A Flow Level Model for Wireless Multihop Ad Hoc Network Throughput

Tom Coenen, Hans van den Berg, Richard J. Boucherie

1 Department of Applied Mathematics, University of Twente, The Netherlands.
2 Department of Computer Science, University of Twente, The Netherlands.
3 TNO Information and Communication Technology, The Netherlands.

Abstract

A flow level model for multihop wireless ad hoc networks is presented in this paper. Using a flow level view, we show the main properties and modeling challenges for ad hoc networks. Considering different scenarios, a multihop WLAN and a serial network with a TCP-like flow control protocol, we investigate how capacity is allocated between the users of a network. This leads us to two different Processor Sharing models, BPS and DPS, which are discussed and compared. Simulation is used to validate the proposed models. The flow level view leads to new insights into the capacity of ad hoc networks, opening new opportunities for analyzing this type of network.

KEYWORDS: Ad hoc network, IEEE 802.11, multi-hop, capacity-allocation

1 Introduction

Ad hoc networks are characterized by their multihop wireless connections and lack of infrastructure. This multihop property of the network presents new challenges to make this type of network effective. For instance, whenever a transmission is started between two nodes in the network, other nodes in the neighbourhood are not allowed to transmit, otherwise a collision would occur. This problem of interference at the MAC layer does not occur in wired networks and raises questions about the impact on the performance of ad hoc networks. A commonly used MAC protocol is IEEE 802.11, which uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA).

The specific characteristics of multihop ad hoc networks calls for new models to analyze the performance. In this paper important performance measures, like the system throughput and the transfer time of flows, are investigated. The allocation of the capacity over the different network nodes plays an important role in this.

We extend the successful approach for analyzing flow transfer times in (single hop) WLANs as presented in [1] to multihop networks. Two different network scenarios are considered. In the first scenario flows may have different path lengths (in terms of number of hops), but follow disjoint routes. The other scenario deals with a serial network in which multiple flows may travel through a particular node. Considering the capacity allocation in both scenarios, we propose two processor-sharing (PS) models for describing the behaviour of the network at flow level. The first model, called Batch Processor Sharing (BPS), deals with a queueing system where batches of jobs arrive. All jobs in the BPS model are served at the same time and are given an equal share of the capacity of the server. In the second model, called Discriminatory Processor Sharing (DPS), jobs arrive one by one and all jobs in the queue are served at the same time, but some jobs get a bigger share of the capacity of the server than others. The batch sizes in the BPS model and the capacity shares in the DPS model reflect the different path lengths of the flows in the ad hoc network scenarios. The modeling results are compared with results obtained by simulation.

The rest of this paper is constructed as follows. In the remaining part of this section, we will give a review of related literature on the subject. Next, in Section 2 the IEEE 802.11 protocol is described. Section 3 presents the two main ad hoc network scenarios under consideration, and investigates the distribution of the capacity over the users in the network. The resulting processor-sharing models for analyzing flow transfer times are shown in Section 4. These models are validated by simulation in Section 5. Finally, Section 6 summarizes and concludes the paper.

*Corresponding Author: Department of Applied Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands; e-mail: t.j.m.coenen@utwente.nl, phone: +31 53 489 3940
1.1 Literature review

Many papers have been devoted to the capacity and throughput of wireless (multihop) networks. Most of them use results from simulation to describe the characteristics of ad hoc networks, whereas analytical studies are scarce. The impact of MAC layer interference on the capacity of ad hoc networks as addressed in this paper has been studied in several settings. For instance [2] uses simulation to show that capacity can be very low in ad hoc networks. Scaling appears only to be possible if the distance between the source and destination remains small as the network grows. An analytical approach for determining the capacity is presented in [3], where it is shown how the throughput depends on the number in the network (when this number becomes large). A paper that focuses more on the multihop property of ad hoc networks is [4], which gives asymptotic results for a wireless network under a relay traffic pattern, whereas [5] considers mesh networks, slightly different from ad hoc networks. Both focus on the throughput of a chain of users processing flows in one direction over multiple hops. A bottleneck is found which determines the throughput that the network can achieve.

