*UCLA Computer Science Department Technical Report CSD-TR No. 050005*

# AdHoc Probe: Path Capacity Probing in Wireless Ad Hoc Networks

Ling-Jyh Chen, Tony Sun, Guang Yang, M. Y. Sanadidi, and Mario Gerla
*Department of Computer Science, UCLA*
*{cclljj, tonysun, yangg, medy, gerla}@CS.UCLA.EDU*

***Abstract* – Knowledge of end-to-end path capacity is useful for video/audio stream adaptation, rate control and overlay design. Capacity estimation in wired and last-hop wireless networks has been extensively investigated, but an in-depth study of path capacity in ad hoc, multihop wireless networks is still lacking. Since the "capacity" of a wireless link can vary dynamically and rapidly due to changes in interference, mobility or energy optimization policy, timely path capacity tracking is the key to efficient routing, traffic management and application deployment. In this paper, we present AdHoc Probe, a "packet-pair" based technique, to estimate end-to-end path capacity in ad hoc wireless networks. Using analysis and simulation, we show that AdHoc Probe converges fast and thus works well in mobile, rapidly changing scenarios. AdHoc Probe is simpler, faster and less intrusive than previously proposed schemes.**

## 1 Introduction

With the increasing deployment of wireless devices (e.g. laptops, PDAs, cell phones, etc), ad hoc networking is becoming an increasingly important strategy for interconnecting wireless devices. Ad hoc wireless protocols have been extensively investigated at all layers from physical to applications. However, a systematic development of ad hoc wireless network tools is still lacking. In particular, there are no efficient tools to probe ad hoc network resources (e.g. path capacity, available bandwidth, etc).

Yet, timely information about available resources such as path capacity is important for network and applications management. Consider for example a video and multimedia call from a fixed customer in the Internet to a PDA mobile user. Suppose the mobile user leaves the Starbuck coffee shop (where he enjoyed good WLAN connectivity) and ventures in the park, where only "opportunistic" multihop connectivity is available. The rate may have dropped by more than one order of magnitude, from Mbps to less than 100 Kbps. If the Internet caller can rapidly detect the change in path rate (using an end to end capacity probing tool), it can switch from full motion video to fixed frame video or to audio only. Without path capacity detection, the source may struggle to find the proper “content” format to deliver. As another example, in a video conference application supported by an “overlay” that spans wired and wireless ad hoc users, the knowledge of path capacities to different destinations can help the “virtual” design algorithm construct more efficient topologies. Moreover, the media sources and proxies can adapt the audio/video streaming rates to match user capabilities and provide better quality of service. A third example is an application broadcasting a live sport event over the Internet. The server delivers a soccer game, say, to Internet subscribers, many of them mobile, with PDA handheld devices. Capacity probing helps the source determine how many types of “formats” it should broadcast for that event and in what regions/directions.

The end-to-end path capacity estimation tool must be simple and accurate. The estimation process must be fast so that it can reflect the path capacity in a timely manner, even when the actual capacity is varying. The estimation must be independent of cross traffic (in this case we are interested in evaluating the equivalent of the “bottleneck” capacity in the Internet, as opposed to the “available” bandwidth). The estimation tool must be applicable to mixed wired and wireless paths, since several applications (especially commercial applications) will include “opportunistic” ad hoc wireless extensions as to the Internet. Finally, the estimation must be non-intrusive so that it does not disturb the ongoing applications traffic in the network.

The path capacity estimation problem has been extensively studied in the Internet. Various tools have been proposed to accurately estimate bottleneck link capacity in the wired and/or last-hop wireless networks [2][3][4][5]. However, the complexity and convergence time required by these schemes make them unsuitable for ad hoc wireless networks. Moreover, the bidirectional setup of some of the techniques proves to be detrimental in ad hoc networks as discussed later.

There also exist several techniques for ad hoc path capacity estimation, which are supported by specific routing algorithms. For example, on demand routing algorithms (e.g. AODV [1]) as well as proactive algorithms (e.g. LANMAR and Fisheye Routing [7]) have been successfully extended to yield path capacity. These estimations however are dependent on specific routing schemes. Most importantly, they require feedback from the network layer. Our goal is to develop an *end to end only* estimation technique that is independent of the network layer. This is critical in overlay applications, for example, where the source across the Internet does not know which routing scheme runs in the remote adhoc network being probed.

Li et al directly addressed the end to end estimation of path capacity of a static multihop network based on 802.11b. Their approach was to send a brute force UDP packet stream and measure the maximum achievable data throughput [6]. This scheme gives realistic results in that it reflects the currently available capacity if a UDP stream is injected. However, as discussed later it is not very practical due to latency and to impact on ongoing traffic. Moreover, the capacity measurement is affected by volatile cross traffic conditions.

End-to-end path capacity estimation in ad hoc wireless networks is much more challenging than in wired nets. Recall that the capacity estimate must be independent of cross traffic, since the capacity of each link by definition is independent of any other traffic present in the neighborhood. However, in an ad hoc network, the estimate depends not only on the "narrow" link along the path (as in a wired network). It must also account on "self interference" among packets in the same session that are a few hops away; on the topology, path layout, interference between nodes along the path, and several other environmental parameters. Motivated by the above goals, in this study we propose AdHoc Probe, a packet-pair based end to end technique that estimates path capacity in ad hoc wireless networks. We evaluated AdHoc Probe in static and mobile networks with simulation experiments. In all cases, we show that AdHoc Probe can estimate path capacity rapidly and correctly.

