Efficient Monitoring in Wireless Mesh Networks: Overheads and Accuracy Trade-offs

Dhruv Gupta Prasant Mohapatra Chen-Nee Chuah
University of California Davis
{dhgupta, pmohapatra, chuah}@ucdavis.edu,

Abstract*—802.11-based multi-hop wireless mesh networks have become increasingly prevalent over the last few years. Recently, a lot of focus has been on deploying monitoring frameworks for enterprise and municipal multi-hop wireless networks. A lot of work has also been done on developing measurement-based schemes for resource management and fault management in these networks.** ***The above goals require an efficient monitoring infrastructure to be deployed in the wireless network, which can provide the maximum amount of information regarding the network status, while utilizing the least possible amount of network resources. **However, network monitoring introduces overheads, which can impact network performance, from the perspective of the end user. The impact of monitoring overheads on data traffic has been overlooked in most of the previous works. It remains unclear, as to how parameters such as number of monitoring agents, frequency of reporting monitoring data, and others, impact the performance of a wireless network. In this work, we first evaluate the impact of monitoring overheads on data traffic, and show that even small amounts of overheads can cause large degradation in network performance. We then explore several different techniques for reducing monitoring overheads, while maintaining the objective (resource management, fault management and others) that needs to be achieved. Via extensive simulations, we investigate whether a monitoring framework, which is constrained in terms of number of monitors or periodicity of monitoring, can achieve similar performance as a monitoring framework spanning the entire network. We show that different techniques lend themselves to different application scenarios, and evaluate the trade-offs involved in terms of monitoring overheads and quality of monitoring data.**

I. INTRODUCTION

802.11-based wireless mesh networks (WMNs) have witnessed a tremendous growth over the last few years [1] [2] [3]. A lot of work has been done in terms of understanding the behavior of mesh networks and analyzing the impact of various factors, such as interference and number of hops, on its performance [4] [5] [6]. A lot of focus has also been on providing Quality-of-Service (QoS) in WMNs and several schemes have been proposed for admission control and QoS-based routing [7] [8] [9].

There have been some recent studies on developing measurement-based schemes for resource management and fault management in WMNs [10] [11]. These schemes usually involve measuring certain parameters from the network (such as packet loss rate or signal quality), and utilizing this data for the purpose of QoS provisioning (routing algorithms using loss rate as metric) or for fault management (not using links with low signal quality). WMNs have also found wide applications in enterprise and municipal networks. Such networks need to be monitored constantly for performance degradation and other anomalies. Network operators would like to have an efficient monitoring framework that can provide them with up-to-date network statistics. This is not a trivial task and the challenges involving network management and diagnosis have been addressed in the past [12] [13] [14].

Both the above goals require an efficient monitoring framework that can provide accurate statistics about the network in a timely manner. An online network management system would require transmitting measurement data from various locations to a central server, or the exchange of data among various mesh nodes. *However, in most cases, the same links are used for carrying both the user traffic and the monitoring data. In case of 802.11-based wireless networks, an out-of-band channel (or a wired infrastructure) may not always be available for transferring measurement data from the mesh nodes to a central server.* Deploying dedicated monitoring sniffers, with each node having a connection to the wired backbone, may not always be feasible, and may also be cost-prohibitive. As a result, the transmission of measurement information will contend with the data traffic and reduce the channel time available to the end users. As a result, in a multi-hop wireless network, even a small amount of monitoring traffic can have a large impact on the existing data traffic in the network and cause performance degradation for the end users. Thus, reducing the amount of monitoring overheads in a wireless network is an important goal.

Monitoring techniques can be largely classified into two categories: *statistical monitoring and reactive monitoring* [15]. In statistical monitoring, the central management server collects data from various network sites and derives statistical properties from it, in order to predict future trends. In reactive monitoring, the cen-

1-4244-2575-4/08/$20.00 ©2008 IEEE

tral management server needs measurements from the network in order to react to any events in the network that might signal a potential anomaly or performance degradation. Both the above mentioned monitoring techniques lend themselves to different application scenarios. For example, statistical monitoring is useful for QoS provisioning, while reactive monitoring can be used for network diagnosis and fault management.

For both the above mentioned scenarios, continuous periodic reporting can also be used. An ideal case would be where each wireless router in the WMN also acts as a monitoring agent, reporting periodic data to the central server. This gives us the advantage of having an accurate and up-to-date image of the network status. However, a network-wide monitoring infrastructure, reporting continuous periodic data, may introduce large amounts of overhead.

In this work, we propose to evaluate the following techniques for reducing monitoring overheads in the WMN:

- For QoS provisioning, with statistical monitoring, we can reduce the amount of overheads by reducing the number of wireless routers used as monitors in the network. We use an approximation algorithm for finding vertex covers to locate the nodes that should also serve as monitoring agents. This way, we can reduce the number of monitors in the wireless network, while still being able to monitor every link in the network.
- For detecting anomalies in network performance, we employ reactive monitoring. We evaluate a threshold-based monitoring scheme, wherein the nodes acting as monitoring agents report data only when the measured parameter crosses a certain threshold. This way, we can reduce the volume of monitoring traffic, in comparison with the periodic reporting of monitoring data.
- We evaluate the impact of the frequency of reporting monitoring data to the central server on the end-user's performance. We evaluate the trade-offs, in terms of monitoring overheads and "quality" of information, with varying frequencies of reporting monitoring data.