In the work of Litjens et al. [1], an integrated packet/flow level approach is used to analyze flow transfer times in a single hop WLAN scenario. Considering, the system throughput at the packet level, and taking the system dynamics at flow level into account, leads to a processor-sharing (PS) type of queueing model for the flow level. This PS model captures the equal allocation of transmission capacity among the active flows. Using known results for this PS model an approximation for the mean flow transfer time is proposed. Simulation results show that the approximation is very accurate.

Modeling bandwidth sharing in fixed communication networks by PS systems has been done by amongst others Bonald and Nunez-Queija. In the papers of Bonald [6],[7], the main notion is that modeling the network with processor-sharing can lead to balanced fairness, which means that each user in the network receives an equal share of the available network resources. This type of PS network is analyzed by considering the bottleneck node and distributing the capacity there first. All nodes servicing the same flow adjust their capacity allocation accordingly, avoiding congestion. The capacity allocated to each flow is determined analytically. In his dissertation, Nunez-Queija discusses many different PS models for integrated services networks [8].

Batch arrival processor-sharing models have been investigated extensively in the past. It was Kleinrock who started with this approach. In his paper with Muntz and Rodemich [9] a start was made in giving a complete analytical approach to determine the throughput of a PS network. A discriminatory processor-sharing model has also been used for modeling networks. Kleinrock [10] already started with this in 1967 which created much interest in this type of network. In 1980, Fayolle, Mitran and Iasnogorodski [11] built on the work of Kleinrock. New results have been obtained in [12],[13] where the queue length distribution and sojourn times for PS models are determined. The results presented hold for general service requirements and are used to analyze WLANs with Quality of Service support [14].

2 IEEE 802.11 MAC Layer Protocol

Ad hoc networks are characterized by their multihop wireless connections and lack of infrastructure. Interference plays an important role in this type of network. Signals transmitted by a user will not only be heard by the receiver, but also by all other nodes in the vicinity of the sender. If multiple signals reach a node at the same time, a collision occurs and the signals cannot be received correctly and are lost. To reduce the number of transmissions that fail and the impact this has on the throughput of the network, IEEE 802.11 uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). IEEE 802.11 can function in infrastructure or ad hoc mode, depending if an access point is being used. This has no implications on the MAC layer.

When a node wants to transmit, it will first listen to find out if other nodes are already transmitting: carrier sensing. If other nodes are transmitting, the node will not transmit. When the network becomes available, the node waits for a certain time (DIFS) and if the network is then still free, a timer is started to avoid collisions. This timer is paused as soon as the node senses a transmission from another node. When the network becomes free again and stays free for a DIFS, the timer continues. When the timer ends, transmission starts. This approach does not make it completely sure that collisions will not occur. Therefore, instead of sending the packets of the data immediately, a node first transmits a request-to-send (RTS). The receiver replies to this RTS by sending a clear-to-send message (CTS). The time between these transmissions (SIFS) is smaller than DIFS. After receiving the CTS, transmission of the data starts. This approach is used so that in case of a collision, only the RTS is lost, and not a much bigger packet containing data. This way the impact of a collision on the throughput of the network is reduced. When a collision occurs, a timer starts again but is set to a time taken from a window that is twice as big as before. When the transmission was successful, the procedure repeats as long as there still are packets that
need to be transmitted. The operation of CSMA/CA is shown in Figure 1.

![CSMA/CA with RTS-CTS](image)

Figure 1: CSMA/CA with RTS-CTS

Under CSMA/CA, all nodes that want to transmit compete for network resources. In the multihop wireless ad hoc network we are considering, packets from a multihop flow are present at multiple nodes. Such a flow is competing for the network resources through multiple nodes at the same time. Hence even if there is only one multihop flow in the network, there is interference between the different users that are involved in the transmission. All flows in the network will have to share the MAC layer capacity, and the capacity allocation over these flows will influence the throughput of the network.

3 Scenarios

In this section, first, the analysis of the single hop WLAN considered in [1] is shortly reviewed, after which two multihop scenarios are described. For the analysis of these scenarios we use a similar approach as used for the WLAN in [1], with the extension of considering the multihop aspects involved in these scenarios.