The rest of the paper is organized as follows. In section 2, we summarize related work and recap the Internet CapProbe technique on which AdHoc Probe is based. An in-depth description of AdHoc Probe follows in Section 3. In section 4, we present an analysis of the path capacity in various adhoc wireless network configurations. Next, using NS-2 simulation, we validate AdHoc Probe in configurations including multi-hopping, relocation and mobility, in section 5. We conclude the paper in section 6.

## 2 Related Work and Background

### *2.1 Related Work*

Link capacity estimation has been extensively studied in wired networks, e.g. [2][3][4][5]. Early research on capacity estimation relied on either delay variations among probe packets [3] or dispersion among probe packets[2][5]. Dovrolis revealed in [2] that the dispersion distributions can indeed be multi-modal without multi-channels, and that the strongest mode in the multimodal distribution of the dispersion may correspond to either (1) the capacity of the path, or (2) a "compressed" dispersion, resulting in capacity over-estimation, or (3) the Average Dispersion Rate (ADR), which is lower than the capacity. Kapoor et al proposed a packet-pair based approach called CapProbe in [4]. CapProbe combines delay and dispersion measurements to estimate the bottleneck link capacity fast and accurately.

However, all of the above schemes have been and evaluated in wired and last-hop wireless scenarios. They have never been tested in ad hoc wireless networks. Capacity estimation in ad hoc wireless networks remains challenging due to the rapidly varying channel conditions, presence of node mobility, and multiple hops of wireless links.

In [6], Li et al examined the interaction of 802.11 MAC and ad hoc forwarding and the ability to infer path capacity for several simple configurations and traffic patterns. A brutal force approach (i.e. using UDP packets to probe the maximum throughputs of the network) was used in simulation and experiments to validate the hypothetical limits of utilization in a chain network (1/4 of effective capacity). However, this approach was only able to obtain a substantially lower utilization of 1/7 of effective capacity.

In addition, as we mentioned earlier, Li's approach is intrusive and the result approximates path capacity only if the network is idle and static.

### *2.2 CapProbe*

CapProbe [4] is a recently proposed bottleneck capacity and path rate estimation technique shown to be both fast and accurate over a large range of scenarios. When two back-to-back packets are launched into a network, they are always separated at the bottleneck link by an interval related to bottleneck capacity. If such interval is preserved until destination unperturbed, it will allow us to compute the bottleneck link capacity (as shown in Figure 1). The bottleneck capacity is equal to the UDP stream rate that can be sustained by the path – hence the name of "path rate" estimator often given to these tools. The interval between "interfered" packet-pair samples might be either expanded or compressed, where "expansion" leads to under-estimation and "compression" leads to over-estimation of the capacity.

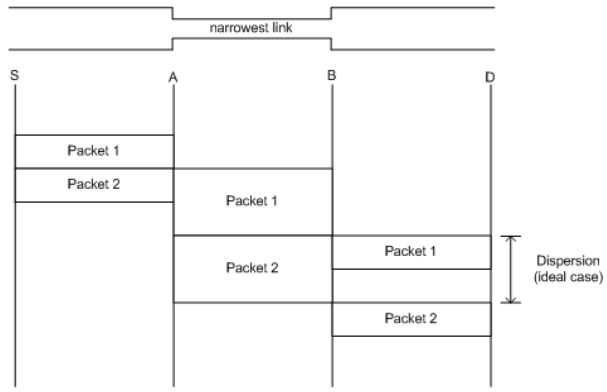


**Figure 1: Packet-pair technique: the ideal case.**

CapProbe combines the use of time interval measurements and end-to-end delay measurements to filter out packet pair samples interfered by cross traffic. By construction, an incorrect value of capacity estimate can occur only if cross traffic has interfered with the packet pair. In this case, queueing ensues and the delay of one or both packets will be larger than the minimum observed in absence of cross traffic. The sum of the delays of the packets in the packet pair is defined as *delay sum*. A delay sum, which does not include any cross-traffic induced queuing delay, is referred to as the *minimum delay sum*. Any sample with delay sum larger than the observed minimum is thus discarded as it must have been interfered with. The capacity is easily derived from the equation:

$$C = P / T \qquad (1)$$

where $P$ is the sampling packet size, and $T$ is the interval between packets with minimum delay sum.

CapProbe has been validated in wired and last-hop wireless networks with a variety of configurations. However, it has not been tested in ad hoc wireless networks yet. In wired networks, CapProbe is generally used in the round-trip mode, to evaluate the minimum capacity over the two directions. In ad hoc wireless networks, the round-trip mode is inadequate, since the first packet once relayed by the receiver may collide with the incoming second packet. Based on the CapProbe concept, we thus design a one-way technique, called AdHoc Probe, to estimate the unidirectional path capacity (or more properly, rate) in ad hoc wireless networks. We will present the algorithm and evaluations in the following sections.

