A New Routing Metric for High Throughput in Dense Ad Hoc Networks

Hoon Chang, Vishal Misra and Dan Rubenstein

***Abstract*— Routing protocols in most ad hoc networks use the length of paths as the routing metric. Recent findings have revealed that the minimum-hop metric can not achieve the maximum throughput because it tries to reduce the number of hops by containing long range links, where packets need to be transmitted at the lowest transmission rate. In this paper, we investigate the tradeoff between transmission rates and throughputs and show that in dense networks with uniform-distributed traffic, there exists the optimal rate that may not be the lowest rate. Based on our observation, we propose a new routing metric, which measures the expected capability of a path assuming the per-node fairness. We develop a routing protocol based on DSDV and demonstrate that the routing metric enhances the system throughput by 20% compared to the original DSDV.**

I. Introduction

It has been reported that routing protocols using the length of paths as a metric do not achieve the maximum possible throughput with nodes that supports multiple transmission rate [1], [2], [3]. In dense networks, source nodes have many alternative paths to destinations, which have different expected capacities. The shortest-path routing protocols usually choose paths that have the minimum number of hops rather than the maximum expected capacity.

This paper investigates the relationship between the transmission rates and path capacities. The main contributions of our paper are:

1) A model for the expected capacity of a path assuming per-node fairness by CSMA/CA protocol,
2) A new routing metric to select the best path with the maximum expected capacity, and
3) Modifications on DSDV routing protocol to validate our metric in terms of system throughput enhancement.

This paper proceeds as follows: section II presents our capacity model, propose a new path metric and analyze the path capacity. In section III, changes to DSDV routing protocols and implementation issues are discussed. Section IV shows simulation results. Related work is given in section V and we conclude the paper in section VI

Hoon Chang, Vishal Misra and Dan Rubenstein are with the Department of Computer Science and Electrical Engineering, Columbia University. {hchang,misra}@cs.columbia.edu, danr@ee.columbia.edu

II. Dense Network with Multiple Transmission Rates

We assume all nodes can transmit at multiple rates and the node density is high. Nodes use a CSMA/CA protocol to access. With different transmission rates, a node has different ranges of transmission. Figure 1 shows an example.

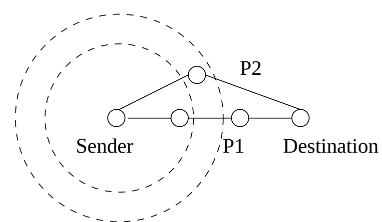

Fig. 1. Multiple Rates and Multiple Paths

Sender in Figure 1 has two transmission rates r and r'. Let $r < r'$ and $C(r)$ be the throughput at rate r without any interference. $C(r)$ is also dependent on packet sizes, but we assume the packet size is fixed. The maximum distance where a sender can reach without interference depends on a SNR value, which is a function of a transmission rate. In the figure, two transmission ranges from two transmission rates are shown in dotted circles. Since a network has a high density, the sender may have alternative paths to the destination as in the figure. If the sender chooses $P1$, packets will be transmitted at r' because the next hop is within the transmission range of r'. When $P2$ is selected, the sender transmits at rate r to the next hop.

Using shortest-path routing protocols, the sender always chooses the shortest path, $P2$. However, this does not always yield the best throughput. If every node can independently make packet transmissions, packets are forwarded in a pipelined way and following $P1$ will achieve higher throughput even through the path has more hops than $P2$.

The key factor of such parallel transmissions is how to share the medium with neighbors. A sender should not start a transmission until the other nodes within the interference range, which is centered at its receiver, are idle. From the sender to the last hop within the range, packets are forwarded in a stop-and-wait fashion; the sender can start the next transmission after a transmitted packet reaches the first outside hop.

There is the tradeoff between the transmission rate and the number of intermediate nodes within the interference range. At high transmission rates, nodes have shorter transmission ranges and it results in larger number of intermediate nodes where packets need to be forwarded one by one. Nodes at lower rates have smaller number of intermediate nodes but the link capacity is reduced.

Active neighbors that are not on the path also have an effect on the parallelism. Note that nodes are active when they have packet flows. One transmitting neighbor stops packet forwarding on links of the path within the neighbor's interference range. Having a large number of active neighbors also reduces the throughput of the path.