3.1 Single hop WLAN scenario

In [1] a single hop WLAN is considered, in which new flow transmissions are initiated according to a Poisson process. Flow sizes are random variables with general distributions. The network operates under the IEEE 802.11 DCF MAC protocol as described in Section 2. First an analysis is made on the packet level of the aggregate system throughput that can be reached in a WLAN with a fixed number of persistent flows. Using Markov-chain analysis, the probability that a node is transmitting is computed, as well as the probability that a transmission fails. From this, the aggregate system throughput is derived, including the influence of the headers and control packets. Simulation validates the result that the average system throughput is about 87% of the total capacity on the MAC layer, and slightly dependent on the number of present flows. Next, using the results from the first step, the transfer time is analyzed, taking the flow level dynamics into account. The assumption is made that the service rate per flow is found by giving each flow an equal share of the aggregate data throughput computed for the persistent number of flows in the network. This leads to a processor-sharing (PS) model with state dependent service rates which is analytically tractable. Simulation shows that the results attained through this approach are very accurate.

3.2 Multihop Ad Hoc scenario

The model presented in [1] only considers single hop flows. This model is expanded by also allowing multihop flows. Where at first a flow was completed if the packets were sent from a node to the access point or vice versa, now there is the possibility that all packets are forwarded to another node before completing the transfer.

![Multihop scenario](image)

Figure 2: Multihop scenario
A transmission of any of the nodes in the cell can be heard by all other nodes in the cell. This means that no two transmissions can take place at the same time, since the data will be lost due to a collision. The situation in Figure 2 is a WLAN cell in ad hoc mode with connections of only one or two hops. A second hop is then used to connect to a user outside the cell. Whenever a node is relaying a flow, this node will only compete for the network resources if there are packets available that need to be sent. If at one point there are no packets available, the node will remain idle until new packets arrive that need to be forwarded.

Following the approach presented in [1], we first determine the aggregate system throughput of the network. The number of persistent users is taken to be the number of nodes active in sending the flows. This means that a flow over two hops, which needs two nodes, is counted as two users. However, there might be moments that the relaying nodes are not competing for the network since there are no packets available. This differs from the single hop WLAN situation described above where each node always has packets that need to be transmitted. Through simulation the aggregate system throughput is determined.

To take the flow level dynamics into account, the capacity allocation needs to be known. As in the WLAN model, we assume that every node receives an equal share of the aggregate system throughput. In general this is the case since all nodes behave according to the IEEE 802.11 protocol. For the flow level dynamics, we assume that flows are big enough to have packets available at all nodes they are going through, for most of the time. Then these nodes are continuously competing for the network and all nodes get an equal share of the capacity. A flow over two hops will get a share of the capacity for both the nodes it uses. Hence the capacity allocated to a flow over two hops will be double the amount of the capacity allocated to a flow over one hop, but note that each packet has to be sent twice. Just as for the WLAN model, this approach leads to a processor-sharing type of model which will be discussed in Section 4.

3.3 Multihop serial network scenario

A different scenario is where nodes can serve more than one flow at the same time. Assuming that a flow will at most need two hops to reach its destination, such a network can be modeled as a network consisting of three nodes, with two connecting links. This network is represented by the model shown in Figure 3.

![Serial model](image)

Figure 3: Serial model

The flows through the wireless medium (the links), represented by the arrows through the tubes, will compete for the channel. There are three types of flows. Flows of type 0 will go over both links (as depicted by the lower arrow going through both tubes); flows of type 1 (2) will only use link 1 (2) (the upper left (right) arrow going through the left (right) tube). All flows consist of packets which first arrive in a buffer before being sent over the wireless medium. Because of interference, the links have to share the MAC layer resources.

Assume that the arrival process of the flows is according to a Poisson distribution and the flows consist of many packets, so that flows over multiple hops will again usually have packets in the queue of every node it passes. The packets of all flows join the same queue at a node. With all flows arriving simultaneously, the packets will be in the queue in a mixed order and are served according to a FCFS discipline. Hence all flows are serviced, packet by packet, in a more or less random order, as shown in Figure 4.

This way of servicing is approximated by a processor-sharing (PS) service discipline for the flows, where each link is assumed to have the same capacity. Just as in the previous scenarios, the aggregate throughput needs to be determined, which is done through simulation.