## 3 AdHoc Probe

AdHoc Probe is based on CapProbe, a well-proved bottleneck link capacity estimation tool for wired and last-hop wireless networks [4]. However, AdHoc Probe differs from CapProbe in many significant ways. First, AdHoc Probe is a one-way (instead of round-trip) estimation technique. Secondly, AdHoc probe measures the maximum rate achievable on an "unloaded" path (i.e. no other users present) when intermittent environmental problems (e.g. short range mobility, random errors, etc) are factored out. It turns out that the achievable rate is generally considerably less than the minimum link capacity (while the two values are identical on a wired path). Thirdly, AdHoc Probe is designed to work under conditions not present in a typical Internet path, for instance when the wireless network is mobile, multihopped, interfered, and subject to rapid changes in link data rates.

In the following subsections, we first present the AdHoc Probe algorithms in 3.1, and then discuss the system time synchronization issue in 3.2.

### *3.1 AdHoc Probe Algorithms*

Similar to CapProbe, AdHoc Probe relies on the packet-pair technique to provide capacity estimation in wireless networks. However, while CapProbe is designed to estimate the *bottleneck link* capacity in a round-trip fashion, AdHoc Probe intends to estimate the *end-to-end path* rate based on one-way measurements. The end-to-end path rate is the *maximum achievable rate* over the wireless path in the absence of any competing traffic. Such metric can be used in end to end applications with QoS requirements, overlay designs, network traffic management, etc. The maximum *achievable rate* (in the absence of competing traffic) is typically lower than the nominal channel transmission rate due to a variety of reasons including multihopping, and unique features of wireless networks (e.g. RTS/CTS mechanism), etc. AdHoc Probe is able to accurately measure such achievable rate.

The basic AdHoc Probe algorithm can be obtained by modifying the round-trip CapProbe into a one-way mechanism. Probing packet pairs of fixed size are sent back-to-back from the sender to the receiver. The sending time is stamped on every packet by the sender, the one way delay (OWD) is calculated at the receiver, and the path capacity (i.e. rate) estimation is performed at the receiver and communicated to the sender.

The receiver measures the OWD of each packet as the difference between time received (clocked at the receiver) and time sent (stamped in the packet header) The OWD sum is then computed. The "good" packet-pair samples (i.e. the packet pairs encountering no cross traffic) are those with minimum OWD sum (as shown in Figure 2), and the corresponding capacity is given by $C=P/T$, where $P$ is the packet size and $T$ is the dispersion of the packet pair.

AdHoc Probe does not implement the "convergence test" (as CapProbe does), in order to make the algorithm simple, fast, and timely to the highly varying characteristics of wireless networks. Instead, AdHoc Probe simply reports the capacity estimation after receiving a number of samples, say $N$. Similar to CapProbe, the accuracy of capacity estimation increases as $N$ grows. However, a large $N$ value is not suitable for mobile wireless networks as it will lead to high latency in estimation and may not allow us to capture the dynamic properties of the wireless network.

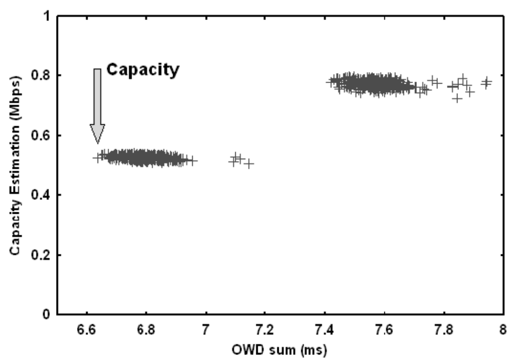


**Figure 2: AdHoc Probe capacity estimate using the sample with minimum OWD sum.**

Apart from the number of samples, $N$, the latency of the estimate also depends on the probing packets sending rate (i.e. the probing rate). For simplicity, AdHoc Probe simply sends probing packets (with the packet size of $P$ bytes) using a CBR rate (i.e. $R$ packet-pair/second, or equivalently $2*P*R$ bytes/second). As a result, the expected duration of one estimation is approximately $N/R$ seconds. Clearly, the larger $R$ is, the less time a capacity estimation process takes. However, $R$ should be upper-bounded since a large $R$ may disturb the ongoing foreground traffic in the network or even congest the network. As a result, the capacity estimate may become inaccurate (hard to get one good sample) or extremely slow (packets are lost due to congestion).

The probing parameters $N$ and $R$ need to be carefully tuned in accordance with the underlying network properties and by trading off precision for speed. This tradeoff clearly depends on the application. In this paper, we simply set $N$ = 200, $P$= 1500, and $R$ = 4 sample pairs/sec for all simulations and testbed experiments.

*3.2 System Time Synchronization Issue*

The OWD measurement in AdHoc Probe is problematic on a real testbed. Unlike the perfect time synchronization provided by the simulator, the system clocks of the two end hosts are usually not synchronized. As a result, the measured OWD will not be identical to the actual OWD. Though some software packages and service protocols have been proposed to enable time synchronization of network hosts, one can not guarantee the two end hosts are always synchronized before the estimation. Thus, a successful capacity estimation technique should not rely on any assumptions of a perfectly time-synchronized system. We now provide simple analysis on the AdHoc Probe algorithm and show that AdHoc Probe works well even when the system time is not perfectly synchronized.