We propose these techniques not as a replacement for existing network monitoring solutions, but to compliment these schemes by making them more efficient in terms of overheads. For each of the above techniques, we evaluate the gain, in terms of reduction in overheads, and the sacrifice, in terms of the accuracy and quality of measurement data. *This is crucial, as a reduction in overheads will not be desirable if it causes a large degradation in the quality of measurement data.* **Our main contributions can be summarized as follows:**

- **We investigate the effects of monitoring overheads on the forwarding of user data traffic. We show that even small amounts of monitoring traffic can result in increased delays and packet loss for the user data.**
- **We investigate different approaches such as reducing the number of monitors, and threshold-based monitoring, for reducing monitoring overheads in a WMN and evaluate their performance for different application scenarios.**
- **We look at trade-offs involved in accuracy of estimation and reduction in overheads. We investigate as to how accurately can we measure different metrics, and the impact it has on the performance of various applications, while reducing monitoring overheads.**

The rest of the paper is organized as follows. Section II outlines some of the previous work in this area, and the motivation behind this work. Section III explains the proposed methodology. Section IV gives the performance evaluation of the proposed schemes and the simulation results. Section V concludes the paper.

II. RELATED WORK & MOTIVATION

The problem of efficient monitoring in a network has been studied in the past, in the context of the Internet. Several works have studied how to use different polling mechanisms for lowering overheads [16] [17]. However, these works did not consider the impact on the functionality to be achieved. Other works have looked at improving the performance of reactive monitoring in wired networks. In [18] and [15], the authors look at how to combine global polling with local event-based reporting for reducing monitoring overheads.

Several recent works have studied this problem from the aspect of jointly reducing the number of monitors and controlling the sampling rate at the monitors, in order to bring down the monitoring cost, while maximizing the monitoring coverage in terms of the number of flows monitored. In [19], the authors consider the minimizing cost (sum of deployment and monitoring cost) and maximizing coverage (in terms of monitoring reward) problem under various budget constraints. In [20], the authors propose that the monitor placement should be a dynamic process, based on the variations in the network traffic. They propose an approach where a monitor is assumed to be present at every node. The problem is to decide which monitors to activate and what sampling rate to set at each monitor, in order to achieve a measurement task with high accuracy and low resource consumption. In [21], the authors look at the problem of placing a small set of active beacons in the Internet topology. They show that the problem is NP hard for a BGP-like routing topology and present the upper and lower bounds for the number of beacons needed for a given network.

In [22], the authors present the active monitor placement as a combinatorial problem and present a mixed integer programming solution. They propose algorithms to both minimize the number of monitoring beacons and the sampling rate. In [23], the authors consider the problem of placing monitors and setting the sampling rate. Like the previous works, they show the problem to be NP hard and present approximation algorithms to solve the problems.

All the above works approach the monitor placement problem as a joint problem of minimizing cost and maximizing coverage and have presented heuristics solutions. Such an approach is valid for a wired network, especially in an environment such as the Internet backbone, where the number of links or flows to be monitored is huge and thus choosing a sampling rate along with the placement of monitors becomes crucial.

Our work differs from the above in both its setting, and the approach used to solve the problem. All the above works are restricted to wired networks. They do not take into account the special characteristics of wireless networks. A wireless mesh network, being used as an access network, does not compare to the Internet in terms of network size and number of flows and hence does not require sampling rates to be determined for each monitor. Also, wireless networks usually have highly dynamic characteristics (varying interference and link quality), which should be taken into account during the deployment of the monitoring infrastructure. *Thus, we not only evaluate techniques for reducing the monitoring traffic in the network, but we also evaluate their impact on the accuracy of measurement information and performance of various applications.*

One of the primary reasons to investigate the issue of overheads was the emergence of several active-measurement-based schemes for network management in WMNs. For example, routing schemes based on metrics such as ETX [11] and ETT [4], rely on periodic broadcasts to estimate the link quality. However, the impact of overheads on data traffic has not been quantified in these works. In [24], authors have evaluated the impact of overheads associated with ETT-based routing on the wireless network. It was shown that the active probes used by ETT-based routing protocol contend with the data flows for channel access and result in reduced throughput for the end users. These results show that the issue of efficient monitoring in WMNs is an important problem. Our goal is to evaluate different techniques that can help reduce the volume of monitoring data in the network, while achieving the desired performance and functionality. Lower monitoring traffic will translate to lower contention and interference in a wireless network, thereby providing better end-to-end performance to the clients.

III. PROPOSED METHODOLOGY

Our conjecture is that different application scenarios will need different forms of monitoring. Based on the objective for which the monitoring data is being used, different techniques for reducing the measurement traffic volume can be used. As part of this study, we consider two different application scenarios, and show how different techniques lend themselves to each scenario:

- *QoS provisioning:* A municipal ISP may have different service classes for its users, where each service class comes with certain guarantees in terms of network performance (such as bandwidth and delay). In such a scenario, it is necessary for the ISP to continuously monitor the network and collect measurement statistics. Statistical monitoring lends itself to such application scenarios.
- *Network diagnosis & fault management:* An enterprise network administrator would be interested in maintaining the performance of the network above a certain level. In such a scenario, the administrator may not be interested in periodic reports, but would like to get a measurement value only if it signals a degradation in the network performance. Threshold-based monitoring is apt for such application scenarios.

The above examples outline only two of the several different application scenarios that require a monitoring infrastructure. In each of the above scenarios, the volume of measurement data generated by the monitoring framework will directly impact the performance of the end users. Based on the application scenario, we propose to evaluate different approaches for reducing overheads:

A. Monitor Selection Approach

For the first application of QoS provisioning, we consider QoS-based routing. We consider a delay-based routing algorithm, where the objective is to find a path with minimum delay for each client. In order to achieve this, we need to monitor each link in the network, and report the associated delays to the central management server. The central server would then use this information to estimate the end-to-end delays for various network paths. By utilizing this information, the central server can assign a path with the least delay to an incoming client.