An important aspect regarding the throughput of the different type of flows in the network is how the capacity is allocated among them. This depends on the flow control protocol used in the network. A commonly used transport protocol in (wired) networks is TCP. This protocol tries to avoid congestion by fairly sharing the available resources among active flows. A similar protocol can be used in wireless networks. Suppose that in the reference model there are $n_0$, $n_1$ and $n_2$ flows of type 0,1 and 2 respectively. At first, the available capacity will be shared in a fair way
over both nodes, each node receives half of the total capacity. Both nodes will then use this capacity to process the flows that are in their queues. This is assumed to be done so that each flow gets an equal share, which is called egalitarian processor-sharing. Now the situation can occur that flows of type 0 get a different amount of the capacity over the first link than over the second link, as will be discussed later. If the capacity at the first link is higher, the queue at the second server will build up, since it cannot serve the flow as fast as it arrives. If the capacity at the first link is lower, then the queue at node two will often not contain any packets of the flow of type 0, since these are processed faster than the rate at which they arrive. These are unwanted situations, and so the flow control protocol will notify the source to transmit at different rates for the flows that are causing the congestion. We assume a TCP-like flow control protocol to be active, which alters the use of capacity in case of queues building up or being empty most of the time. The flow control protocol resembles TCP, but assumes a perfect version with instantaneous rate adaptation, so it will be e.g. independent of the round trip times (RTT) of the flows.

In our model there can be a loss of packets due to a build up in the queue of node 2. This happens if the flows of type 0 get more capacity in node 1 than in node 2 for a long time. This is for example the case when there are flows of type 2, while there are no flows of type 1 in the system and both links get half of the total capacity. If the flows of type 0 keep being processed at this rate, packets will be lost. However, the flow control protocol will make sure that the rate at which packets are transmitted over link 1 is lowered. The rate to lower it to is the rate at which the flow is processed at node 2. The capacity used by node 1 hence drops due to the lowering of the transmission rate. This capacity can then be used by node 2.

**Theorem 1** The capacity a flow receives at a link is equal for any type of flow at any link, namely $\frac{1}{n_0+n_1}$ when there are only flows of type 0 and one other type (i).

**Proof.** Consider the network where only flows of type 0 and 1 are present. Let there be $n_0$ ($n_1$) flows of type 0 (1). If both links get half of the total capacity, which we set to be one, then the capacity allocated to a flow at node 1 will be equal to $\frac{1}{2n_0+n_1}$. At node 2, the flows of type 0 will receive a capacity of $\frac{1}{2n_0+n_1}$. If this situation persists, the queue at node 2 will often be empty, since there the flows of type 0 are processed faster than the rate at which they arrive. The flow control protocol therefore lowers the rate at node 2 and sets the capacity of a flow to $\frac{1}{2n_0+n_1}$. Node 1 can now use the residual capacity and a flow will get a capacity of $\frac{1}{n_0+n_1}(1-\frac{1}{2n_0+n_1})$. This rate however is higher than the rate at node 2 and so the buffer will fill. The rate at node 2 is adjusted by the flow control protocol to $\frac{1}{n_0+n_1}(1-\frac{1}{2n_0+n_1})$, which leaves a capacity for a flow at node 1 of $\frac{1}{n_0+n_1}(1-\frac{n_0}{n_0+n_1}(1-\frac{1}{2n_0+n_1}))$. This process continues (where each step is instantaneous by assumption) and it can easily be seen that this converges to a capacity allocation of $\frac{1}{2n_0+n_1}$ for each flow on any link. In total, a flow of type 0 will receive $\frac{1}{2n_0+n_1}$ of the capacity, whereas a flow of type 1 will get $\frac{1}{2n_0+n_1}$. The proof for the situation with only type 0 and type 2 flows follows in the same manner. ■

If however there are flows of both type 1 and type 2 in the system, this situation will not occur. Supposing that there are more flows of type 2 than of type 1, a flow of type 0 will get a rate of $\frac{1}{n_0+n_1}$ at node 2, but receives a higher rate of $\frac{1}{2n_0+n_1}$ at node 1. The protocol will hence lower the rate for all type 0 flows to $\frac{1}{n_0+n_1}$ at node 1. The capacity that now becomes available is not given to node 2, but the flows of type 1 will claim this extra capacity since the node has the right to use half of the total capacity. This is a different and interesting situation for further research, but we will not consider it any further in the analysis in this paper.

We see that in the first situation the flow control protocol leads to the situation in which all flows get the same share of capacity per link. Hence the total capacity, a flow over two links will get twice as much capacity as a flow over one link. This is equivalent to what we found for the two-hop WLAN model. Hence we need to analyze

![Figure 4: Processor sharing on flow level](image-url)
the same situation, which as noted before leads to a processor-sharing model, which will be discussed in the next section.