Thus, to select the best path with the maximum throughput, we need a new path metric instead of a simple measurement of the number of hops to a destination. In the remaining part of this section, we develop a new path metric, the expected capacity of a path and perform a simple analysis to show the relationship between transmission rates and the throughput.

A. Expected Capacity of a Path

The expected capacity here means the maximum throughput of a path if the path is selected. To compute the throughput, we assume the medium access protocol provides per-node fairness. Note that 802.11b, which is the most popular one, is implicitly able to provide only per-node fairness [17]. There are also many other scheduling protocols and algorithms for per-node fairness [10], [14]. Based on per-node fairness, a node gets time to transmit inverse-proportional to the number of active neighbors. That is, during a given time interval T, the per-node fairness scheduler guarantees for node i transmission time as much as T/N, where N is the expected number of active nodes. Note that some of intermediate nodes will become active if a flow goes over the path and we have to consider these nodes. We, thus set $N = 1 + N_i + M_i$, where N_i is the number of neighbors that will become active if this path is chosen and M_i is the number of neighbors that are active. If node i transmits at rate r_i, the maximum amount of packets transmitted for time T is $T/(1+N_i+M_i)\times C(r_i)$ and thus the maximum throughput is $C(r_i)/(1 + N_i + M_i)$. Figure 2 shows an example.

In Figure 2, inactive nodes are in white while active ones are in black. The number of inactive intermediate nodes within the carrier-sensing range of node i is one.

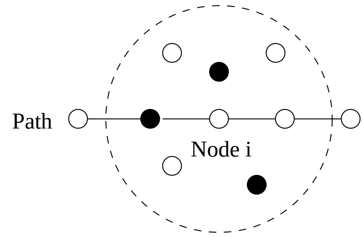

Fig. 2. Expected Capacity of Node i

The number of active nodes within the range is three. Thus, the maximum expected throughput is $C(r_i)/5$ if the transmission rate of node i is r_i.

Note that the throughput we examined above is that of a link from node i to its next hop. Let j be the next hop and C_{ij} be the capacity. Let $C_L(i,n)$ be the expected capacity of path L from node i to n. Since the maximum throughput of a path is that of the bottleneck node, C_L is given by the following:

$$\begin{aligned} C_L(i,n) &= \min_{(x,y)\in L} C_{xy} && (1)\\ &= \min\left(C_{ij}, \min_{(x,y)\in L-\{(i,j)\}} C_{xy}\right) && (2) \end{aligned}$$

Equation 2 shows that C_L can be used as a path metric. Every node sets the capacity of a path to itself to infinity and broadcasts it to neighbors. When a neighbor node h gets a message with metric $C_L(i,n)$ from node i, it computes C_{hi} and takes the minimum of C_{hi} and $C_L(i,n)$ as $C_{L'}(h,n)$, where $L' = L \cup \{(h,i)\}$. Then, it compares $C_{L'}(h,n)$ with $C_{L^*}(h,n)$, where L^* is the current optimal path to node n. If $C_{L'}(h,n) > C_{L^*}(h,n)$, node h informs its neighbors of the new path metric. Ties can be broken by the length of paths, $|L'|$ and $|L^*|$

The next subsection presents a simple analysis on the capacity of a chain with multiple rates.

B. Simple Analysis on a Chain

Using CSMA/CA protocol, a sender can avoid any interference by holding its transmission while the channel is sensed busy. Let d_{CS} be the carrier-sensing range centered in a receiver, and let $d_T(r)$ and $d_I(r)$ be the transmission and interference range at rate r. Assuming a free path loss model with the loss exponent n, $d_I(r)$ is approximately $\sqrt[n]{SNR(r)} \cdot d_T(r)$ [15]. To detect interference by carrier-sensing mechanisms, $d_{CS} = \max_r d_I(r)$. Note that we ignore here the thermal noise since it is ignorable comparing to interference signal.

Now consider a single flow in a dense network. Let r_i and M_i be a transmission rate and the number of active neighbors of the bottleneck node i on a path.