Suppose $\delta$ is the constant time offset between the AdHoc Probe sender and receiver. For the $i$-th packet pair sample, the sending time is stamped $T_{send,i}$, and the receiving times (on the receiver) are $T_{recv1,i}$ for the first packet and $T_{recv2,i}$ for the second packet, respectively. Therefore, the measured OWD sum ($S'$) and the actual OWD sum ($S$) of the $i$-th packet pair sample are:

$$S_i' = (T_{recv1,i} - T_{send,i}) + (T_{recv2,i} - T_{send,i}) \quad (2)$$

$$\begin{aligned} S_i &= (T_{recv1,i} - T_{send,i} - \delta) + (T_{recv2,i} - T_{send,i} - \delta) \\ &= S_i' - 2\delta \end{aligned} \quad (3)$$

Thus the difference between $S_i$ *and* $S_i'$ is a constant $2\delta$ for all packet pairs. If $S_k = \min_{i=1..n}(S_i)$ for $k \in [1..n]$, then $S_k' = \min_{i=i..n}(S_i')$, and vice versa. By filtering out those samples of non-minimum $S'$, it is easy to identify the "good sample" that has the minimum $S'$ and $S$, and the capacity estimation is computed by using the interval between this packet pair sample. Clearly, the interval is not affected by the time offset. Therefore, AdHoc Probe is able to absorb the constant time offset $\delta$ between the sender and the receiver and produce an accurate capacity estimate.

## 4 What does AdHoc Probe actually measure?

A major difference between CapProbe and AdHoc Probe is the interpretation of the result. While in a wired network the path rate is equal to the bottleneck capacity, the path rate of a wireless multihop path is related to, but not necessarily identical, to the minimum of the link rates along the path. In the sequel we first review existing models that predict the effective rate and then show that AdHoc Probe indeed estimates such rate.

We recall that the effective end-to-end rate is defined as the maximum achievable data rate in the absence of any cross traffic connection. As mentioned earlier, this is smaller than the raw data rate at the physical layer. The difference is due to packet O/H and channel access coordination to handle multiple, pipelined packets on the path. We derive specific models below.

More specifically, in the 802.11b standard, since a RTS packet is 40 bytes, CTS and ACK packets are 39 bytes, and the MAC header of a data packet is 47 bytes, the effective capacity of a one-hop wireless link can be calculated by using the following equation[1]:

$$C = \frac{S}{S + 40 + 39 + 47} \times C_P \qquad (4)$$

where $S$ is size of the network layer packet (including IP header), and $C_P$ is the link capacity of the physical layer. For instance, when the data packet size is 1500 bytes and the data rate of the wireless link is 2Mbps, the effective capacity is at most $\frac{1500}{1500+40+39+47} \times 2 \approx 1.8 Mbps$.

However, due to the collision avoidance mechanism provided by RTS/CTS exchanges, the effective capacity of the wireless link decreases when there is more than one node within the same collision domain. This is clearly our case where the two packets are attempting transmissions within the same connection and path. It will be true also when a UDP stream will be maintained on the path. Suppose for instance that on the path in question there are *N-1* active nodes within node A's transmission range, all engaged in forwarding packets on the same path. The maximum effective rate for that path is *C/N* since only one of the *N* nodes can transmit a packet at a time. Naturally, it is unusual to have an ad hoc network path that hops several times within the same collision domain. However, this would clearly cause a reduction in effective rate. Such rate reduction must be captured by AdHoc Probe.

Much more common is the reduction in capacity incurred when the path is multihop. We consider a simple chain topology as shown in Figure 3. For simplicity, we suppose the nodes are placed on a line with 200 meters to its immediate neighbor node, and the effective transmission range of each node is 250 meters. When the radio interfering range is the same as the transmission range, previous study by Li et al [6] has shown that the effective capacity of a chain topology becomes just 1/3 of the effective capacity of a single cell connection.

[1] We assume physical layer overhead, such as preamble, header and idle time (IFS), are negligible.

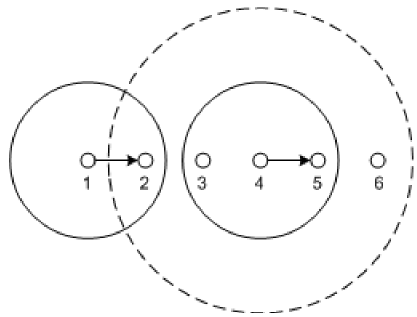


**Figure 3: The chain topology, where the solid-line circle denotes a node's effective transmission range and the dotted-line circle denotes a node's interference range.**

Moreover, as identified in [8], the radio interfering range is usually much larger than the transmission range. Therefore, the effective end-to-end capacity of a chain configuration will further decrease. For instance, in Figure 3, if the interference range (marked by a dotted-line circle) is 550 meters, a packet transmission from node 4 will interfere with a packet transmission from node 1 to 2. In other words, simultaneous data transmission is not possible among nodes 1, 2, 3, and 4. It turns out that the effective end-to-end capacity of the chain topology in Figure 3 will be at most 1/4 of the effective capacity of a single cell topology.

Another issue in a multihop path is that data rates may be different hop by hop due to different environment conditions. Thus, the link rate is no longer uniform along the path. In this case, the effective rate can still be computed with the above models, using the minimum rate link along the path.