The analogy between the scenario as shown in Figure 2 and the scenario of Figure 3 can be seen as follows. A flow over two hops coincides with a flow of type 0 and flows over one hop coincide with a flow of type 1 or 2 (but not both), which is arbitrary. In the serial network, just as in the two-hop WLAN scenario when a node relays, a flow taking the second hop will only compete for the network resources if it has packets available to be sent. If at one point no such packets are available in the buffer, no capacity is allocated to this flow at that node.

4 Flow level models

This section presents models that approximate the flow level dynamics of the multihop scenarios presented in the previous section. For the multihop ad hoc scenario, the equal share given to each of the nodes determines the capacity allocation. For the serial scenario, the transmission control protocol determines the capacity allocation at the MAC layer. This section presents two analytical models that capture the flow level dynamics of both scenarios. The two models take the capacity allocation into account by either varying the amount of jobs in an egalitarian processor-sharing queue or the priority and size of jobs in a discriminatory processor-sharing queue.

4.1 Batch Arrival Processor Sharing model

We can consider the network as a server with one queue, where all flows enter the queue, independent of the link(s) they have to be transmitted over. Since all flows are processed at the same time, we can consider the flows to be processed according to a processor-sharing discipline. As a flow over two hops gets the double amount of capacity, we can consider a flow over two hops as asking for capacity twice. Hence we can see the arrival of a flow over two hops as the arrival of two flows at the same time. A flow over two hops will be at both servers at the same time, so it will ask for capacity as if it were two different flows, which is captured in this abstract view. We thus arrive at a batch arrival processor-sharing model (BPS) with egalitarian processor-sharing, since the capacity for a flow is equal at every hop. A flow over a single hop is then equivalent to a single arrival, whereas an arriving flow over two hops is equivalent to two jobs arriving as a batch. It is important to note here is that we do not need to consider that all jobs in a batch should not only have the same arrival time but also the same departure time, i.e. jobs in a single batch have the same service demand, since they represent only one flow.

Consider the $M^X/G/1$ PS queue where $\lambda$ is the batch arrival rate, $a$ is the average batch size, $b$ is the average number of jobs that arrive in addition to the tagged job and $F(x)$ is the complementary distribution function of the job size. The conditional response time of a job with service requirement $x$, $T(x)$, has to satisfy the system of differential equations ([9]):

$$T'(x) = \lambda a \int_0^\infty T'(y)F(x + y)dy + \lambda a \int_0^x T'(y)F(x - y)dy + bF(x) + 1$$

The load in the system is given by:

$$\rho = \lambda a E[X].$$

When flows have an exponential service requirement, solutions can be found ([15]). For the $M^X/M/1$ PS queue, this leads to:

$$T(x) = \frac{x}{1 - \rho} + \frac{b(2 - \rho)E[X]}{2(1 - \rho)^2} \left(1 - e^{-\frac{\rho}{(1 - \rho)^2}}\right)$$

and bounds are given by:

$$\frac{x}{1 - \rho} \leq T(x) \leq \min\left(\frac{b + 1}{1 - \rho}, \frac{x}{1 - \rho} + \frac{b(2 - \rho)E[X]}{2(1 - \rho)^2}\right)$$

where the bounds coincide when $x^* = \frac{(2 - \rho)E[X]}{b(1 - \rho)}$.

In these models, the departure moments of jobs inside a batch will not be the same. Only when the service times are deterministic, is this model applicable. Therefore, we also propose a more appropriate model.

4.2 Discriminatory Processor Sharing model

A flow over two hops receives more capacity than a flow over one hop, hence we can instead consider the processor-sharing not to be egalitarian. The jobs are then processed at the same time, but not all jobs get an equal share. As
a flow over two hops takes twice the amount of capacity, it can be seen as being serviced twice as fast as a single hop flow. A flow over two hops however has an expected service requirement that is twice the expected service requirement of a single hop flow. We thus arrive at a discriminatory processor-sharing model (DPS). In this type of model, all jobs get processed at the same time, but not all jobs get the same amount of service. Customers are given a certain weight which shows how much more service they receive in comparison to other users. In our case, a job that represents a two-hop flow will get a weight twice as high as a job representing a single-hop flow.

