by figure 13, like figure 10, as we increase the ellipse factor, the message delivery rate does not increase proportionally.

3. Message delivery rate as a function of number of vertices for different values of transmission range. Now let us fix the ellipse factor and the relay probability to see delivery rate as a function of number of vertices for different values of transmission range. As you can see in figure 14, the bigger the number of vertices is, the earlier the jump occurs.

Figures 12–14 study the number of vertices that are involved in the message delivery. In these figures, we found that there are some strange jumps when the number of

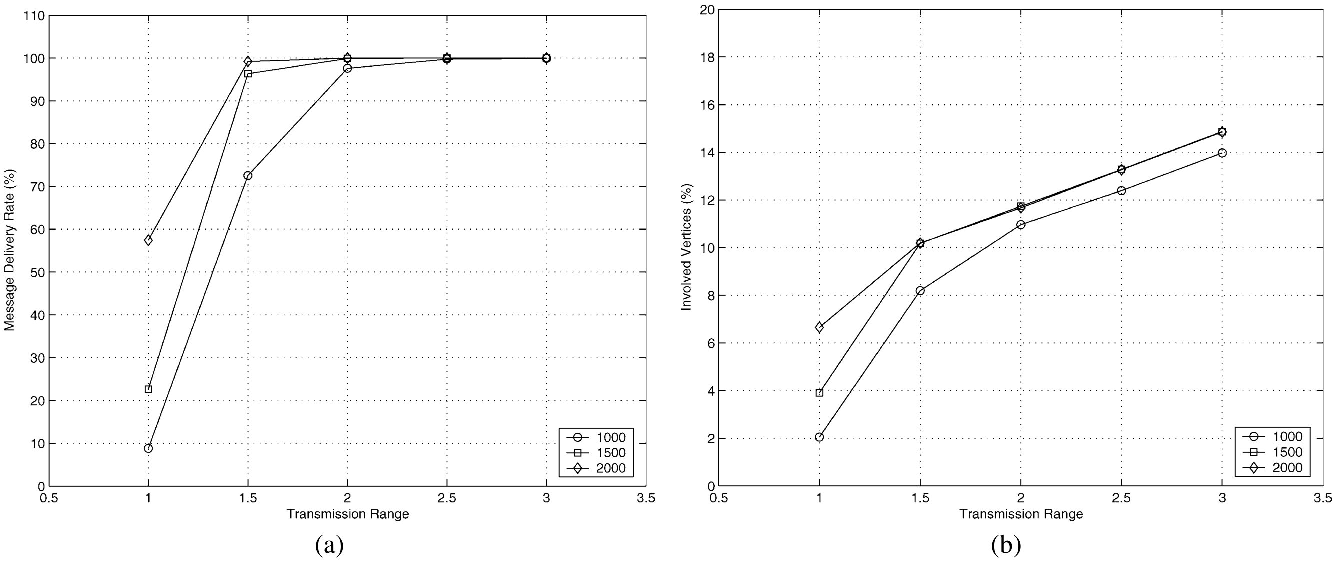


Figure 11. (a) Message delivery rate as a function of transmission range for different number of vertices. Here ellipse factor is 1.6 and relay probability is 0.2. (b) Number of nodes involved in message delivery as a function of transmission range for different number of vertices.Here ellipse factor is 1.6 and relay probability is 0.2.