From the above we can conclude that the effective rate in an ad hoc wireless path is more complex to model than in the wired network counterpart. One important feature of AdHoc Probe is that it does measure the correct path rate no matter how complex the underlying channel model (a physical system) is.

In the following sections, we will validate AdHoc Probe by comparing the path capacity estimation with the analytical results using simulation. We will also study the path capacity of wireless networks in various mobile wireless scenarios.

## 5 Simulation Results

This section presents simulation results illustrating the used of AdHoc Probe in estimating the end-to-end path capacities in a number of wireless network configurations.

After Li's simulation configurations in [6] , we use the *ns-2* [9] simulator with CMU wireless extensions. The wireless channel is tuned to imitate the Lucent WaveLan card at 2Mbps, with the effective transmission range set at 250 meters and the interference range at 550 meters. Nodes remain stationary during the simulations, and all simulated data packets are preceded by an RTS/CTS exchange, in accordance with the 802.11 standards. Adhoc probe is implemented in *ns-2* and used to estimate the end-to-end path capacity in various wireless network configurations.

*5.1 Distinguishing One-way and Round-trip Techniques*

We first show the difference of one-way AdHoc Probe and round-trip CapProbe techniques by evaluating them on a simple one-hop wireless link (source and destination separated by 200 meters with no external interference). Capacity estimation results from both one-way and round-trip techniques are shown in Figure 4.

The results are as expected. As earlier explained in our throughput model, the round-trip estimates are always about half of the one-way estimates in the one-hop wireless link scenario. Wireless contention and backoff resulting from packet collisions (between the $2^{nd}$ packet of the packet pair and the acknowledgement of the $1^{st}$ packet) is the main reason why round-trip CapProbe consistently measures a lower end-to-end capacity. One intuitive way to see this is that the path capacity is shared by the two directions of the probing flow, and thus it is halved. Since in our applications we are interested in the "one direction" capacity, we will restrict ourselves to AdHoc Probe in the remainder of the paper.

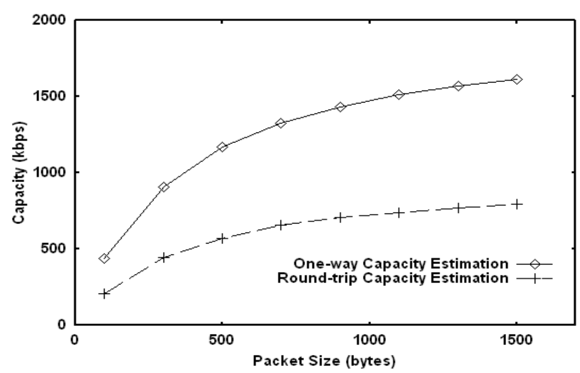


**Figure 4: Capacity estimation results of a wireless link (with no interference from other nodes) using one-way and round-trip CapProbe.**

*5.2 Path Capacity on Chain Topologies*

This subsection studies the capacity on a single chain topology, where packets originate from the first node and are forwarded to the last node on the chain. Forwarding nodes are expected to contend and interfere with their neighbors, meaning that the effective path capacity will be adversely affected.

Here, we use the same scenario as shown in Figure 3. The transmission range (marked by solid-line circle) of an 802.11 node is 250 meters, the interference range (marked by a dotted-line circle) is 550 meters, and the nodes are placed on a straight line with 200 meters in between. We have run a set of AdHoc Probe simulations on chain topologies of various packet sizes and path lengths; the results are shown in Figure 5. As expected, the estimate value increases as the packet size increases, consistent with the analytical results from Eq. 4.

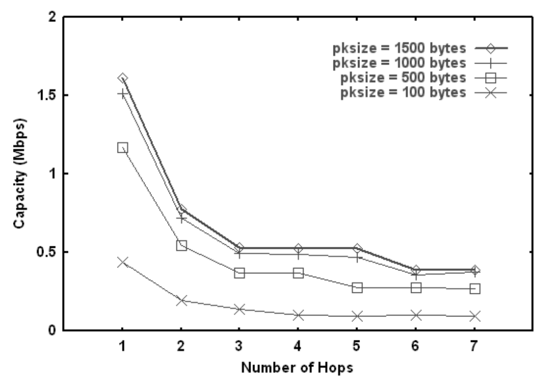


**Figure 5: Capacity estimation along a chain of nodes with different chain lengths and probing packet sizes.**

Moreover, the effective end-to-end capacity decreases as the chain grows longer, demonstrating an inverse relationship between the two variables. When the chain length exceeds four, at the packet size of 1500 bytes, the estimated end-to-end capacity converges to 400~500 Kbps. It is approximately ¼ of the single cell capacity, close to the analytical results of $\frac{1500}{1500+40+39+47}\times 2\times\frac{1}{4}\approx 450Kbps$ as discussed earlier.

With the same wireless network configuration as specified in [6], AdHoc Probe was able to achieve end-to-end capacity estimation that closely matches the analytical prediction (1/4 of single hop capacity). A previous empirical method for evaluating wireless end to end capacity [6] converged to a lower estimate of single cell capacity (~250 Kbps) using the same channel and packet size assumptions employed in our simulation.