In an ideal case, every wireless router in the WMN would also be used as a monitoring agent. Using such a framework would enable us to collect continuous measurement data from every link in the wireless network. However, such a framework may introduce large volumes of monitoring traffic in the network, thereby adversely affecting the performance of data traffic. *In order to reduce the monitoring overheads, we propose to limit the number of wireless routers for monitoring* *purposes, while still achieving the goal of delay-based routing.*

We decided to evaluate the performance of vertex-cover algorithm for this purpose. We use this algorithm to locate the network sites to be used for monitoring purposes. A mesh network can be modeled as a graph $G = (V, E)$, where V is the set of nodes, representing the mesh access points, and E is the set of edges, representing the links between the mesh access points. We want to choose a set of k nodes, from N nodes in the network, to be used for monitoring. The above problem is similar to the vertex-cover problem in graph theory. For our problem, if we can find a vertex cover for our network, then we have a set of nodes which we can use as monitoring agents. This would ensure that we cover all the links in the network for the purpose of monitoring, while using the minimum possible number of nodes.

A simple approximation to the vertex-covering algorithm consists of picking a random edge from the graph and adding the vertices of the edge to the vertex cover. It then removes all the edges incident on these two vertices, as they have been covered, and then repeats the above process. The running time of this algorithm is $O(V + E)$. This algorithm is a polynomial-time 2-approximation algorithm [25].

However, we should not select any random network site to be used as a monitoring agent. The selection process should take into account some network characteristics. We include the effects of network topology in the monitor selection decision. To do this, we use the vertex cover approximation algorithm that chooses vertices in decreasing order of their degrees. The rationale behind this approach is that the vertex with the maximum degree would reflect the node that has the maximum number of links with other nodes in the mesh network. Thus by choosing the nodes with higher degrees, we will be monitoring a larger number of links in the network.

Algorithm 1 MAX-DEGREE-NODE-VERTEX-COVER (G)

```
$C \leftarrow NULL$
$V' \leftarrow V[G]$
$E' \leftarrow E[G]$
while $E' \neq NULL$ do
    $find\_max\_degree\_node(V')$
    $C \leftarrow C \cup v$
    remove $v$ from $V'$
    remove from $E'$ all edges incident on $v$
end while
return $C$
```

B. Threshold-based Monitoring

For the second application of detecting performance degradations in the network, we propose to use a threshold-based monitoring framework in order to reduce the volume of monitoring traffic. Our objective here is to report data to the central server only when a certain event occurs. Specifically, in our case, we consider the traffic load on a node as an indication of congestion. If this load value crosses a certain threshold, then it might indicate an onset of congestion, and the administrator may want to route packets around that node. Ideally, we could use the periodic reporting mechanism, where every wireless router continuously sends monitoring data to the central server. However, this would generate large amounts of overheads. We thus evaluate the performance of the threshold-based reporting mechanism to achieve our objective of reducing monitoring overheads while maintaining the desired functionality of identifying network performance anomalies.

It should be noted that we cannot use the threshold-based monitoring framework for QoS provisioning. This is because, for providing performance guarantees to end users, in terms of throughput or delay, we need continuous measurements from the network, in order to have an accurate and up-to-date image of the network status. Thus, even though this framework would help meet our objective of lowering monitoring overheads, it would not be able to achieve the desired functionality of providing QoS. Hence the need for different monitoring techniques for different application scenarios.

C. Impact of Monitoring Frequency

An important parameter in any monitoring framework is the frequency at which we report monitoring data. Reporting data at a high frequency (such as per-second basis) enables us to maintain a more accurate image of the network. More fine-grained information will be available to the network administrator for providing resource control or fault management. However, this approach suffers from high monitoring overheads. In order to avoid this, we can use a framework where the monitoring nodes maintain an average of various parameters and report data at longer intervals. This approach will result in less monitoring overheads being generated in the network. However, the reporting interval (also referred to as monitoring frequency here) should be selected appropriately, so as to not impact the desired functionality. We investigate the performance of our delay-based routing protocol using different reporting intervals and evaluate the trade-offs in terms of overheads and accuracy of monitored data.

IV. PERFORMANCE EVALUATION

A. Simulation Setup

In order to evaluate our proposed scheme, we have used the QualNet simulator. We use three topologies for comparing the performance of various schemes. The first one is a twenty-five node grid topology generated in QualNet (hereafter referred to as the $GRID$ topology). The second topology is a fifteen node topology, derived from the indoor testbed used in [26] (hereafter referred to as the $INDOOR$ topology). The third topology was a fifteen node topology, based on the outdoor network described in [2] (hereafter referred to as the $OUTDOOR$ topology). The readers can refer to the publications cited for more information on these networks, such as topology etcetera. The gateway node (having a wired connection to the Internet) was chosen randomly for all the topologies. Ad-hoc On-demand Distance Vector (AODV) routing protocol [27] was used as the routing protocol of choice.

For $GRID$ and $OUTDOOR$ topologies, we had six traffic sources placed uniformly across the wireless mesh network. For $INDOOR$ topology, we had three traffic sources placed uniformly across the topology. The traffic was generated to simulate a file transfer. In all three cases, the data traffic was destined for the gateway node. The amount of data generated was the same for each scenario. The data packet size was set to 1500 bytes. The modulation rate of every node was fixed at 2 Mbps, and all the nodes were operating on the same channel.

We set the default reporting frequency for the monitoring agents at one packet per second, that is, each monitoring node will send a packet with measurement data every one second. We assume that all the data has to be sent to a central server which is co-located with the gateway node. In a practical network, this server could be the network operation center where the network operator can collect and analyze all the data on-the-fly or it could be a server for storing measurement data, which can be used later for off-line analysis. The monitoring data is sent using UDP at the transport layer. Each scenario was repeated five times. The following sections describe the various results.

