Throughput Improvement by Reducing Dropped Packets at Interface Queue (IFQ) in Multi-Channel Wireless Mesh Networks

Aisha Mousa Mashraqi  
Department of Informatics  
University of Leicester  
Leicester, UK  
e-mail: amym2@leicester.ac.uk

Thomas Erlebach  
Department of Informatics  
University of Leicester  
Leicester, UK  
e-mail: te17@leicester.ac.uk

Abstract—Wireless Mesh Networks (WMNs) are gaining popularity due to the features provided, especially the low cost and self-configuration ability. In WMNs, the data traffic is transmitted through intermediate nodes and a number of packets are dropped during the transmission due to various reasons. One significant reason is the limitation of network resources such as bandwidth and buffer size. Because of these limitations, the network becomes congested under high traffic volumes. Therefore, a large number of packets will be dropped due to buffer overflow in the Interface Queue (IFQ). Hence, controlling the congestion at the IFQ is essential for achieving high throughput. In this paper, we address the node congestion level in multi-channel WMNs and reduce the dropped packets at IFQ by adjusting the traffic rate based on the solution of a linear program (LP). The benefits of these traffic rate adjustments are demonstrated in chain networks using the Network Simulator (NS2). In addition, for complex networks we show that such traffic rate adjustments alone are not sufficient and propose a simple forwarding delay scheme for the Ad Hoc On-Demand Distance Vector protocol with Forwarding delay (AODV-F) that reduces node congestion and improves throughput, which is again demonstrated in simulations with NS2.

Keywords-Multi-channel wireless mesh networks, Linear programming, Congestion control, Throughput.

I. INTRODUCTION

In recent years, wireless mesh networks (WMNs) have gained popularity, especially in the local community due to features such as self-configuration, easy access to the network, and low cost. In addition, the nodes in WMNs can act as a router or a client or both and communicate via radio. IEEE 802.11 is one of the wireless communication technologies that is commonly used in wireless broadband services and is known as Wi-Fi. This popularity comes with easy deployment and cheap equipment.

Watching streaming video in wireless networks requires huge bandwidth and buffer sizes. Therefore, the network bandwidth and buffer sizes are the main resources that need to be managed effectively to improve the network performance. There are various issues that affect the network performance. A serious issue is packet loss due to interference or congestion [4]. One of the reasons for network congestion is high traffic load. Congestion causes a lot of packet drops due to buffer overflow, especially at forwarding nodes. In this work, we consider the node congestion occurring under high traffic load leading to buffer overflow as shown in Figure 1. Figure 1 illustrates the occurrence of congestion due to buffer overflow at the Interface Queue (IFQ) level at node 2. The node receives packets at a higher rate than it can transmit, and hence becomes a bottleneck node. In this case, a huge number of packets will be dropped at the IFQ buffer, leading to a throughput reduction. To address this problem, a congestion control mechanism is required. To solve the congestion in wireless networks, several studies have been conducted. It is essential to adjust the flow rate to reduce the excess packets and control the network congestion in order to reduce the packet losses and improve network performance.

Our contributions are as follows: First, we use a linear programming (LP) approach to determine the maximum total throughput and corresponding traffic flow rates for a given scenario. After solving the LP, we use the flow rates provided by the LP to adjust the actual traffic flow rates in NS2 simulations. For simple networks, this flow rate adjustment reduces the number of dropped packets at the IFQ. Second, for more complex networks we propose a new combination of a forwarding delay scheme with flow rate adjustment for tackling packet losses at the IFQ buffer. We measure the performance of the flow rate adjustment and forwarding delay scheme by conducting simulation experiments and comparing it with the performance of standard Ad Hoc On-Demand Distance Vector (AODV) in multi-channel WMNs. The paper is organized as follows: Section II discusses the previous work. We explain the linear programming model and flow rate adjustment in Section III and the forwarding delay scheme in Section IV, followed by the performance evaluation in Section V. Section VI gives our conclusion.