That method measured the path rate by pushing a UDP stream and evaluating the achieved throughput. AdHoc Probe attains a path capacity estimate that is more consistent with analytical results as compared with [6], with considerably less intrusion. In general, a stream or flow based testing strategy like the one reported in [6] can be more appealing as it gives a measure of real achieved throughput (i.e. what you measure is what you would actually get at this instant with a UDP stream). However, the technique can be too disruptive for some applications. Moreover, the estimate is more limited as it depends on the type of stream used (e.g. the UDP experiment cannot be directly used to predict TCP throughput). If there is other traffic in the network, the probing stream will interact with it in a way that is very difficult to analyze; it will be difficult then to extract the "idle path" rate estimate that we are set out to discover. AdHoc Probe in contrast is much less intrusive. It only needs one "good" sample to correctly estimate the path capacity no matter what the cross traffic load of interference is. Consequently, the path capacity so estimated is of more general use as it can be employed to model and predict the throughput of different types of streams (e.g. UDP, TCP, etc) in different loading conditions.

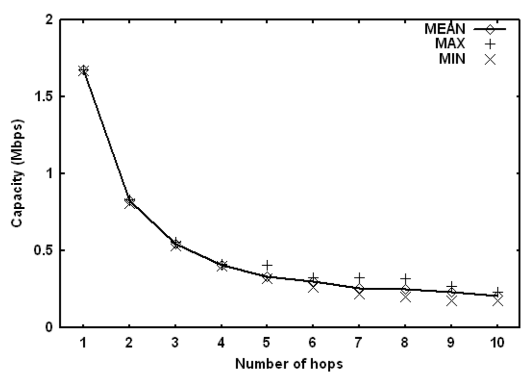


**Figure 6: Capacity estimation of wireless multihop connections within the same collision domain.**

*5.3 Path capacity within the same interfering domain*

Next, we evaluate the capacity of a highly interfered wireless path. More precisely, we wish to validate the *C/N* relationship derived from the model in section 4. To this end, we have designed a simulation experiment where the hops of the multihop path are all in the same collision domain. The topology and configurations used here are the same as in section 5.2, except that the distance between a node and its next-hop neighbor is reduced to 10 meters here. We have run AdHoc Probe using the packet size of 1500 bytes with various numbers of hops. Figure 6 shows the average estimation results of 20 runs, as well as maximum and minimum estimates, at each number of hops. As predicted by the model, the end-to-end capacity estimate decreases as the inverse of the number of interfering nodes (or equivalently, the number of hops in the same collision domain).

*5.4 Path Capacity Estimation on Grid Topologies*

Since grid topologies are more representative of ad hoc configurations than chain topologies, we now consider the *n* x *n* regular grid shown in Figure 7. Nodes are placed 200 meters away from their horizontal and vertical neighbors. Radio transmission range is set to 250 meters, and the radio interfering range is set to 550 meters. We consider two types of traffic patterns: horizontal flows only (Figure 7a), and horizontal plus vertical flows (Figure 7b). Path capacity is measured for the mid horizontal row via AdHoc Probe; other paths carry flows with Poisson distribution at an average rate that varies up to 100Kbps. Figure 8 and Figure 9 show the respective path capacity estimates.

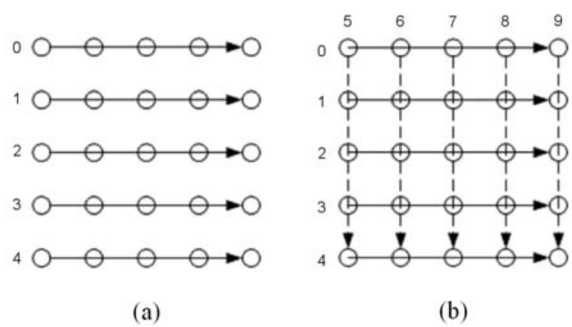


**Figure 7: A regular 5x5 Grid topology. (a) left-to-right horizontal flows; (b) both left-to-right horizontal and top-down vertical flows.**

As shown in Figure 8 and Figure 9, AdHoc Probe gives the correct estimate for all configurations. For example, in a *4* x *4* regular grid topology, the path length is 4 and (as shown in fig 6) capacity is 520Kbps. For the 7x7 grid, path length is 7 and capacity is 400Kbps. The estimates become incorrect (by defect) when the grid becomes totally saturated with cross traffic. For example, consider the 4x4 grid with both horizontal and vertical cross traffic. Because of the transmission and interfere ranges, only one packet at a time, vertical or horizontal can go through the grid with perfect scheduling. For 1,500B packet size, this translates into an upper bound on the maximum 4x4 grid capacity of 60 Kbps per flow. Figure 9 shows that below saturation the 4x4 capacity estimate is accurate. Around saturation, at 60Kbps, the estimate is not correct by a small “defect”. In this situation, the grid never offers an “idle window” through which AdHoc Probe pairs can sneak through and provide minimum sum estimates. All pairs tend to be separated by an extra amount due to intervening traffic. This leads to underestimates.