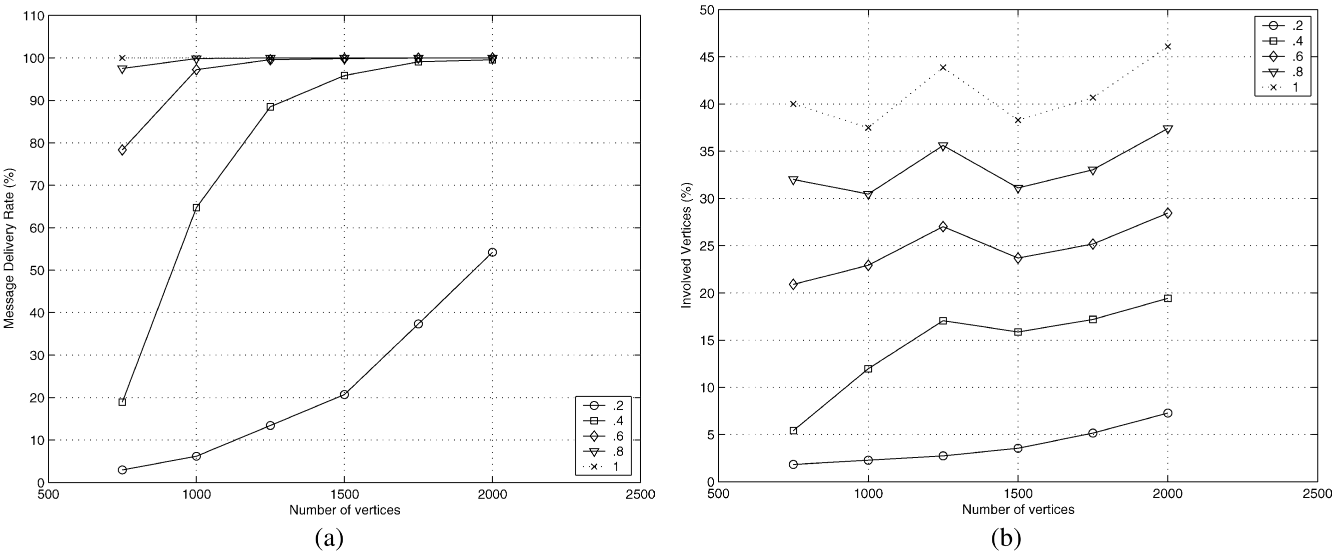
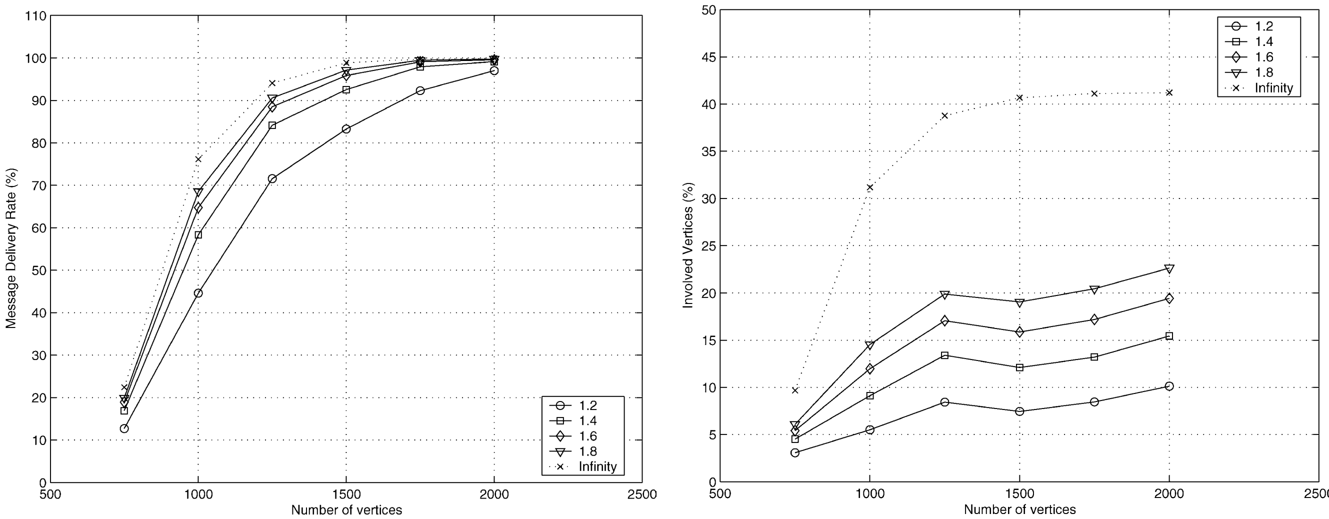


Figure 12. (a) Message delivery rate as a function of number of vertices for different values of relay probability. Here ellipse factor is 1.6 and transmission range is 1. (b) Number of nodes involved in message delivery as a function of number of vertices for different values of relay probability. Here ellipse factor is 1.6 and transmission range is 1.

vertices is around 1250. We are studying why this happens.

