Report

Computing throughput capacity for realistic wireless multihop networks

Author(s):
Stuedi, Patrick; Alonso, Gustavo

Publication Date:
2011

Permanent Link:
https://doi.org/10.3929/ethz-a-006781951

Rights / License:
In Copyright - Non-Commercial Use Permitted
Computing Throughput Capacity for Realistic Wireless Multihop Networks

Patrick Stuedi and Gustavo Alonso

Department of Computer Science
Swiss Federal Institute of Technology (ETH Zurich)
8092, Zurich, Switzerland
{stuedip, alonso}@inf.ethz.ch

ABSTRACT

Capacity is an important property for QoS support in Mobile Ad Hoc Networks (MANETs) and has been extensively studied. However, most approaches rely on simplified models (isotropic radio propagation, unidirectional links, perfect scheduling, perfect routing, etc.) and either provide asymptotic bounds or are based on integer linear programming solvers. In this paper we present a probabilistic approach to capacity calculation by linking the normalized throughput of a communication pair in an ad hoc network to the connection probability of the two nodes in a so called schedule graph \( G_T(N, E) \). The effective throughput of a random network is modelled as a random variable and expected values of it are computed using monte-carlo methods. A schedule graph \( G_T(N, E) \) for a given network directly emerges from the physical properties of the network, like node distribution, radio propagation or channel assignment. The modularity of the approach allows for capacity analysis under more realistic network models. In the paper this is demonstrated by showing that throughput capacity increases in the presence of randomized radio propagation, even in the case of acknowledgement-based communication and decreasing network connectivity.

1. INTRODUCTION

Capacity is typically studied by choosing a network model that facilitates analytical treatment. In doing so, the problem has to be simplified by either making assumptions about the network (e.g. symmetric links), radio propagation (e.g. isotropic signal propagation) or the size of the network (e.g., very large number of nodes). In this paper, we eliminate many of these restrictions by looking at throughput capacity from a probabilistic perspective. Since capacity of random networks must be random as well, we model the achievable throughput per communication pair in a multihop wireless network as a random variable. The approach is centered around a so called schedule graph \( G_T(N, E) \) which is directly derived from the physical properties of the network. The effective throughput capacity of a pair of nodes in an ad hoc network is then shown to be related to the connection probability of these two nodes in \( G_T(N, E) \). Due to its modularity, our approach is decoupled from specific network properties such as, e.g., the channel multiplex schema, the signal propagation and interference model, the routing or the node distribution. In that sense, our approach can be seen as a powerful tool to analyze any form of interaction between physical and logical properties of the network with regard to throughput capacity. In the paper we demonstrate this using various examples. For instance, we show that throughput capacity increases in the presence of randomized radio propagation, even in the case of acknowledgment-based communication and decreasing connectivity.

1.1 Related Work

Most existing work on capacity assumes a network to have \( n \) nodes distributed within a certain area and defines a packet transmission between two nodes to be successful if its signal-to-noise ratio is bigger than a given threshold. In their seminal work [11], Gupta and Kumar have studied capacity asymptotically for an increasing node density. They have shown that the throughput capacity \( \lambda(n) \) for a network of \( n \) nodes within an area of \([0,1]^2\) is in the order of \( \Theta(1/\sqrt{n \log n}) \). This result was extended for models including variable transmission power [9], bound attenuation functions [8] and multiple channels [13]. While asymptotic bounds certainly indicate the generic behavior of ad hoc networks for large \( n \), they do not give any information on concrete throughput capacity and small networks. Recently, there has been some effort to compute concrete throughput values [4, 6, 20] using integer linear programming (ILP). However, ILP makes it very difficult to model physical network properties such as realistic signal propagation, link asymmetry or interference. Hence, most of these studies are based on a simplified network model. For instance, it is common to predict the received power as a deterministic function of distance, thereby representing the communication range as an ideal circle. In reality, the received power at a certain distance is a random variable due to fading effects. Effects of shadowed radio propagation on capacity have also been analyzed [22] but without considering multihop networks. In such networks, any variation in the signal pattern impacts the perceived interference at a given node. Non-deterministic variation of signal power may further lead to link asymmetry. This behavior was measured experimen-
In [2], IEEE 802.11, the MAC protocol often mentioned in combination with ad hoc networks, allows for data transmission only if there exists a bi-directional connection between the two communicating nodes since data packets need to be acknowledged by the receiving node. Effects of asymmetric links on higher network layers were investigated in [21].

1.2 Contribution

The contributions of this paper are as follows:

- The paper presents an abstract model to compute throughput capacity in multihop wireless networks. By combining the model with Monte Carlo methods, the paper proposes a new way of throughput capacity computation for more realistic network configurations with complex random properties. The approach of first transforming the physical properties of the network into a graph representation makes the actual throughput computation independent of low level network details and at the same time facilitates the analysis of various physical and logical effects with regard to throughput capacity.

- As a demonstration of the model and as a value on its own, the paper shows that the throughput capacity increases in the case of randomized radio propagation, even in the case of acknowledgement-based communication and decreasing network connectivity. Further applications of the model would include, e.g., investigations on the complex interaction between routing and scheduling as well as effects of different forms of interference calculation on throughput.