Consider the $M/G/1$ DPS queue where $\lambda_j$ denotes the arrival intensity of class $j$ jobs, $g_j$ denotes the ‘weight’ of class $j$ customers and $F_j(x)$ the distribution of the required service with mean $1/\mu_j$ and a total of $M$ classes. The conditional response time of a job of class $k$, given its size $t$, $W_k(t)$, satisfies the system of differential equations ([10]):

$$W_k'(t) = 1 + \sum_{j=1}^{M} \int_{0}^{\infty} \frac{\lambda_j g_j W_j'(u)}{g_k} (1 - F_j(u + \frac{g_j t}{g_k})) du$$

$$+ \int_{0}^{t} W_k'(u) \sum_{j=1}^{M} \lambda_j g_j (1 - F_j(g_j(t-u)/g_k)) \frac{1}{g_k} du, \quad k = 1...M.$$  

For the $M/M/1$ DPS queue, we know that (for the derivation see [11]):

$$W_k(t) = \frac{t}{1-\rho} + \sum_{j=1}^{m} g_k c_j \alpha_j + d_j (1 - e^{-\alpha_j t/g_k}),$$

where the $\alpha_j$'s are the distinct roots of $1 - \Psi(s) = 1 - \sum_{j=1}^{M} \frac{\lambda_j g_j}{\mu_j g_j + s} = 0$ and $c_j$ and $d_j$ are given by:

$$c_j = (s + \alpha_j)\alpha^*(s)|_{s=-\alpha_j} = \frac{\prod_{k=1}^{m} (g_k \mu_k - \alpha_j)}{-\alpha_j \prod_{k \neq j}(\alpha_k - \alpha_j)}$$

$$d_j = (s^2 + \alpha_j^2)\theta(s)|_{s=-\alpha_j} = \frac{\prod_{k=1}^{m} \lambda_k g_k^2 (\mu_k^2 \mu_k^2 - \alpha_j^2)}{\prod_{k \neq j}(\alpha_k^2 - \alpha_j^2)}.$$

Since each flow is represented by only one job in the system, the departure of a job is equivalent to the departure of a flow from the network. Therefore, this approach gives a better approximation of the situation that we want to model. When considering deterministic service requirements, we see that both models give equivalent results.

5 Numerical results

To verify that the proposed model is an accurate approximation of the network under consideration, a simulation model has been constructed to obtain data on the sojourn time of a flow in the network. The simulation model mimics the transmissions as they occur in the scenario depicted in Figure 2. The simulation model uses the following standard settings for the parameters:

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
<th>parameter</th>
<th>value</th>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHY</td>
<td>192 bit</td>
<td>Payload size</td>
<td>12 kbit</td>
<td>SIFS</td>
<td>10 µs</td>
</tr>
<tr>
<td>MAC</td>
<td>272 bit</td>
<td>r_net</td>
<td>1 Mbit/s</td>
<td>DIFS</td>
<td>SIFS + 2τ</td>
</tr>
<tr>
<td>RTS</td>
<td>PHY + 160 bit</td>
<td>n_max</td>
<td>100</td>
<td>c_Wmin/max</td>
<td>31/1023</td>
</tr>
<tr>
<td>CTS</td>
<td>PHY + 112 bit</td>
<td>δ</td>
<td>1 µs</td>
<td>r*</td>
<td>5</td>
</tr>
<tr>
<td>ACK</td>
<td>PHY + 112 bit</td>
<td>τ</td>
<td>20 µs</td>
<td>r_max</td>
<td>6</td>
</tr>
</tbody>
</table>

Here $r_{net}$ is the rate at which the network can transmit data, $n_{max}$ is the maximum number of users, PHY, MAC, RTS, CTS and ACK give the sizes of the headers and interframe spaces, $δ$ is the propagation delay, $τ$ is the slot duration, $c_w$ are the values for the contention windows, $r^*$ is the maximum number of times the contention window may be doubled and $r_{max}$ is the maximum number of retransmissions for one packet. The payload size is set at 12 kbit, the maximum amount of data that can be sent within a packet. The probability that a flow is over two hops is given as input to the simulator. All flows are either single or double hop flows. Files arrive at the system according to a Poisson process, where users try to transmit the packets according to the IEEE 802.11 protocol discussed earlier. When the first packet of a double hop flow has been sent over the first hop, a free user is assigned.
as the relaying node and this user will also start transmitting the packets that it receives from the first user. When
the second user has no packets in its queue to relay, it will go into waiting, meaning that he will not compete for
the channel. The DCF function of IEEE 802.11 is incorporated in the simulation, where collisions are considered
to be fatal, meaning that all packets involved in the collision are lost and retransmitted after backing off.