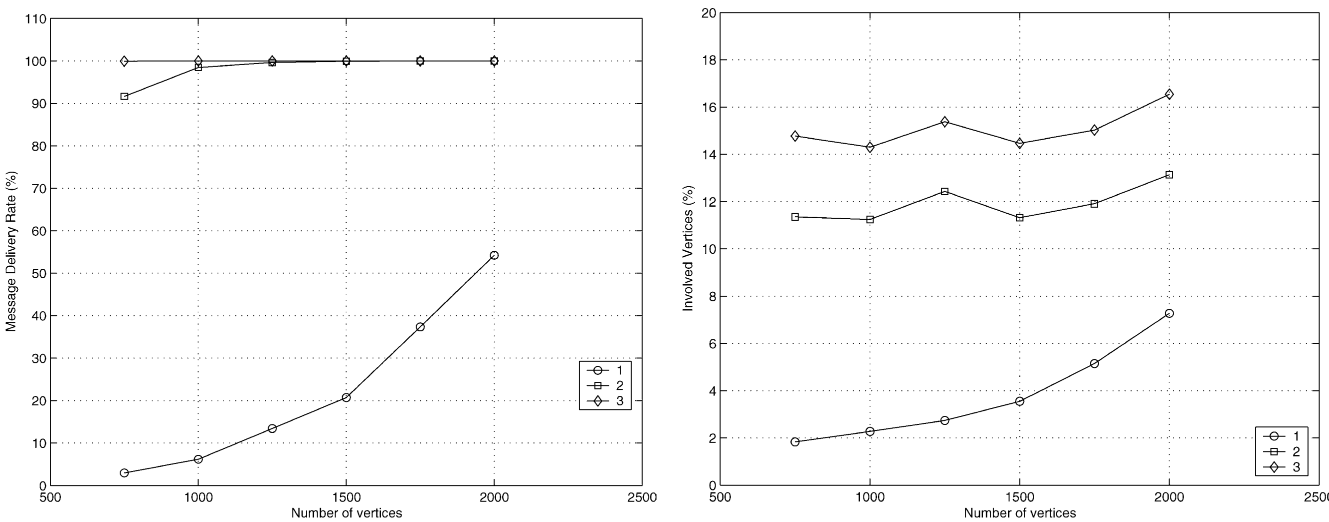
5. Fault tolerance

To study the fault tolerance of the ad-hoc networks, we simulated the cases in which the target receives the message more than once. The figure 15 shows the number of times that the message is delivered to the target at least twice as a function of relay probability for different values of ellipse factor. If target has h neighbors inside the ellipse in the best case (i.e., all neighbors of the target receive the message) we expect the message to be delivered $p \times h$ times. Note that if the target has only one neighbor inside the ellipse, then the target has no chance to receive the message more than once. Observe that figure 15 is a little bit misleading. It shows that with a narrow ellipse and the replay probability fixed to 1 the probability that the target receives the message more than once is below 95%. The reason is in our simulations, the source–target pairs are chosen randomly, so in some cases the target is only one hop away from the source, thus the target gets the message for sure but at the same time, due to the closeness of source and target, there might not be another neighbor inside the ellipse for target. Thus the target has no chance to receive the message more than once. In other words, in some cases, although the message delivery rate is 100%, the chance that the target receives the message more than once is 0%.



(a) (b)

Figure 13. (a) Message delivery rate as a function of number of vertices for different values of ellipse factor. Here relay probability is 0.4 and transmission range is 1. (b) Number of nodes involved in message delivery as a function of number of vertices for different values of ellipse factor. Here relay probability is 0.4 and transmission range is 1.