B. Monitor-Selection Approach

In this section, we present the results for the monitor selection approach, wherein we use an approximation algorithm for finding vertex-covers to locate the wireless routers to be used as monitoring agents. The max-degree VC algorithm returned a thirteen node vertex cover for $GRID$ topology, eight node vertex-cover for $OUTDOOR$ topology and a nine node vertex-cover for the $INDOOR$ topology.

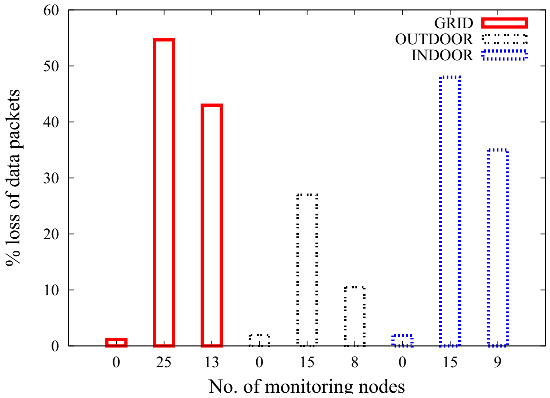

Fig. 1. Comparison of data loss for monitor selection approach.

We first evaluated the impact of monitoring overheads on the data traffic, in order to justify the need for reducing monitoring overheads in WMNs. We specifically look at how the periodic reporting of monitoring information to a central server impacts the flow of user data, in terms of packet loss and end-to-end packet delay. Figure 1 shows that when no monitoring agents are used in the network, the associated losses are minimal. This means that the network is not saturated to begin with.

Figure 1 also shows the percentage loss of data packets, when all the nodes in the topology are used as monitoring agents (25, 15 and 15 for the $GRID$, $OUTDOOR$ and $INDOOR$ topology), and when only the nodes in the vertex-cover set are used as monitoring agents (13, 8 and 9 for the $GRID$, $OUTDOOR$ and $INDOOR$ topology). As can be seen, for a network-wide monitoring infrastructure, the impact of monitoring overheads on data traffic can be substantial. However, by using a limited set of nodes, we can reduce the amount of overheads in the network. A similar impact was seen on the end-to-end delays of the data flows (shown in Figure 2). These results clearly indicate that using a network-wide monitoring infrastructure can generate large amounts of overheads, which can adversely impact the end users' performance. By using a fewer number of monitoring agents, we are able to alleviate the performance by a significant amount.

In order to further quantify the impact of monitoring overheads on WMNs, we also looked at the frame retransmissions at the link layer. In 802.11 protocol, if a frame is lost, the link layer will re-transmit the frame a certain number of times, before reporting it as lost to the higher layers. If the number of re-transmissions is high, it will result in increased contention in the WMN, and will also increase the end-to-end delay for the flow. Figure 3 shows the improvement in link layer re-transmissions when only the nodes in the vertex-cover are used for monitoring, as compared to the network-wide monitoring. The reduction in the number of packets that are re-transmitted also explains the reduction in end-to-end delay in Figure 2.

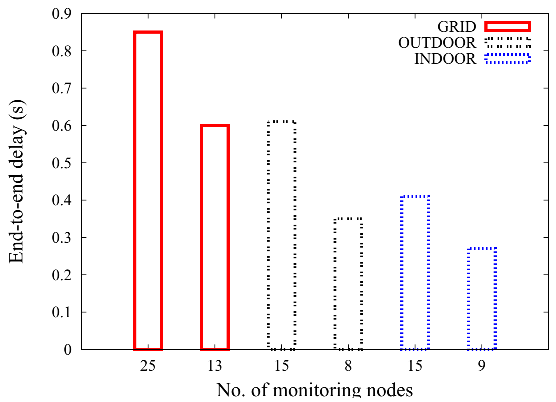

Fig. 2. Comparison of end-to-end delay for monitor-selection approach.

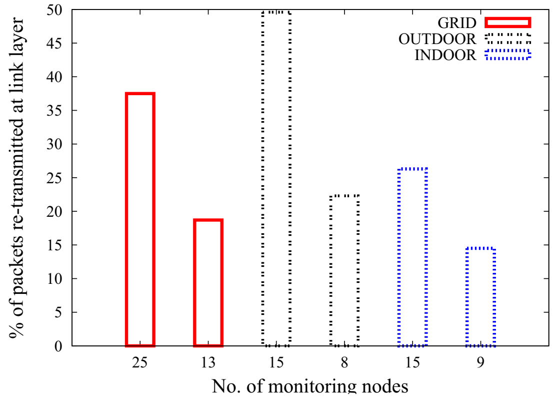

Fig. 3. Comparison of link layer re-transmissions for monitor-selection approach.

Topology	**No. of monitoring nodes**	**Average Throughput**	**Average Delay**
GRID	25	1.16 Mbps	0.72 s
	13	1.25 Mbps	0.68 s
OUTDOOR	15	1.27 Mbps	0.70 s
	8	1.42 Mbps	0.62 s
INDOOR	15	1.03 Mbps	1.17 s
	9	1.15 Mbps	0.94 s

TABLE II
PERFORMANCE OF DELAY-BASED AODV FOR MONITOR-SELECTION APPROACH.

The results presented above confirm the fact that monitoring overheads can severely impact the performance of end users and that using a smaller number of monitors will reduce the communication overheads involved. However, it is also essential to maintain the accuracy of estimation of various parameters. By using a smaller number of monitors, we may have to sacrifice some “quality of information” in lieu of reduced monitoring overheads.