II. RELATED WORK

Controlling congestion in wireless networks is a big challenge. In congested networks, a source and destination often cannot communicate properly. There are different
approaches for controlling congestion in wireless networks. Some of the existing work aims to control the congestion of the links in the network by finding alternative paths or using multipath mechanisms. In this Section, we briefly discuss some of the previous work. One of the primary congestion control approaches is the Transmission Control Protocol (TCP) [3]. The TCP applies a mechanism for minimising network congestion by controlling the flow rate. The TCP provides a method to manage the amount of sending data. Tran and Raghavendra proposed a routing protocol called Congestion Adaptive Routing Protocol (CRP) using the bypass concept to prevent congestion from occurring [7]. A bypass is a subpath connecting a node with a non-congested node. If a node detects congestion, then CRP distributes the incoming traffic over the bypass and the primary routes. Another algorithm called Traffic-Aware Dynamic Routing (TADR), proposed by He et al. [1], employs two fields, the depth field and the queue length field. The depth field provides a shortest path mechanism while the queue length field is used by the TADR mechanism. Using the queue length field, if congestion appears, the packets get distributed along multipaths to idle or underloaded nodes. Because using multipath increases the probability of link collision, the performance of CRP [7] and TADR [1] routing protocols might be affected. On the other hand, other studies have applied alternative techniques. For example, Sergiou and Vassiliou [5] proposed a congestion avoidance and control algorithm (DAIPaS). DAIPaS employs two stages (soft stage and hard stage). Every node enters the soft stage when it receives packets from more than one flow, attempting to reach a situation where each node receives data from only one flow. In the case of high traffic volume this is not achievable, and therefore the node enters the hard stage that forces the selection of an alternative path. The alternative path selection is based on path cost, remaining node energy and buffer threshold. Another algorithm for controlling congestion proposed by Sergiou et al. is Hierarchical Tree Alternative Path (HTAP) [6]. When congestion arises, an alternative path selection is created based on local information.

III. LINEAR PROGRAMMING

The reduction of network throughput in multi-hop wireless mesh networks is a significant drawback due to the increase of dropped packets. Many studies have formulated the problem of finding the maximum achievable throughput as a linear programming (LP) problem in order to determine the optimal wireless network throughput.

In this work we use the following LP formulation that is similar to the one proposed by Jain et al. [2]. The notation is illustrated in Table I.

\[
\max \sum_{k=1}^{K} f_k
\]

subject to

\[
f_k = \sum_{l \in s_k} f_{lk}, \text{ for all } k
\]

\[
\sum_{l : l \neq s_k} f_{lk} = \sum_{l : l \neq d_k} f_{lk}, \text{ for all } k \text{ and } i \neq s_k, d_k
\]

\[
\sum_{l \in s_k} f_{lk} = 0, \text{ for all } k = 1, \ldots, K
\]

\[
\sum_{l \in d_k} f_{lk} = 0, \text{ for all } k = 1, \ldots, K
\]

\[
\sum_{l \in I} \lambda_I C_{ij} \leq 1
\]

\[
\lambda_I \geq 0, \text{ for all } I \in \mathcal{I}
\]

\[
f_k \geq 0, \text{ for all } k = 1, \ldots, K
\]

\[
f_{ij} \geq 0, \text{ for all } k = 1, \ldots, K \text{ and all } l_{ij}
\]

The objective function is to maximize total throughput. The constraint (2) states that the total flow from \(s_k\) to \(d_k\) is the sum of the flow amounts of flow \(k\) on the links leaving the source \(s_k\) of flow \(k\). The third constraint expresses flow conservation, i.e., the incoming amount of flow \(k\) must equal to the outgoing amount of flow \(k\) at every node except \(s_k\), \(d_k\). Constraint (4) ensures that there is no incoming flow \(k\) in source \(s_k\), and constraint (5) ensures that there is no outgoing flow \(k\) from destination \(d_k\). The set \(I\) represents a feasible set of links, i.e., a set of links that can transmit at the same time without interference. The physical interference model is used to determine the feasible link sets \(I\) for a given scenario. The SINR at all receivers of the links in \(I\) must be above threshold for \(I\) to be feasible. \(\mathcal{I}\) contains all feasible link sets. The amount of flow on a link cannot exceed the usable capacity of the link as stated by constraint (6). The usable capacity is the full capacity \(C_{ij}\) of the link multiplied with the fraction of time during which it is used. Constraint (7) ensures that the sum of \(\lambda_I\) over all feasible link sets \(I\) does not exceed one. We have implemented a simple tool for generating some types of wireless network configurations by producing node positions and transmission powers, and a second tool that reads such a network configuration and generates the corresponding LP. Then, the LP solver \(lp\_solve\)
was used to solve the LP. The flow rates $f_k$ provided by solving the LP have then been used to adjust the flow rates in an NS2 simulation of the same network configuration, in order to study the effect on the network throughput.