(a) (b)

Figure 14. (a) Message delivery rate as a function of number of vertices for different values of transmission range. Here ellipse factor is 1.6 and relay probability is 0.2. (b) Number of nodes involved in message delivery as a function of number of vertices for different values of transmission range. Here ellipse factor is 1.6 and relay probability is 0.2.

6. Conclusion and future work

We proposed a regional gossip approach, where only the nodes within some region forward the routing message with some probability, to reduce the overhead of the routing protocol imposed on the network. We showed how to set the forwarding probability based on the region and the estimated network density both by theoretical analysis and by extensive simulations. Our simulations showed that the number of messages generated using this approach is less than the simple global flooding (up to 94%), which already saves many messages compared with global flooding.

Hass et al. [24] expected that the global gossiping combined with the cluster-based routing can further improve the performance. We doubt this due to two reasons: (1) the backbone formed by clusterheads are already very sparse, and to guarantee that all nodes receive the messages, the gossiping probability is very high; and (2) the communication cost to maintain the backbone will also offset the benefit gained by global gossiping, if there is any. We will conduct simulations to study this.

One of the main questions remaining to be studied is to use non-uniform ellipse factors. In our simulations, the ellipse factor is uniform regardless of the distance between source and target. We believe that using a bigger ellipse factor, when the source and target are close, will get better results.

Another question is studying networks with different densities, meaning that instead of trying different transmission

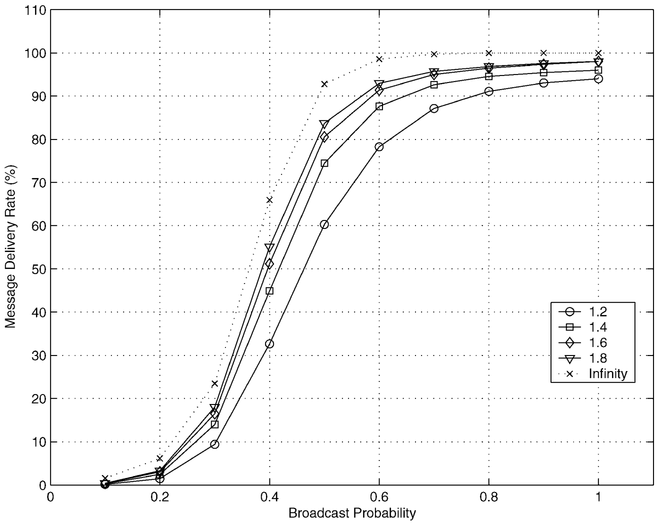


Figure 15. The number of times that the message receives the target more than once as a function of relay probability for different values of ellipse factor. Here transmission range is 1 and number of vertices is 1000.

ranges and different number of nodes, networks with different densities can be studied. To generate a network with a given density with respect to transmission range, we can keep adding nodes to the network until the desired density is reached.

We had assumed that two nodes can always communicate if their distance is no more than the transmission range. However, this is not totally true practically. Some pair of nodes cannot communicate at all even if they are close. We can model this by assigning another link probability p_l: a link exist with probability p_l. Here probability p_l could be uniform or dependent on the distance between the pair of nodes.

References

[1] S. Capkun, M. Hamdi and J.P. Hubaux, GPS-free positioning in mobile ad-hoc networks, in: *Proc. Hawaii Internat. Conf. on System Sciences* (2001).

[2] C. Bettstetter, On the minimum node degree and connectivity of a wireless multihop network, in: *3rd ACM Internat. Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02)* (June 2002).

[3] G. Călinescu, I. Mandoiu and A. Zelikovsky, Symmetric connectivity with minimum power consumption in radio networks, in: *IFIP–TCS* (2002) to appear.