In order to investigate this, we decided to measure the end-to-end delays for various flows in our simulation. We first measured this value using all the nodes in the WMN as monitoring agents, and then using only the nodes in the vertex-cover set as monitoring agents. In both cases, we compared the measured value with that reported by the simulator. In order to measure the end-to-end delays, we measure the delay incurred by a packet at each hop along the route it takes to the gateway node. The sum of the per-hop delays gives us an estimate of the end-to-end delay. Table I shows the comparisons between the measured end-to-end delay, and the delay reported by the simulator, for each of the topologies. The client, for which the delay was estimated, was four hops away from the gateway node for all three topologies.

It was observed that reducing the number of monitoring agents caused the estimation error to double for all the topologies. The reason for the measured values of delay to have larger errors when using a smaller number of monitors is as follows. When all the nodes in the network were being used for monitoring, each node could measure the delay involved in sending a packet to its neighbor. However, with the reduced number of monitors, for some links we are able to measure the delay in one direction only, and use this value as the delay in both the directions. Owing to the asymmetry of links in a wireless network, the delay on a link could be different in both the directions, which is captured in the first case, but not in the second.

The subsequent question that arises is whether the quality of information has degraded significantly or not. In other words, can such a measurement framework still be utilized for achieving a certain goal? In order to investigate this further, we decided to modify the existing implementation of AODV protocol in QualNet, to choose the next hop neighbors based on the delay values. That is, when AODV selects which next hop to choose to forward the packet to, the hop with the lowest delay is chosen. This mechanism can be used to provide delay-based QoS guarantees to the end users. We compare the performance of AODV for the two monitoring frameworks for all the topologies. Table II shows the average throughput and delay values measured from the network for the three topologies. *It can be seen that the proposed framework (with reduced monitoring agents) achieves better performance than the ideal case of network-wide monitoring.* This is because the increase in estimation error is offset by the reduction in overheads, resulting in improved network performance. Even though the protocol may not choose the best routes, there is less contention in the network, which improves the network performance. Hence, we can use the proposed monitoring framework for provisioning QoS in the network, while reducing overheads at the same time.

Topology	No. of monitoring nodes	Delay from simulator	Delay from measurements	% Error
GRID	25	0.8 s	0.76 s	5%
	13	0.54 s	0.49 s	9.26%
OUTDOOR	15	0.62 s	0.595 s	4%
	8	0.36 s	0.317 s	11.9%
INDOOR	15	0.41 s	0.384 s	6.3%
	9	0.263 s	0.23 s	12.5%

TABLE I
ACCURACY OF DELAY ESTIMATION FOR MONITOR-SELECTION APPROACH.

No. of nodes removed	Degree-based node removal	Load-based node removal
1	97.5%	96.35%
2	95%	92.5%
3	92.5%	88.75%

TABLE III
IMPACT OF NODE REMOVAL TECHNIQUE ON LINK COVERAGE.

For the monitor selection approach, we also considered the problem of which wireless routers to choose as the monitoring agents. In other words, if we want to reduce the number of monitoring agents in the mesh network, then should we choose the nodes randomly, or should the selection process depend on certain criterion. In order to investigate this, we consider two different approaches. The first one is where we choose the nodes based on their degrees, that is, the node with lowest degree will be removed first from the set of monitoring nodes. The second case is where we choose the nodes based on their traffic loads. In this case, the node with the maximum traffic load will be removed first from the set of monitoring nodes. We compare these two approaches in terms of loss of data packets and network coverage (in terms of number of links monitored), thus evaluating the trade-off involved between reduction in packet loss and loss of network coverage. Figure 4 shows the impact on packet loss.

As can be seen, the degree-based node removal achieves a slightly better performance is terms of reducing packet loss. Another advantage of using this technique is that it helps is maintain a greater link coverage. For example, Table III shows the link coverage for $GRID$ topology. As we can see, removing nodes with lower degrees helps us maintain higher link coverage. Similar results were seen for the other two topologies.

C. Impact of Monitoring Frequency

In this section, we evaluated the impact of varying the frequency of reporting monitoring information. We investigate as to how this parameter impacts the measurement overheads and the accuracy of estimation. Reporting data very frequently (such as per-second basis) would provide us with a more accurate image of the network. However, such an approach may also incur large overheads. Figure 5 shows the percentage of packets that were re-transmitted at the link layer for varying monitoring frequencies. As the reporting interval increases (monitoring frequency decreases), there are less overheads in the network. As a result, fewer number of packets need to be re-transmitted at the link layer. The improvement in the number of re-transmissions results in improved performance for end users (Figure 6 and 7). Since the amount of monitoring traffic reduces with decreasing frequency, there is less contention in the network, resulting in better performance for end users.

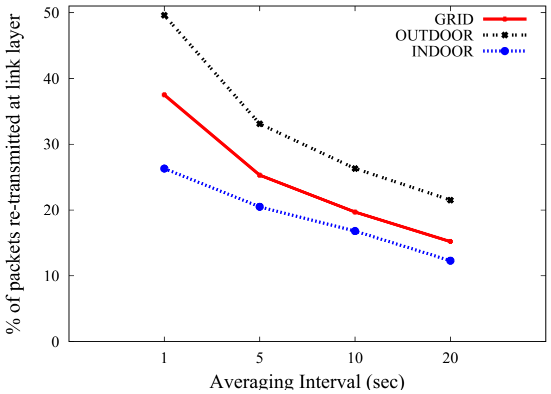

Fig. 5. % of packets re-transmitted at link layer.