Table I: Explanation of Notation.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K$</td>
<td>Number of flows</td>
</tr>
<tr>
<td>$s_k$</td>
<td>Source of flow $k$</td>
</tr>
<tr>
<td>$d_k$</td>
<td>Destination of flow $k$</td>
</tr>
<tr>
<td>$f_k$</td>
<td>Total amount of flow $k$</td>
</tr>
<tr>
<td>$F_k$</td>
<td>Amount of flow $k$ through $l_{ij}$</td>
</tr>
<tr>
<td>$c_{ij}$</td>
<td>Capacity of link $l_{ij}$</td>
</tr>
<tr>
<td>$I$</td>
<td>Feasible link set</td>
</tr>
<tr>
<td>$\lambda_I$</td>
<td>Fraction of time when link set $I$ is active</td>
</tr>
<tr>
<td>$\mathcal{I}$</td>
<td>all feasible link sets</td>
</tr>
<tr>
<td>SINR</td>
<td>Signal-to-Interference-plus-Noise Ratio</td>
</tr>
</tbody>
</table>

IV. FORWARDING DELAY SCHEME

First of all, a brief description of the data link layer in the wireless network protocol stack will be given and then the idea of the forwarding delay scheme will follow. The data link layer has two sub-layers, the Link Layer (LL) and the Medium Access Layer (MAC), Figure 2 shows that an outgoing packet from LL must wait at the IFQ before being sent out onto the channel via the MAC layer. In addition, before transmitting the packet, the MAC layer senses the channel; if the channel is idle then the packet will be transmitted. Otherwise, the packet will be queued until be sent. Moreover, if the queue is full the incoming packet will be dropped.

Our idea is now to introduce a small delay at forwarding nodes in multi-hop networks for each received packet. Every intermediate node adds a small delay to a data packet at the Network layer instead of immediately passing it down to the link layer. This delay increases the chance for the IFQ buffer between LL and MAC to have space for a new packet to be forwarded. Therefore, this delay reduces the number of packets that get dropped because of a full queue. For example, assume the introduced delay at a forwarding node is $d$ seconds and two packets arrive at the network layer at time $t_1$ and $t_2$, respectively. Consequently, the forwarding node will delay the two packets until time $t_1 + d$ and $t_2 + d$, respectively. As a result, this delay will increase the chance to have space in the IFQ buffer between LL and MAC. Therefore, the network throughput will improve.

V. PERFORMANCE EVALUATION

The performance of LP-based flow rate adjustments and the forwarding delay scheme have been evaluated in the NS2 simulator. The LP solution provides the traffic flow rate that can be transmitted through one channel. We added constraints to the LP ensuring that the flow rate of every flow is less than or equal to 2.25 Mbps. In addition, to reduce the interference at consecutive links, a multi-channel setting with two channels was used. Essentially, this means that intermediate nodes receive a packet through one channel and then transmit the packet on the second channel.

A. Simulation Model

The simulation assumes a static wireless mesh network with nodes that have IEEE 802.11 radios. Every node is equipped with two radio interfaces and uses two channels. The AODV routing protocol has been used. Table II shows the simulation parameters.

In Scenario A, the simulation was run for a chain network in two different settings: one with standard AODV with 2.25 Mbps as flow rate for each flow, and one with standard AODV but with flow rates adjusted according to the LP solution. In Scenarios B–D, the simulation was run four times for a complex network: The first run used standard AODV with 2.25 Mbps for each flow, and the second run used standard AODV with flow rates adjusted according to the LP solution. The third and fourth simulations use AODV with forwarding delay scheme (AODV-F) for both 2.25 Mbps flow rate and flow rates adjusted according to the LP solution, respectively. These simulations have been carried out using UDP traffic. In addition, we have also carried out simulations with TCP and compared the performance to that observed in the UDP simulations.