We assume all nodes are inactive with probability p. Then, the capacity of the path is $C(r_i)/(1 + N_i + M_i) = C(r_i)/(1 + p \cdot N'_i + (1 - p) \cdot M'_i)$, where N'_i is the total number of intermediate nodes within node i's carrier-sensing range and M'_i is the total number of neighbors. Note that we can not reduce M'_i. To increase the throughput of node i, it is be better to minimize N'_i. Actually, if M'_i is a constant M for any node i and p is given, there exists the optimal rate to minimize N'_i.

Let $R(r)$ be $C(r)/(1 + p \cdot N'(r) + (1 - p) \cdot M)$, where $N'(r) = \lfloor 2 \cdot d_{CS}/d_T(r) \rfloor$. Note that $N'(r)$ is the minimum number of nodes within the carrier-sensing range if all nodes on the path transmit at rate r. Let r^* be the optimal rate such that $R(r^*) = \max_r R(r)$. We will prove the rate r^* is the optimal rate for paths that are much longer than d_{CS}.

Let node i be the bottleneck node. If r_i is less than r^* and the capacity of node i is greater than $R(r^*)$, N'_i must be smaller than $N'(r_i)$ and at least one node j within the carrier-sensing range of i should have the transmission rate lower than r_i in order to have N'_i smaller. Since node i is the bottleneck, the capacity of node j needs to be larger than $R(r^*)$ and we have to find another node whose transmission rate is lower than r_j. Certainly, we can not repeat this process forever because the rate set is finite. Thus, for very long paths, it is not possible to have r_i less than r^*.

Now consider $r_i > r^*$. If r_i is greater than r^* and the capacity of node i is greater than $R(r^*)$, N'_i must be again less than $N'(r_i)$ because $R(r^*) \geq R(r_i)$. By the same reasoning, the bottlenect node i can not have smaller N'_i because of the finite rate set. Thus, the bottleneck node on a long path must have the optimal rate r^* and $R(r^*)$ is the capacity of the path.

Assuming p is one, we computed the $R(r)$ for all transmission rate r in 802.11a. The packet size is 500 bytes and RTS/CTS messages are exchanged. We used SNR values from `Qualnet` [13] and set n to 2. Figure 3 shows the results.

The figure clearly shows that shortest-path routing protocols, which select a path with the minimal number of intermediate nodes, does not perform best in a dense network. The minimum number of intermediate nodes is achieved by sending at the lowest transmission rate to the farthest node. The figure shows that the lowest rate does not accomplish the maximum throughput. Note that the maximum transmission rate in 802.11a, 54 Mbps is also not the optimal rate. That is because too many intermediate nodes within the carrier-sensing range prevents nodes from transmitting for a long time. 12 Mbps, instead, is the optimal rate r^* and achieves the maximum throughput.

If p is close to zero, $R(r)$ is roughly close to $C(r)/(1 + M)$. Since $C(r)$ is maximized when r is the highest rate, the highest transmission rate becomes the optimal rate in this case. Thus, the larger M, the bigger optimal rate we expect.

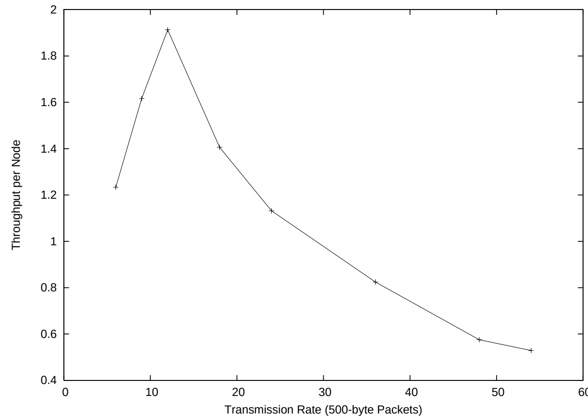

Fig. 3. Optimal Throughput in 802.11a

III. New Metric and Routing Protocol

In section II, we presented the new path metric and a prototype of routing protocol. As a shortest path metric, our path metric can be incorporated into existing distance vector or link-state protocols. The majority of existing wireless ad hoc routing protocols fall into these categories (AODV, DSR, OLSR, DSDV) and we made changes to DSDV protocol [12] from the Rice/CMU implementation in the `ns` simulator [11], [9].