However, our objective is not to just reduce the volume of monitoring overheads in the network. We also want to maintain the functionality that needs to be achieved using the measurement data. We once again consider the example of delay-based routing. We evaluate the performance of our modified protocol for the two cases. In the first case, from every wireless node, we report network statistics to the central server every one second. That is, each node will send out one monitoring packet per second, destined for the central server. The second case is where each node sends out a measurement packet every ten seconds. In this case, every node maintains a simple moving average of the parameters that it is measuring and sends out a packet with the monitoring

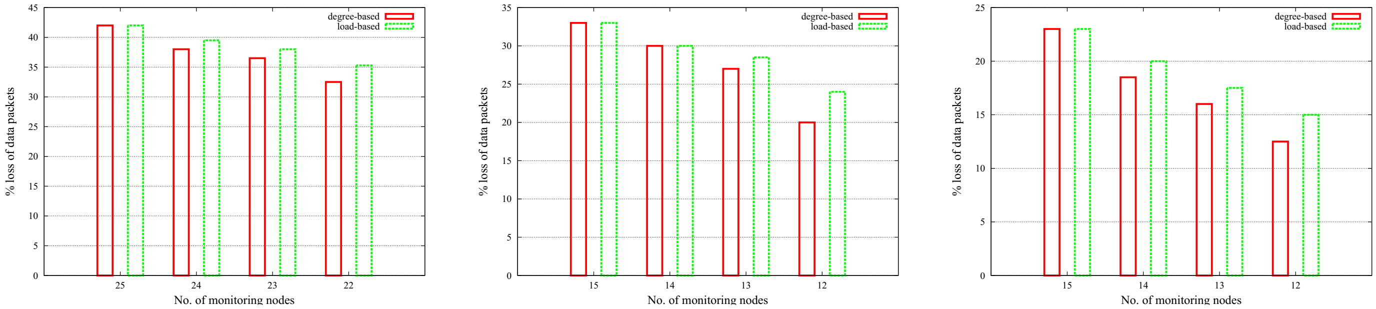

Fig. 4. Impact of node removal technique on packet loss for GRID, OUTDOOR, INDOOR topologies.

Topology	**Averaging Interval**	**Delay from simulator**	**Delay from measurements**	**% Error**
GRID	1	0.8 s	0.76 s	5%
	10	0.68 s	0.60 s	11.76%
OUTDOOR	1	0.46 s	0.43 s	6.52%
	10	0.327 s	0.285 s	12.8%
INDOOR	1	0.41 s	0.384 s	6.3%
	10	0.32 s	0.278 s	13.12%

TABLE IV

ACCURACY OF DELAY ESTIMATION WITH VARYING MONITORING FREQUENCY.

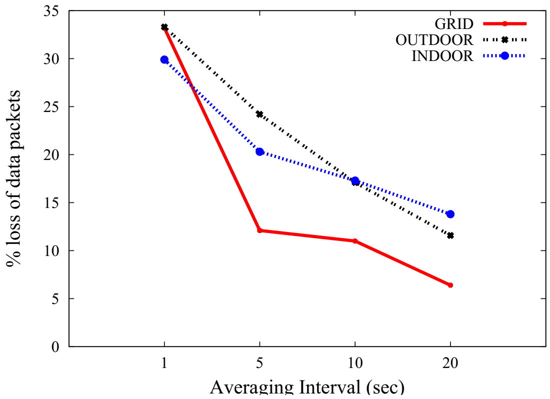

Fig. 6. % loss of data packets with varying monitoring frequency.

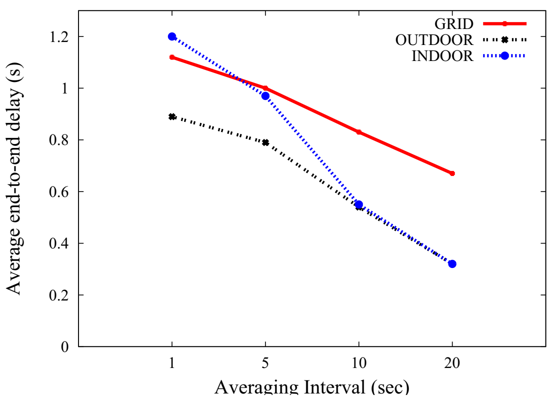

Fig. 7. Average end-to-end delay with varying monitoring frequency.

information after every ten seconds.

In order to analyze the performance of this scheme, we first compare the delay measurements using the two schemes. Table IV shows the delay values for a particular client for the three topologies. When the monitoring information is sent every second, the accuracy is fairly high. When we send monitoring data averaged over ten seconds, the accuracy of estimation drops a little. We further investigated the efficiency of this approach by comparing the performance of our delay-based routing protocol for both the scenarios. Tables V compares the results for the three topologies. As can be seen, averaging the data over ten seconds, achieves slightly better performance than the first case. THis is because the reduction in overheads compensates for the loss of accuracy, and hence there is more capacity available in the network, for data transmission. These results indicate that we can easily use a lower frequency of reporting monitoring data for QoS provisioning, without sacrificing our performance.

D. Threshold-based Monitoring

In this section, we present the results for the threshold-based monitoring approach. As explained earlier, such a scheme would be highly beneficial for tracking anomalies in network performance. We wish to evaluate the trade-offs involved, in terms of overheads and accuracy, between the two choices of using periodic monitoring and threshold-based monitoring. We consider the specific example of tracking the traffic load on each node, as an

Topology	Averaging Interval (s)	Average Throughput (Mbps)	Average Delay (s)
GRID	1	0.86	0.825
	10	1.09	0.64
OUTDOOR	1	0.97	0.485
	10	1.25	0.34
INDOOR	1	1.03	1.17
	10	1.24	0.95

TABLE V
PERFORMANCE OF DELAY-BASED AODV WITH VARYING MONITORING FREQUENCY.

indication of network being congested. We wish to identify the events when the traffic load on a node crosses a certain pre-defined threshold, and report it to the central controller. For every packet generated by a node, the node keeps track of the time at which the packet was put in the output queue, and the time at which the link layer acknowledgement for the packet was received. The difference in these two time intervals gives us an estimate of the round trip delay on a particular hop. We use this as an indication of the congestion at the node. If the network is highly congested, neighboring transmissions will cause the packet to stay in the output queue for a longer time, thereby increasing the round-trip delay. It may also happen that the transmitted frame is lost (or ACK timeout occurs), and the frame is re-transmitted. This will increase the time interval of the ACK reception, indicating that the network is congested. Each node keeps track of the round trip delay to its respective neighbors, and reports it to the central controller, either periodically, or using the threshold-based mechanism.