[4] A. Clementi, P. Penna and R. Silvestri, The power range assignment problem in radio networks on the plane, in: *XVII Symposium on Theoretical Aspects of Computer Science (STACS'00)*, Lecture Notes in Computer Science, Vol. 1770 (Springer, New York, 2000) pp. 651–660.

[5] P. Gupta and P. R. Kumar, Critical power for asymptotic connectivity in wireless networks, in: *Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W.H. Fleming*, eds. W.M. McEneaney, G. Yin and Q. Zhang (1998).

[6] J. Monks, V. Bharghavan and W.-M Hwu, A power controlled multiple access protocol for wireless packet networks, in: *IEEE INFOCOM* (2001).

[7] M. Sanchez, P. Manzoni and Z. Haas, Determination of critical transmission range in ad-hoc networks, in: *Multiaccess, Mobility and Teletraffic for Wireless Communications (MMT'99)* (1999).

[8] M. Bahramgiri, M.T. Hajiaghayi and V.S. Mirrokni, Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks, in: *Proc. of the 11th Annual IEEE Internat. Conf. on Computer Communications and Networks (ICCCN)* (2002) pp. 392–397.

[9] L. Hu, Topology control for multihop packet radio networks, IEEE Transactions on Communications 41(10) (1993).

[10] X.-Y. Li, P.-J. Wan, Y. Wang and O. Frieder, Sparse power efficient topology for wireless networks, Journal of Parallel and Distributed Computing (2002) to appear; preliminary version appeared in: *ICCCN* (2001).

[11] X.-Y. Li, G. Calinescu, P.-J. Wan and Y. Wang, Localized delaunay triangulation with application in wireless ad hoc networks, IEEE Transactions on Parallel and Distributed Systems (2003); short version appeared in: *IEEE INFOCOM* (2002).

[12] X.-Y. Li, G. Calinescu and P.-J. Wan, Distributed construction of planar spanner and routing for ad hoc wireless networks, in: *21st Annual Joint Conf. of the IEEE Computer and Communications Societies (INFOCOM)*, Vol. 3 (2002).

[13] R. Ramanathan and R. Rosales-Hain, Topology control of multihop wireless networks using transmit power adjustment, in: *IEEE INFOCOM* (2000).

[14] R. Wattenhofer, L. Li, P. Bahl and Y.-M. Wang, Distributed topology control for wireless multihop ad-hoc networks, in: *IEEE INFOCOM'01* (2001).

[15] S. Singh, M. Woo and C. Raghavendra, Power-aware routing in mobile ad hoc networks, in: *Proc. of MOBICOM* (1998) pp. 181–190.

[16] I. Stojmenovic and X. Lin, Power aware localized routing in ad hoc wireless networks, IEEE Transactions on Parallel and Distributed Systems 12(11) (2001) 1122–1133.

[17] J. Wu, F. Dai, M. Gao and I. Stojmenovic, On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks, IEEE/KICS Journal of Communication and Networks 4(1) (2002) 59–70.

[18] P. Bose, P. Morin, I. Stojmenovic and J. Urrutia, Routing with guaranteed delivery in ad hoc wireless networks, ACM/Kluwer Wireless Networks 7(6) (2001) 609–616, also in: *3rd Internat. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications* (1999) pp. 48–55.

[19] B. Das and V. Bharghavan, Routing in ad-hoc networks using minimum connected dominating sets, in: *1997 IEEE Internat. Conf. on Communications (ICC'97)*, Vol. 1 (1997) pp. 376–380.

[20] C.E. Perkins and P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, Computer Communications Review (October 1994) 234–244.

[21] D.B. Johnson and D.A. Maltz, Dynamic source routing in ad hoc wireless networks, in: *Mobile Computing*, eds. Imielinski and Korth, Vol. 353 (Kluwer Academic, Dordrecht, 1996).

[22] B. Karp and H.T. Kung, Gpsr: Greedy perimeter stateless routing for wireless networks, in: *ACM/IEEE Internat. Conf. on Mobile Computing and Networking* (2000).