DSDV is a distance-vector protocol propagating paths with their metrics, which are the number of hops to destinations. We made changes to the original DSDV design in order to ensure that it uses the path with the metric, the expected capacity. Note that our new metric consists of two parts: the capacity and the number of hops. A path with the minimum number of hops will be selected if all alternative paths have the same capacity. Problems involved in implementation are listed in following subsections.

A. The Expected Capacity

The key metric in our protocol is the expected capacity. These path capacities are not usually integers and passing real numbers may involve the compatibility problem. Instead, we used a pair of $(r_i, (1 + N_i + M_i))$, where i is the bottleneck node; nodes locally compute the expected capacity from the pair and the metric formula, $C(r_i)/(1 + N_i + M_i)$. Since the transmission rate r_i may be a real number, we provided index numbers to all transmission rates, which are integers. Thus, our metric information now has three numbers: a transmission rate index, the expected number of active nodes and the number of hops to the destination.

B. The Number of Active Neighbors

In our protocol, every node has to know how many neighbors are active. Several researches have done to address this problem. [4] proves that the expected number of collisions in a binary backoff algorithm grows asymptotically with $O(\log M)$, where M is the number of nodes. From the results in [4], [6] shows that how to estimate the number of wireless nodes by keeping track of the number of collisions.

If nodes are on and off frequently, a network will be flooded with routing update messages. It may take some time to be stabiized again. One simple approach is to assume a certain portion of neighbors are always active on average, which eliminates dynamic measuring of active neighbors. Routing by the number always keep stable and is not affected by changes of the other paths. The extreme end of the approach is to assume that all neighbors are active, in which case a node achieves the worst throughput. Our routing protocol then attempts to identify a path to maximize the worst-case throughput.

C. The Number of Neighbors to be Active

The Number of Neighbors to be Active is how many inactive neighbors will be turned active if a path is chosen and a flow goes over it. Note that in DSDV protocol, a node only knows the expected capacity of a path that starts from the node. Figure 4 shows an example.

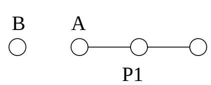

Fig. 4. Inactive Node Problem

In Figure 4, node A advertises the expected capacity of path $P1$. Assume node B is inactive. If the bottleneck node of the path is A itself, node B can not rely on the advertised capacity because the capacity will decrease after node B becomes active. Node A, however, can not count B because node B is currently inactive.

We propose one solution to this problem. To measure the number, nodes first find out their neighbors by exchanging ‘Hello’ messages. Nodes also advertise the bottleneck node of a path as well as the path metric. If a node gets an advertisement of an path and the bottleneck node is one of its neighbors, it increases the expected number of active nodes by one and computes the expected capacity, which is the correct one of the path starting from the node. If the path is the best, then the node advertises the adjusted number of active nodes.

If a network has a large number of traffic flows, the number of neighbors to be active will be close to zero. Thus, one approach is to simply assume that the number is zero. In section IV, we measure the performance of this approach.

D. Multiple Flows on a Node

Since the medium access protocol provides per-node fairness, multiple flows on a node must share the bandwidth allocated to the node. Fair-sharing scheduler at the node will assign each flow the same amount of the bandwidth. To measure the capacity of a path, a node compares the capacity that could be assigned to a flow with the advertised capacity; the node take the minimum as a path metric and broadcasts if needed.

Here, we again have to cope with the dynamics of flows. As mentioned above, it may cause message flooding and have a network fluctuate. Thus, we simply use the capacity of a node instead of that of a flow to compute the expected capacity of a path. We discuss this in detail in section IV.

IV. Simulation

We performed simulations with `ns` simulator. Nodes were randomly placed in 1000 $\times$ 1000 m^2 area. Each node runs 802.11b protocol, which support 4 different transmission rates. 20 pairs of source and sink nodes were selected at random and CBR applications on sources continuously supplied 1000-byte packets to 802.11b through UDP layer.