Both periodic and threshold-based monitoring can be used to achieve this objective. However, it was observed that periodic reporting can lead to large delays and packet losses for end users. In the absence of any monitoring in the network, we did not see any packet loss. For the periodic reporting mechanism, we set the reporting frequency to the default value of 1 second. Table VI shows the percentage loss for data packets for the three topologies, while Table VII shows the average end-to-end delay. In periodic reporting, monitored data is sent to the central controller, irrespective of what the network status is. On the other hand, with the threshold-based mechanism, each node will report the round-trip delay only when it crosses the pre-defined threshold. Thus, by using threshold-based monitoring, we can reduce the amount of overheads in the network, thereby improving the performance for the data flows (as indicated by lower packet losses and delay).

We further investigated whether the threshold-based scheme satisfies our second objective of maintaining the desired application performance. We use both the periodic and the threshold-based monitoring mechanisms to report data to the central server. In the periodic framework, one monitoring packet per second is sent from every node to the central server. The central server will check the data in the packet and determine if the reporting node is congested. In the threshold-based mechanism, each node in the wireless mesh network will send a monitoring packet only if it perceives itself as being congested, based on the criterion explained above. This helps us in reducing the volume of monitoring traffic in the mesh network, as shown by the results above. At the same time, we saw an improvement in the performance of the monitoring framework. Table VIII shows the ratio of the “total *useful* monitoring packets” received at the central server to the “total monitoring packets” sent out by all the nodes in the WMN. We define the “total *useful* monitoring packets” as the number of monitoring packets that help us identify node congestion (that is, the reported value of traffic load on the node is above the threshold). For example, for the periodic reporting mechanism, only some of the delivered packets will report the round-trip delay value to be greater than the threshold, thereby signalling a congested node. However, the rest of the monitoring packets sent out are not relaying any *useful* information. Thus, a greater delivery ratio translates to more useful information being transmitted per monitoring packet.

It was observed that with the periodic monitoring framework, a lot of monitoring packets were lost. *Mon-*

TOPOLOGY	Reporting Mechanism	% loss of data packets
GRID	Periodic	21.6
	Threshold	5.86
OUTDOOR	Periodic	27.26
	Threshold	14.69
INDOOR	Periodic	29.9
	Threshold	11.6

TABLE VI
% LOSS OF DATA PACKETS.

TOPOLOGY	Reporting Mechanism	Avg. end-to-end delay (s)
GRID	Periodic	1.15
	Threshold	0.82
OUTDOOR	Periodic	0.58
	Threshold	0.375
INDOOR	Periodic	1.32
	Threshold	0.61

TABLE VII
AVERAGE END-TO-END DELAY.

Topology	**Periodic Monitoring**	**Threshold-based Monitoring**
GRID	0.213	0.647
OUTDOOR	0.369	0.736
INDOOR	0.265	0.936

TABLE VIII
DELIVERY RATIO OF USEFUL MONITORING PACKETS.

itoring packets that report traffic load below the threshold, do not actually provide us with useful information, while consuming network resources. These unnecessary transmissions add to the contention in the network, and as a result, there is loss of information at the central server, due to which we may not be able to accurately identify node congestion. On the other hand, with the threshold-based monitoring framework, the loss of monitoring packets is much less, and the central server has more accurate and up-to-date information of the traffic load on various nodes. As a result, even though fewer monitoring packets are being transmitted on the whole, the delivery ratio is higher. This is because fewer packets are lost and a higher percentage of monitoring packets reach the central server. Moreover, all these packets contain *useful* information. Hence, the threshold-based scheme will be able to predict anomalies in the network performance (increased congestion in our case) with greater accuracy.

V. CONCLUSIONS, INFERENCES AND FUTURE WORK

In this paper, we have looked at the issue of efficient monitoring in wireless mesh networks. With their growing popularity and increasing applications, several schemes for implementing Quality of Service and developing measurement-based models for wireless mesh networks have been proposed. Most of these schemes rely on an underlying monitoring framework, which collects the necessary statistics from the wireless mesh network. However, *the impact of monitoring overheads on the performance of data traffic in the wireless network have not been studied so far.* Most previous works that propose active-measurement-based schemes for routing or fault management in wireless mesh networks, have overlooked the issue of overheads. Thus, we first look at the impact of monitoring traffic, on the forwarding of user data traffic, for different applications. We show that even small amounts of monitoring overheads can cause a large degradation in the end user's performance. Via extensive simulations, we evaluate the performance of several schemes for reducing the monitoring overheads in WMNs. We look at monitor selection based on network characteristics such as topology, changing frequency of reporting monitoring data and threshold-based monitoring, as possible solutions to the problem of reducing overheads. We evaluate as to how these schemes lead to an improvement for the end users' performance, in terms of packet loss and delay. We also investigate whether these techniques impact the desired functionality for which the network is being monitored. We evaluate the performance of different applications using these monitoring techniques. Some of the important lessons learned as part of our work are:

- Given the importance of measurement-based approaches for providing Quality of Service and fault management in wireless mesh networks, it is crucial to study the impact of monitoring traffic on the user data traffic. Through our study, we find that periodic monitoring of a network can cause data loss of as much as 40% and can severely impact the network performance from an end user's perspective.
- By using different techniques such as constrained number of monitors and threshold-based monitoring, we can greatly improve the network performance. These techniques help us in maintaining the desired level of measurement accuracy, while reducing the associated overheads.
- We observed that different monitoring techniques lend themselves to different application scenarios. It is crucial to use the right technique for an application, in order to maintain the balance between reduction in overheads and accuracy of measurement data.