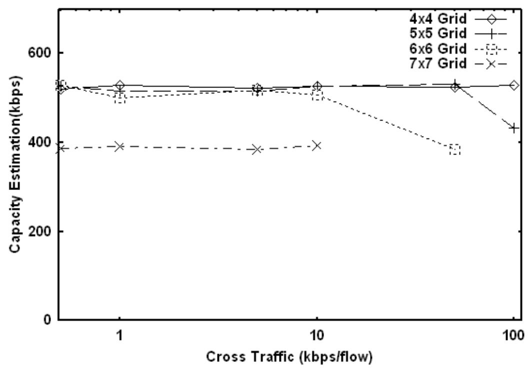


**Figure 8: Capacity estimation in a grid wireless network with horizontal cross traffic**

This experiment essentially reconfirms for the ad hoc network case the property that we already discovered in wired networks. Namely, CapProbe (and now AdHoc Probe) can estimate the capacity correctly up to the point where the path becomes saturated! This is not very intuitive in multihop ad hoc networks where the packets in the pair travel separated by 4 hops. Any cross traffic transmitted by 2-hop neighbors during this 4 hop window will interfere with the pair and invalidate the minimum sum requirement. Thus, for the same network loading, the risk of interference with the packet pair appears to be much higher in ad hoc nets than in wired nets. Yet, AdHoc Probe can still estimate the correct capacity!

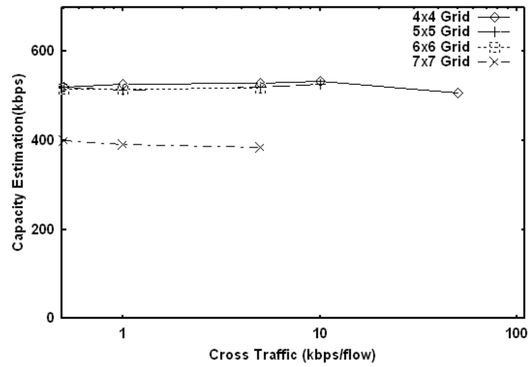


**Figure 9: Capacity estimation in a grid wireless network with both horizontal and vertical cross traffic**

*5.5 Ad Hoc Probe with mobile nodes/Hosts*

After evaluating AdHoc Probe in stationary wireless scenarios, we now apply AdHoc Probe to carefully engineered mobile scenarios where we can control the "path breakage" rate. We want to study AdHoc Probe robustness to path breakage and path reestablishment. Figure 10 depicts the scenario where the AdHoc Probe sender and receiver are fixed, and the intermediate forwarding hosts are mobile. For each wireless node, radio transmission range is set to 250 meters, and the radio interfering range is set to 550 meters. Node 2, 3, 4, and 5 are moving as a group and clockwise along the dotted square path with a fixed speed (which is varying from 10 m/sec to 100 m/sec). Naturally, similar performance is observed when the intermediate nodes move randomly. But, our scenario permits us to control the path breakage rate. In the scenario, Dynamic Source Routing (DSR) is used so that the AdHoc Probe has a path (3 hops) to destination (but the route continuously breaks and must be reconstructed due to mobility). The capacity estimation is performed with 20 separate runs for each configuration (each run with 200 packet pair samples). The average capacity estimates and standard deviations are shown in Figure 11.

The results clearly show that AdHoc Probe is able to accurately estimate path capacity. In other words, enough minimum sum samples are collected while the path is up to allow the correct rate estimation. AdHoc Probe measured around 530 Kbps capacities, which confirmed the estimation results we obtained from the simulation of the chain topology of 3 hops length.

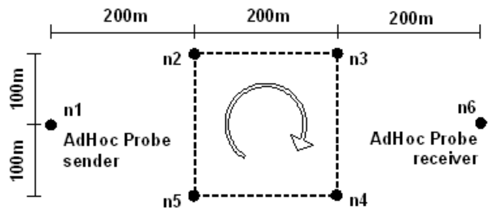


**Figure 10: Scenario of fixed source/destination and mobile intermediate nodes.**

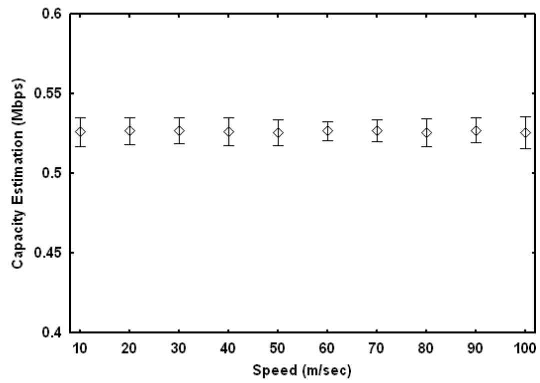


**Figure 11: AdHoc Probe capacity estimates (average of 200 runs and the standard deviation) in simulation scenario.**

*5.6 Monitoring path capacity in Ad Hoc Networks w/ Mobile End Hosts*

After the mobile intermediate nodes scenario, we now examine another mobile scenario, where one of the AdHoc Probe hosts is mobile and the intermediate hosts are fixed.

Figure 12 depicts the mobile Host scenario, consisting of 25 stationary nodes (numbered from 0 to 24) and one mobile Host (node 25). Stationary nodes are arranged as a 5x5 grid, spaced 200 meters apart from their horizontal and vertical neighbors. The mobile node travels along the indicated path at a fixed speed of 1 meter/second.

For the purpose of reducing the number of variables, every node is configured to transmit at a fixed rate of 2Mbps, with a transmission range of 250 meters and an interfering range of 550 meters. DSR routing protocol is used.