Table II: Simulation Parameters.

<table>
<thead>
<tr>
<th>Simulation Parameters</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAC type</td>
<td>IEEE 802.11b</td>
</tr>
<tr>
<td>Routing protocol</td>
<td>AODV</td>
</tr>
<tr>
<td>Transmission range</td>
<td>250m</td>
</tr>
<tr>
<td>Interference range</td>
<td>550m</td>
</tr>
<tr>
<td>Traffic type</td>
<td>CBR</td>
</tr>
<tr>
<td>Packet size</td>
<td>1000 bytes</td>
</tr>
<tr>
<td>Channel bandwidth</td>
<td>5.5 Mbps</td>
</tr>
</tbody>
</table>
B. Scenario A

We have examined the effect of flow rate adjustment in a basic chain network with 5 nodes as shown in Figure 3. The examination was done for two cases: a) flows in the same direction. b) flows in different directions (bidirectional traffic).

1) Scenario A-1: In this scenario, the network has two flows, one from node 0 to node 4, and one from node 1 to node 3. The traffic rate after solving the LP is 0.33 Mbps for the flow from 0 to 4 and 2.25 Mbps for the flow from 1 to 3.

As shown in Figure 4, the LP-based flow rate adjustment yields 20% better total throughput than TCP and using a 2.25 Mbps flow rate for each flow. The improvement of throughput in the LP-adjusted flow rate can be explained by dramatic reduction of the number of dropped packets at the IFQ to 0 as Figure 5 illustrates. Although TCP showed similar amount of dropped packets reduction at IFQ, the LP-based flow rate enables a much higher network throughput.

2) Scenario A-2: This section investigates the benefit of the LP-based flow rate adjustment for bidirectional traffic. The network has two flows: one from node 4 to node 0, and one from node 1 to node 3. As shown in Figure 6, the adjusted flow rates again led to better total throughput compared to that observed with flow rate 2.25 Mbps for each flow and TCP. Figure 7 illustrates that although the LP-based flow rate adjustment revealed an immense reduction of the number of dropped packets at IFQ compared to that of 2.25 Mbps, the TCP showed the lowest amount of dropped packets at IFQ. The observed improvement in the LP-based flow rate adjustment in comparison to the TCP is perhaps due to the fact that the TCP reduces the transmission rate when the network becomes congested.

C. Scenario B

We now consider a network with 7 nodes as shown in Figure 8. The performance of LP-based flow adjustments
and the forwarding delay scheme has been examined in two cases: a) Two flows with the same destination b) Two flows with the same source. The two scenarios are referred to as Scenario B-1 and Scenario B-2, respectively.

1) Scenario B-1: The network in Figure 8 is used with 2 flows: nodes 0 and 1 are the sources, and node 3 is the destination. The LP solution for this scenario achieves 3.87 Mbps total throughput with flow rate 1.62 Mbps for the flow from 0 to 3 and flow rate 2.25 Mbps for the flow from 1 to 3.

We have tried out a range of delay values to find an optimal delay for packet forwarding. The results show that among the considered values the optimal delay in this scenario is 0.6s, yielding the maximum throughput in Figure 9. This optimisation achieves almost twice the throughput of immediate forwarding. Then, we have applied the 0.6s forwarding delay to the different flow rate scenarios (flow rate 2.25 Mbps and LP-based flow rates). Figure 10 shows that the forwarding delay leads to a significant increase in the total throughput in both flow rate scenarios. The throughput is about 100% better than immediate forwarding. This significant improvement is perhaps due to the dramatic decrease in the number of dropped packets at IFQ as shown in Figure 11. Although the TCP has the lowest number of dropped packets at IFQ, it has lower throughput in comparison to the forwarding delay scheme. The TCP throughput reduction may be because of the decrease of the transmission rate. It can be concluded that adjusting the flow rate in combination with an optimal forwarding delay will improve the network performance significantly.