The results of the simulation and the M^X/M/1 BPS and M/M/1 DPS model are compared, we hence are
considering Poisson arrivals and exponentially distributed file sizes. The model considers the network in a situation
that the full capacity of the network can be used for the files, which is not the case for the simulation program.
Here headers are added to all the packets and the number of active users influence the aggregate system throughput
as described in [1]. Following their approach, the aggregate throughput for persistent flows is determined. For the
determination of this aggregate throughput the amount of double hop flows can be of influence. The interference
that the flow causes for itself deteriorates the throughput of the network. However, the second node in a double
hop flow may not always have packets to transmit, at which point it will not cause interference. Simulation is used
to compare the results of single and double hop flows. In the following figure, the aggregate system throughput
is computed for single hop flows, and compared with the aggregate system throughput of double hop flows, where
the number of persistent flows is taken to be twice the amount of double hop flows.

Figure 5: Aggregate system throughput

Figure 5 clearly shows that the aggregate throughput is hardly influenced by double or single hop flows. This
shows that we can use the results for single hops, but counting the double hop flows as if there are two users in the
system.

Under the RTS/CTS mode, the aggregate throughput is roughly constant as was also found for the WLAN
situation in [1]. As can be read from the figure only about 88% of the capacity can really be used. This is taken
into account in the calculation of the average transfer time in the models. First we compare the average transfer
time of a job in the system with the results from the BPS model. A drawback of this model is that it is not possible
to make a distinction between the single and double hop flows in this model. Results are shown in Figure 6a for the
situation where 30% and 70% of all the flows are double hop flows and the file sizes are exponentially distributed
with a mean of 150 kbytes.

Figure 6: BPS versus simulation

The figure shows that for a lower amount of double hop flows, the approximation is better. When 70% of all
the flows are double hop flows, the difference becomes bigger. Next, the comparison is made between simulation
and the DPS model, where we can distinguish between the different type of flows, which is shown in Figure 6b and
6c.
The approximations are accurate, independent of the amount of double hop flows in the system. It can be seen that for a higher load the approximation is slightly worse. Interesting is to see that the model overestimates the transfer time for single hop flows, whereas is underestimates the transfer time of double hop flows. This is due to the fact that the assumption that a double hop flow receives twice the capacity of a single hop flow is not exact, since packets are not always available at all the different nodes on a multihop path. Therefore the capacity allocated to double hop flows is slightly less than the double of single hop flows, resulting in the differences seen in the figures.

6 Conclusion

Using a flow level approach for modeling an ad hoc network, many difficulties might be avoided that would occur when using a packet level view. Results from the past have shown that simulation is often the only possible approach to get results and for analytical approaches to be possible many assumptions have to be made. However, the flow level view has a promising future, even for analytical approaches.

This paper proposes considering the network from a flow level point of view, with two types of models for approximating the throughput of such a network, namely the BPS and DPS models. The $M^X/G/1$ PS and $M/G/1$ DPS queues can be used, since these models take the allocation of the capacity over the different flows in the network into account. Much work on these types of models has already been done, and some analytical results have been presented. When considering the network as a BPS queue, the problem arises that all flows inside a batch should have the same service requirement, which is not the case for the $M^X/G/1$ PS queue, making it impossible to distinguish between different classes of flows. Therefore, it is more accurate to use the $M/G/1$ DPS queue for modeling the ad hoc network.

The flow level view as presented in the paper opens new opportunities for modeling ad hoc networks, taking into account interference at the MAC layer, especially self-interference within a flow. Through simulation the model has been validated, showing that the results obtained using the $M/M/1$ DPS queue gives a good approximation of the transfer time in an ad hoc network using IEEE 802.11 in RTS/CTS mode, independent of whether many or few flows are double hop flows. For future research it will be interesting to investigate the impact of transmission ranges and file sizes on the performance of the network and the modeling approach presented in this paper.

References