[23] Y.-B. Ko and N.H. Vaidya, Using location information to improve routing in ad hoc networks, Technical Report, Department of Computer Science, Texas A&M University (1997).

[24] Z. Haas, J. Halpern and L. Li, Gossip-based ad hoc routing, in: *IEEE INFOCOM* (2002).

[25] S. Murthy and J. Garcia-Luna-Aceves, An efficient routing protocol for wireless networks, *ACM Mobile Networks and Applications* (Special Issue on Routing in Mobile Communication Networks) 1(2) (1996).

[26] V. Park and M. Corson, A highly adaptive distributed routing algorithm for mobile wireless networks, in: *IEEE INFOCOM* (1997).

[27] C. Perkins, Ad-hoc on-demand distance vector routing, in: *MILCOM '97* (November 1997).

[28] C. Perkins and P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing, in: *Proc. of the ACM SIGCOMM* (October 1994).

[29] P. Sinha, R. Sivakumar and V. Bharghavan, CEDAR: Core extraction distributed ad hoc routing algorithm, IEEE Journal on Selected Areas in Communications 17(8) (1999) 1454–1465.

[30] P. Sinha, R. Sivakumar and V. Bharghavan, Enhancing ad hoc routing with dynamic virtual infrastructures, in: *Proc. of IEEE INFOCOM 2001*, Vol. 3 (2001) pp. 1763–1772.

[31] J. Wu and H. Li, A dominating-set-based routing scheme in ad hoc wireless networks, Telecommunication Systems (Special Issue on Wireless Networks) 3 (2001) 63–84.

[32] M. Joa-Ng and I.-T. Lu, A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks, IEEE Journal on Selected Areas in Communications 17(8) (1999) 1415–1425.

[33] E. Royer and C. Toh, A review of current routing protocols for adhoc mobile wireless networks, IEEE Personal Communications (April 1999).

[34] S. Ramanathan and M. Steenstrup, A survey of routing techniques for mobile communication networks, Mobile Networks and Applications (1996) 89–104.

[35] M. Mauve, J. Widmer and H. Harenstein, A survey on position-based routing in mobile ad hoc networks, IEEE Network Magazine 15(6) (2001) 30–39.

[36] P. Hall, On continuum percolation, The Annals of Probability 13(4) (1985).

[37] R. Meester and R. Roy, *Continuum Percolation* (Cambridge Univ. Press, Cambridge, 1996).

[38] S. Basagni, I. Chlamtac, V.R. Syrotiuk and B.A. Woodward, A distance routing effect algorithm for mobility (DREAM), in: *Proc. of ACM/IEEE MOBICOM'98* (1998).

[39] Z. Haas and B. Liang, Ad-hoc mobility management with uniform quorum systems, IEEE/ACM Transactions on Networking 7(2) (1999) 228–240.

[40] I. Stojmenovic, A routing strategy and quorum based location update scheme for ad hoc wireless networks, Technical Report TR-99-09, Computer Science, SITE, University of Ottawa (1999).

[41] I. Chlamtac and A. Farago, A new approach to design and analysis of peer to peer mobile networks, Wireless Networks 5 (1999) 149–156.

[42] P.-J. Wan, K.M. Alzoubi and O. Frieder, Distributed construction of connected dominating set in wireless ad hoc networks, in: *INFOCOM* (2002).

[43] K.M. Alzoubi, P.-J. Wan and O. Frieder, New distributed algorithm for connected dominating set in wireless ad hoc networks, in: *HICSS*, Hawaii (2002).

[44] K. Alzoubi, P.-J. Wan and O. Frieder, Message-optimal connected-dominating-set construction for routing in mobile ad hoc networks, in: *3rd ACM Internat. Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02)* (2002).

[45] K.N. Amouris, S. Papavassiliou and M. Li, A position based multi-zone routing protocol for wide area mobile ad-hoc networks, in: *Proc. of the 49th IEEE Vehicular Technology Conf.* (1999) pp. 1365–1369.