Four routing metrics were evaluated in simulations: the original shortest-path metric (ORG), the capacity metric with an assumption that all neighbors are active (WORST), the capacity metric with the actual number of active neighbors (ACT) and the greedy metric (HI), which we explain below. Note that the metric ACT does not count inactive neighbors that would be active if a path is selected, as mentioned in the previous section.

The greedy metric always chooses the highest rate links. That is, a node selects a path where it can reach the next hop at the highest transmission rate. Ties are broken by the length of paths. Note that in section II, we showed that the highest rate is the optimal rate under heavy traffic. Traffic of 20 flows in a network with 100 nodes is heavy and thus we compare the throughput of the greedy metric with ours.

Figure 5 shows the total system throughput ratio for 20 flows. 50 independent simulation runs were performed, and the results are plotted in ascending order of throughput ratio.As can be observed, our metric WORST achieved better throughput in most cases. The ratio of the throughput of ours to that of the original metric was more than 1.2 on average.

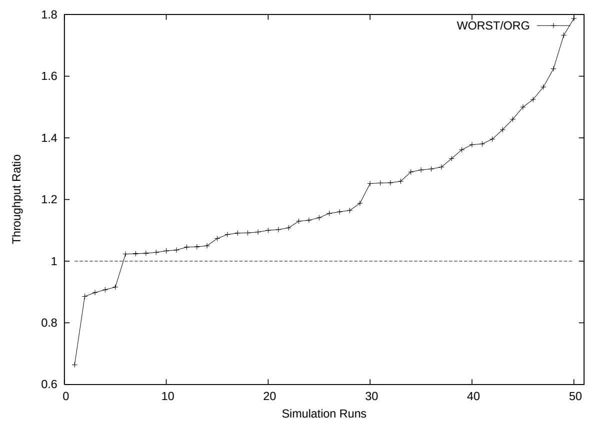

Fig. 5. Throughput Improvement in 802.11b

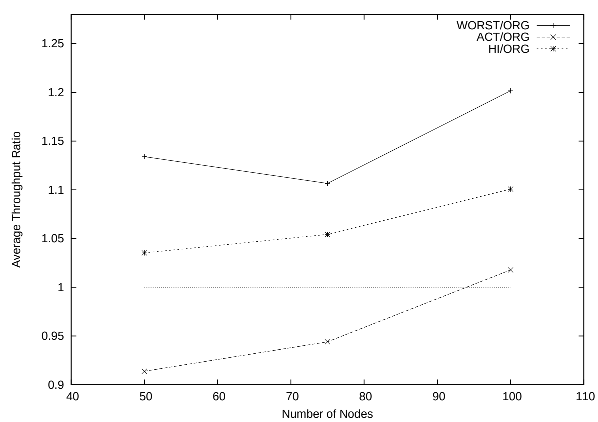

Fig. 6. Throughput Improvement Varying the Number of Nodes

Figure 6 plots the system throughput varying the number of nodes in range of 50 to 100. System throughputs are normalized with respect to one of the original metric. The figure shows that even under heavy traffic, the metric WORST still improves the system throughput more than the metric HI.

The metric ACT performed even poorer than the original metric. The reason is that since we do not count the number of flows on a node, flows tend to go over the same best links. Note that the metric WORST does not count the flow number either; however, the metric usually tries to scatter flows over an area because the closer nodes are to borders of the area, the fewer neighbors they have.

The metric WORST may also have the same problem in some topologies. The simulation results clearly indicate that routing metrics considering the expected capacity must address the number of flows on a hop.

V. Related Work

The traditional technique used by most existing ad hoc routing protocols is to select minimum hop paths. These paths tend to contain long range links that have low effective throughput.

Awerbuch et al. [1] present a metric to help find high-throughput paths when different links can run at different bit-rates. They propose the Medium Time Metric that assigns a weight to each link in the network that is proportional to the amount of medium time used by sending a packet on that link. Note that their medium time includes only transmission time and does not consider any waiting time due to neighbor activities. Their simulations show that the MTM yields an average total network throughput increase of more than 200% over the traditional hop count metric. The throughput gains, however, are achieved only for paths longer than 3200 m in their simulations. For 1600-m paths, their routing metric performs poorer than the traditional metric. Our routing protocol shows better performance even in an 1000×1000 m^2 area.