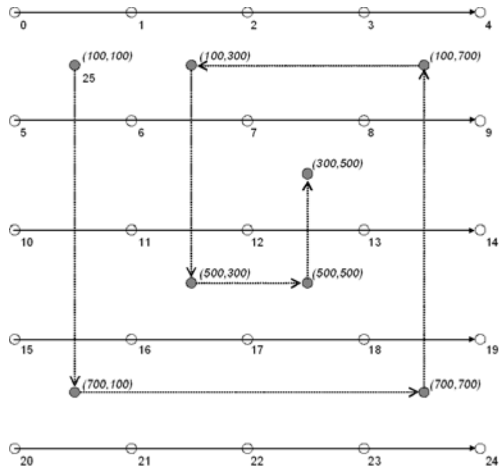


**Figure 12: MANET Scenario, where node 25 (The Host) is moving along the dotted path with a fixed speed (1m/sec).**

AdHoc Probe is deployed to measure the capacity of the connection from a fixed source node (node 0) to the mobile Host (node 25). In this simulation, AdHoc Probe uses packet size = 1500 bytes and probing frequency = 4 samples per second, resulting in a rate equal to 1500*8*2*4 = 96 Kbps. The path capacity is then estimated with and without cross traffic. In the cross traffic experiment, five Poisson UDP flows, each at an average rate of 5kbps, are set up from nodes 0, 5, 10, 15, 20 to nodes 4, 9, 14, 19, 24, respectively. Results are collected and depicted as points in Figure 13.

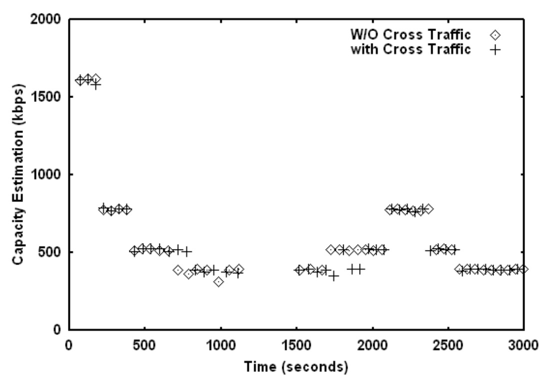


**Figure 13: Capacity estimation of MANET scenario (with and w/o cross traffic).**

In Figure 13, we report the capacity estimation. Note that the capacity will vary as the node moves since the path length in hops changes. AdHoc Probe correctly estimates the capacity as a function of hop distance regardless of cross traffic. The results in Figure 13 match the results of the chain topology in zero load reported in Figure 5.

When node 25 moves away from its initial position (100,100) towards its first destination (700,700), the estimated end-to-end capacity decreases sharply from the original 1600kps to 780kbps, and then to 500kbps. This is because the hop count increases from one to four during the same period.

## 6 Conclusions

In this paper, we have reviewed the definition of "path capacity" in ad hoc wireless networks and have proposed a technique – AdHoc Probe – that can efficiently measure such capacity. The technique is a packet-pair based technique inspired by CapProbe, the equivalent tool used in the Internet. AdHoc Probe measures the wireless path capacity that the user would achieve in absence of competing traffic. The procedure is totally end to end and is thus independent of the specific protocols implemented in the ad hoc network. In this paper we have studied the performance of AdHoc Probe via analysis and simulation. The results showed that AdHoc Probe is able to accurately measure path capacity in all cases. In summary, AdHoc Probe has provided accurate measurements, it is simple, timely, accurate, and much less intrusive than some previously proposed techniques based on sending entire test streams. Work is now underway to extend the AdHoc Probe concept to probe other important properties of wireless ad hoc networks such as path load and random loss rate.

## 7 References


[1] Belding-Royer E. M. and Perkins, C. E., "Evolution and Future Directions of the Ad hoc On-Demand Distance Vector Routing Protocol." Ad hoc Networks Journal, 1(1), pp. 125-150, July 2003.

[2] Dovrolis, C., Ramanathan, P., and Moore, D., *"What do packet dispersion techniques measure?"* in Proceedings of IEEE Infocom'01, 2001.

[3] Jacobson, V., *"Pathchar: A tool to infer characteristics of Internet paths"*, ftp://ftp.ee.lbl.gov/pathchar

[4] Kapoor, R., Chen, L.-J., Lao, L., Gerla, M., Sanadidi, M. Y., "CapProbe: A Simple and Accurate Capacity Estimation Technique," in proc. of ACM SIGCOMM 2004.

[5] Lai, K. and Baker, M., "*Measuring Bandwidth*," In Proceedings of IEEE INFOCOM '99, p. 235-245.

[6] Li, J., Blake, C., Couto, D., Lee, H. I., and Morris, R., "Capacity of Ad Hoc Wireless Networks," in proc. of ACM MobiCom 2001.

[7] Xu, K. , Hong, X., and Gerla M., "An Ad Hoc Network with Mobile Backbones," in Proc. of IEEE ICC 2002.

[8] Xu, K., Gerla, M., and Bae, S., "How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks?" in proc. of Globecom 2002.

[9] NS-2 Simulator, *www.mash.cs.berkeley.edu/ns/*