[46] G. Călinescu, I. Măndoiu, P.-J. Wan and A. Zelikovsky, Selecting forwarding neighbors in wireless ad hoc networks, in: *ACM DialM* (2001).

[47] M. Penrose, The longest edge of the random minimal spanning tree, Annals of Applied Probability 7 (1997) 340–361.

[48] X.-Y. Li, Y. Wang, C.-W. Yi, P.-J. Wan and O. Frieder, Robust wireless ad hoc networks, in: *IEEE ICC* (2003) accepted for publication.

[49] P.-J. Wan, C.-W. Yi, X.-Y. Li, Y. Wang and O. Frieder, Asymptotic distribution of critical transmission range for k-connectivity in wireless ad hoc networks (2002) submitted for publication.

[50] B. Bollobás, *Random Graphs* (Cambridge Univ. Press, Cambrige, 2001).

[51] H. Dette and N. Henze, Some peculiar boundary phenomena for extremes of rth nearest neighbor links, Statistics & Probability Letters 10 (1990) 381–390.

[52] M. Penrose, Extremes for the minimal spanning tree on normally distributed points, Advances in Applied Probability 30 (1998) 628–639.

[53] M. Penrose, On k-connectivity for a geometric random graph, Random Structures and Algorithms 15 (1999) 145–164.

[54] M. Penrose, A strong law for the longest edge of the minimal spanning tree, Annals of Probability 27 (1999) 246–260.

[55] D.M. Blough, M. Leoncini, G. Resta and P. Santi, On the symmetric range assignment problem in wireless ad hoc networks, in: *Proc. of the 2nd IFIP Internat. Conf. on Theoretical Computer Science* (2002).

[56] C. Cooper and A. Frieze, On the connectivity of random kth nearest neighbour graphs, Combinatorics, Probability and Computing 4 (1995) 343–362.

[57] M. Grossglauser and D. Tse, Mobility increases the capacity of ad-hoc wireless networks, in: *INFOCOMM*, Vol. 3 (2001) pp. 1360–1369.

[58] P. Gupta and P. Kumar, Capacity of wireless networks, Technical Report University of Illinois, Urbana-Champaign (1999).

[59] O.D. Patrick, Connectivity in ad-hoc and hybrid networks, in: *IEEE INFOCOM* (2002).

[60] P. Santi and D.M. Blough, An evaluation of connectivity in mobile wireless ad hoc networks, in: *Proc. of IEEE DSN* (2002) pp. 89–98.

[61] F. Xue and P.R. Kumar, The number of neighbors needed for connectivity of wireless networks, Wireless Networks (2002) submitted.

[62] T. Lukovszki, New results of fault tolerant geometric spanners, in: *Workshop on Algorithms and Data Structures* (1999) pp. 193–204.

Xiang-Yang Li has been an Assistant Professor of Computer Science at the Illinois Institute of Technology since 2000. He hold M.S. (2000) and Ph.D. (2001) degree in computer science from University of Illinois at Urbana-Champaign. He received his Bachelor degree in computer science and Bachelor degree in business management from Tsinghua University, P.R. China in 1995. His research interests span the computational geometry, wireless ad hoc networks, optical networks, and cryptography. He is a member of the ACM and IEEE.
E-mail: xli@cs.iit.edu

Kousha Moaveninejad received the Bachelor in computer soft engineering from Sharif University of Technology, Tehran, Iran, in 1997. He joined the Department of Computer Science of Illinois Institute of Technology in 2000, as a M.S. student and received the Masters degree in 2002. He then continued his study as a Ph.D. student at Illinois Institute of Technology and his current research interests include wireless ad-hoc networks, computational geometry, algorithm design, and mobile computing.
E-mail: moavkoo@iit.edu

Ophir Frieder is the IITRI Professor of Computer Science at the Illinois Institute of Technology. His research interests span the general area of distributed information systems. He is a member of ACM and a fellow of the IEEE.
E-mail: ophir@cs.iit.edu