Some techniques [16], [7], [8] explicitly schedule transmission slots in time or frequency division MAC layers to provide bandwidth guarantees. They assume TDMA or FDMA and try to find out the number of available slots or frequencies. CSMA/CA protocols do not need such explicit schedules. We instead proposed the expected capacity of a path.

Several researches have done on routing protocols over lossy links. De Couto et al. [2] presents the expected transmission count metric (ETX), which finds high-throughput paths on multi-hop wireless networks. ETX minimizes the expected total number of packet transmissions (including retransmissions) required to successfully deliver a packet to the ultimate destination. They focus on very sparse networks where link error probability is high and does not consider waiting time, which counts for the throughput in dense networks. Draves et al. [3] extended De Couto's work and present the Expected Transmission Time (ETT) that is a function of the loss rate and the bandwidth of the link. They, however, do not count waiting time caused by neighbor activities for the link bandwidth.

VI. Conclusion

In this paper, we first introduce the tradeoff in dense networks between the transmission rate and the number of intermediate nodes within the interference range. We present a new routing metric, the expected capacity to find out the best path with the maximum possible throughput. Simple analysis on the path throughput shows that there exists the optimal transmission rate for the bottleneck node and the traditional shortest-path metric does not achieve the maximum throughput in dense networks.

We also mention several implementation issues to use our new metric in existing routing protocols such as DSDV. One variation of our routing metric is presented to achieve stability of routes and find a path whose expected throughput in the worst case is the best. Simulation results show that the metric performs better than the traditional one by 20%.

Several aspects of our metric could be improved in the future. Highly adaptive measurement of active and to-be-active neighbors is required. Handling multiple flows on a node is also one of problems to be addressed.

REFERENCES

[1] B. Awerbuch, D. Holmer, and H. Rubens. High throughput route selection in multi-rate ad hoc wireless networks. Technical report, Johns Hopkins University, Computer Science Department, 2003.

[2] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. In *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 134–146. ACM Press, 2003.

[3] Richard Draves, Jitendra Padhye, and Brian Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 114–128. ACM Press, 2004.

[4] Jonathan Goodman, Albert G. Greenberg, Neal Madras, and Peter March. Stability of binary exponential backoff. *J. ACM*, 35(3):579–602, 1988.

[5] Approved draft 802.11-1999 (revisions to IEEE std 802.11-1997). IEEE Standard, 1999.

[6] Hwangnam Kim and Jennifer C. Hou. Improving protocol capacity with model-based frame scheduling in IEEE 802.11-operated WLANs. In *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 190–204. ACM Press, 2003.

[7] Chunhung Richard Lin. An on-demand QoS routing protocol for mobile ad hoc networks. In *IEEE INFOCOM*, pages 1735–1744, 2001.

[8] Anastassios Michail and Anthony Ephremides. Algorithms for routing session traffic in wireless ad-hoc networks with energy and bandwidth limitations. In *Proceedings of 12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2001.

[9] Rice Monarch Project. Wireless and mobility extensions to ns-2.

[10] Thyagarajan Nandagopal, Tae-Eun Kim, Xia Gao, and Vaduvur Bharghavan. Achieving MAC layer fairness in wireless packet networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 87–98. ACM Press, 2000.

[11] The Network Simulator – ns-2, 2004.

[12] Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *Proceedings of the conference on Communications architectures, protocols and applications*, pages 234–244. ACM Press, 1994.

[13] Scalable Network Technologies. Introduction to QualNet. Qualnet User's Manual, 2004.

[14] Nitin H. Vaidya, Paramvir Bahl, and Seema Gupta. Distributed fair scheduling in a wireless lan. In *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 167–178. ACM Press, 2000.

[15] Kaixin Xu, Mario Gerla, and Sang Bae. How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks? In *IEEE Globecom*, 2002.

[16] C. Zhu and M. Corson. QoS routing for mobile ad hoc networks. In *IEEE INFOCOM*, 2001.

[17] Hua Zhu, Ming Li, I. Chlamtac, and B. Prabhakaran. A survey of quality of service in IEEE 802.11 networks. *Wireless Communications*, 11(4):6–14, 2004.