As part of our future work, we intend to study the impact of the proposed schemes on more varied topologies. We also plan to see what different metrics can we measure using such frameworks and with what accuracy. We would also like to investigate further applications of these monitoring frameworks. A monitoring framework with varying characteristics (such as topology and periodicity of measurement), that can adapt itself to the network status, is also an objective of our future research.

VI. ACKNOWLEDGEMENTS

We would like to thank the anonymous reviewers for their helpful comments. The work for this paper was supported in part by grants from National Science Foundation (CNS-07009264), Intel Corp., Army Research Office (W911NF-07-1-0318) and UC MICRO program.

REFERENCES

[1] P. De, A. Raniwala, S. Sharma, and T. Chiueh, "Design considerations for a multihop wireless network testbed," *Communications Magazine, IEEE*, vol. 43, 2005.

[2] D. Wu, D. Gupta, S. Liese, and P. Mohapatra, "Qurinet: quail ridge natural reserve wireless mesh network," in *the 1st International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization (WiNTECH)*, 2006.

[3] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, "Architecture and evaluation of an unplanned 802.11b mesh network," in *the 11th Annual international Conference on Mobile Computing and Networking (MobiCom)*, 2005.

[4] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *the 10th annual International conference on Mobile computing and networking (MobiCom)*, 2004.

[5] A. Raniwala and T. Chiueh, "Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network," in *the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM)*, 2005.

[6] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proceedings of the 11th annual international conference on Mobile computing and networking (MobiCom)*, 2005.

[7] L. Chen and W. B.Henzelman, "Qos-aware routing based on bandwidth estimation for mobile ad hoc networks," *Selected Areas in Communications, IEEE Journal on*, vol. 23, pp. 561–572, 2005.

[8] H. Zhu and I. Chlamtac, "Admission control and bandwidth reservation in multi-hop ad hoc networks," in *Computer Networks*, 2005.

[9] I. D. Chakeres and E. M. Belding-Royer, "Pac:perceptive admission control for mobile wireless networks," in *1st International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QSHINE)*, 2004.

[10] D. Gupta, D. Wu, C. Chen, P. Mohapatra, C. Chuah, and S. Rungta, "Experimental study of measurement-based admission control for wireless mesh networks," in *4th EEE International Conference on Mobile, Ad-hoc and Sensor Systems (MASS)*, 2007.

[11] D. DeCouto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *the 9th Annual international Conference on Mobile Computing and Networking (MobiCom)*, 2003.

[12] R. Chandra, V. Padmanabhan, and M. Zhang, "Wifiprofiler: cooperative diagnosis in wireless lans," in *the 4th international Conference on Mobile Systems, Applications and Services (MobiSys)*, 2006.

[13] P. Bahl, J. Padhye, L. Ravindranath, M. Singh, A. Wolman, and B. Zill, "Dair: A framework for managing enterprise wireless networks using desktop infrastructure," in *the 4th Workshop on Hot Topics in Networking (HotNets)*, 2005.

[14] Y. Cheng, J. Bellardo, P. Benk, A. Snoeren, G. Voelker, and S. Savage, "Jigsaw: solving the puzzle of enterprise 802.11 analysis," in *SIGCOMM Computer Communication Review*, 2006.

[15] M. Dilman and D. Raz, "Efficient reactive monitoring," *Selected Areas in Communications, IEEE Journal on*, vol. 20, 2002.

[16] P. Moghe and M. Evangelista, "An adaptive polling algorithm," in *IEEE/IFIT Network operations and Management Symposium (NOMS)*, 1998.

[17] K. Yoshihara, K. Sugiyama, H. Horiuchi, and S. Obana, "Dynamic polling scheme based on time variation of network management information values," in *Integrated Network Management Symposium (IM)*, 1999.

[18] J. e. a. Jiao, "Towards efficient monitoring," *Selected Areas in Communications, IEEE Journal on*, vol. 18, 2000.

[19] S. Kyoungwon, G. Yang, J. Kurose, and D. Towsley, "Locating network monitors: complexity, heuristics, and coverage," in *Proceedings of 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2005.

[20] G. Cantieni, G. Iannaccone, C. Barakat, C. Diot, and P. Thiran, "Reformulating the monitor placement problem: Optimal network-wide sampling," in *the 40th Annual Conference on Information Sciences and Systems*, 2006.

[21] J. Horton and A. Lpez-Ortiz, "On the number of distributed measurement points for network tomography," in *the 3rd ACM SIGCOMM Conference on Internet Measurement (IMC)*, 2003.

[22] C. Chaudet, E. Fleury, I. Lassous, H. Rivano, and M. Voge, "Optimal positioning of active and passive monitoring device," in *the ACM Conference on Emerging Network Experiment and Technology (CoNEXT)*, 2005.

[23] X. Liu, J. Yin, Z. Cai, and S. Lv, "On the placement of active monitor in ip network," in *the 3rd International Conference on Computer Networks and Mobile Computing (ICCNMC)*, 2005.

[24] B. Keegan, K. Kowalik, and M. Davis, "Experimental measurement of overhead associated with active probing of wireless mesh networks," in *IEEE International Conference on Signal Processing and Communications (ICSPC)*, 2007.

[25] T. H. Cormen, C. E. Leiserson, R. L. Rivest, , and C. Stein, *Introduction to Algorithms*. MIT Press, 2001.

[26] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Measurement-based models of delivery and interference in static wireless networks," in *SIGCOMM Computer and Communications Review*, 2006.

[27] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *2nd IEEE Workshop on Mobile Computer Systems and Applications*, 1999.