2) Scenario B-2: The network in Figure 8 is simulated with 2 flows. Node 0 is the common source, and node 2 and 3 are the destinations. The LP solution for this scenario has 2.75 Mbps total throughput with a flow rate of 2.25 Mbps for the flow from 0 to 3 and a flow rate of 0.5 Mbps from 0 to 2.

First of all, we examined a range of forwarding delays to find an optimal delay for forwarding packets in this scenario, as shown in Figure 12. The optimum delay in this scenario is 0.09s, which gives almost twice the throughput of immediate forwarding. Then, we have applied this delay of 0.09s to the different cases of setting the flow rates (flow rate 2.25 Mbps and LP-adjusted flow rates). Figure 13 illustrates that forwarding delay 0.09s leads to better throughput than immediate forwarding. Moreover, the combination of LP-adjusted flow rate and forwarding delay could maximize the total network throughput because the number of dropped packets at the IFQ is 0, as shown in Figure 14.
D. Scenario C

The aim of studying this scenario is to evaluate the congestion control scheme in a network with high congestion. We have introduced 3 flows in the network shown in Figure 8. The first two flows are from node 3 to node 1 and to node 5, and the third flow is from node 2 to node 0. Solving the LP yields a flow rate of 0.815 Mbps for the flows from node 3 to 5 and from node 2 to 0, while the flow rate from 3 to 1 is 2.25 Mbps.

Figure 16 shows that controlling the network congestion by adjusting the flow rate according to the LP achieves better throughput than using a fixed flow rate of 2.25 Mbps. In addition, the optimum forwarding delay is 0.6s for this scenario, as Figure 15 illustrates. Figure 17 shows that TCP has the lowest amount of dropped packets at IFQ. The observed improvement of throughput in the combination of the LP-based flow rate adjustment and forwarding delay in comparison to the TCP is perhaps due to the fact that the TCP reduces the transmission rate when the network becomes congested. The combination of LP-based flow rates and forwarding delay reduces the number of dropped packets at IFQ, as shown in Figure 17.

E. Scenario D

To construct the LP and solve it using an LP solver, the network needs to have a relatively small number of links as the number of feasible link sets, and thus the size of the LP, can be exponential in the number of links. In this Section we consider a larger network and study the effect of using only the forwarding delay scheme, but not the LP-adjusted flow rates. The network consists of 20 nodes with randomly generated positions. There are four traffic flows with random sources and destinations. Each flow has a flow
rate of 2.25 Mbps. Figure 18 illustrates that the introduction of forwarding delay yields better total throughput than immediate forwarding, especially with the delay values of 0.01s. Figure 19 shows that the forwarding delay scheme achieves better throughput in comparison to TCP and fixed 2.25 Mbps. While TCP has the lowest amount of dropped packets at IFQ as shown in Figure 20, it has additionally the lowest throughput in comparison to the forwarding delay and immediate forwarding. This is perhaps due to the fact that the TCP reduces transmission rate when the network becomes congested. It can be concluded that an optimal forwarding delay will improve the network performance.

Figure 18: Finding Optimum Delay for 20-Nodes Random network.

VI. CONCLUSION

In this paper, we have addressed the problem of packet drops at the IFQ in two stages. Firstly, we have adjusted the traffic rates in order to avoid the congestion occurring due to heavy traffic load. As shown in the chain network, the adjusted flow rates improve the network throughput. Secondly, we have designed a simple forwarding delay scheme to address the congestion due to buffer overflow that arises in more complex network topologies. Combining an optimal forwarding delay and adjusted flow rates can improve the network performance. Our simulation has confirmed that there is a significant throughput improvement and a dramatic reduction of dropped packets at IFQ. For future work, it would be useful to find an alternative approach to constructing and solving the LP for larger networks (for example, using techniques such as column generation), in order to be able to apply the approach to networks with a larger number of links and flows. The effects of flow rate adjustments and forwarding delays can then be examined after solving the large LP in simulations or test-bed experiments.

ACKNOWLEDGMENT

This work was supported by Najran University - Ministry of Education in Saudi Arabia (Grant no. NJU49).

REFERENCES