- By linking throughput capacity of multihop wireless networks with the connection probability in a schedule graph \( G_T(N, E) \), the paper proposes a way of analyzing capacity also in sparse and partially disconnected random networks. This might be particularly helpful with regard to throughput calculations in mobile scenarios where the movement of the nodes often leads to temporarily broken paths.

- The paper further presents and discusses an algorithm for a conflict-free channel assignment under arbitrary interference models, including SINR-based interference.

2. NETWORK MODEL

In a first step, we want to turn physical properties of wireless multihop networks into a so called schedule graph \( G_T(N, E) \). Examples of physical properties are node locations or perceived signal strengths.

In a schedule graph, \( N \) is the set of nodes in the network and \( E \) denotes a set of directed edges between the nodes such that the existence of a sequence of nodes \( n_0, n_1, \ldots, n_k \) – with \( n_i \in N, \forall i \leq k \) and \( (n_i, n_{i+1}) \in E, \forall i < k \) – states that there is also a schedule of channel assignments \( \psi(n_0, n_1), \psi(n_1, n_2), \ldots, \psi(n_{k-1}, n_k) \) such that node \( n_0 \) is able to consecutively transmit data to node \( n_k \) at a rate \( \lambda_{n_0,n_k} > 0 \).

The idea behind building a schedule graph is to create an abstraction that allows us to – later on – reason about the achievable capacity of the underlying wireless network. In this section, we first define some common properties in order to then gradually develop the graph representation by assigning three sets \( D_n \supseteq U_n \supseteq V_n \) of nodes to each node \( n \), with \( D_n \subseteq N \). Nodes within the particular sets correspond to the different forms of interaction nodes can have, such as unidirectional and bidirectional communication. A list of all the notation used within the following two sections, including the aforementioned sets of nodes, can be found in Table 1.

We parameterized the network using the following five properties: The set of \( N \) nodes \( N \), a node distribution \( \delta \), a signal propagation \( \vartheta \), a channel assignment \( \psi \) and an interference model \( \kappa \). We assume \( x_n \in \mathbb{R}^2 \) to be the coordinate\(^1\) of node \( n \), identifying the node’s position with respect to an area \( A \) and we consider the set \( N \) of nodes as being distributed in \( A \) according to some probability function \( \delta : A \rightarrow [0, 1] \). Throughout this paper, we use \( \mathcal{P}(\cdot) \) to refer to the collection of all possible subsets of a set.

2.1 Decodables \( D_n \)

Let us start by defining how signals are propagated. First, each node \( n \) is supposed to transmit with a signal power \( P_n \in [0, \infty[ \). For a certain signal propagation \( \vartheta \), \( P_{n\rightarrow w} = \vartheta |x_n - x_w| \) denotes the power of the received signal at node \( n \) due to a transmission of node \( n' \). In the simplest case, \( \vartheta \) is a direct function of the distance.

The path loss radio propagation model, for example, defines \( \vartheta_{pl}(p, d) = p \cdot (d/d_0)^{-\rho} \) for some path loss exponent \( \rho \) and \( d_0 \) as a reference distance for the antenna far-field. A more sophisticated model is the log normal shadowing radio propagation [16]:

\[
\vartheta_{sh}(p, d) = p \cdot (d/d_0)^{-\rho} \cdot 10^{X/10}
\]

where \( X \) is a gaussian random variable with zero mean and standard deviation \( \sigma \) and \( \rho \) is the aforementioned path loss exponent.

In case of \( \sigma \) equal 0, there is no random effect and \( \vartheta_{sh} \equiv \vartheta_{pl} \). We now define a set \( D_n \) as

\[
D_n = \{ n' \in N \mid P_{n\rightarrow n'} > = \beta_D \}
\]

the set of nodes that can be correctly decoded at node \( n \) in the absence of any other concurrent transmission. Typically \( \beta_D \) is a hardware specific constant referring to the minimal signal power that exceeds some thermal noise \( P_n^* \).

2.2 Senders \( U_n \)

Transmissions from a node \( n' \) to another node \( n \) are bound to a set of predefined channels \( \psi(n', n) \), where \( \psi : N \rightarrow \mathcal{P}(\Gamma) \) and \( \Gamma \) is the set of all available channels. We further use \( \psi(n) = \bigcup_{n' \in D_n} \psi(n, n') \) to refer to all the channels assigned to a node \( n \). Two nodes are not allowed to transmit data in any other than their assigned channels. For the sake of simplicity we use the word channel interchangeably for the set of all nodes transmitting data within that specific channel. A clear separation of concurrent transmissions into non-overlapping channels is very hard to achieve in practice and also not very efficient since modern radio receivers tolerate a certain amount of noise from other transmissions while still being able to correctly decode the signal. Whether a node \( n \) is able to correctly decode the signal of another node \( n' \) in the presence of interfering nodes depends on the so called interference model \( \kappa : N \times N \times \mathcal{P}(N) \rightarrow \{0,1\} \) with

\(1\)The model could also be applied to \( \mathcal{R}^3 \)
As an example of an interference model [16], the well known signal to interference plus noise model \( \kappa_{\text{sinr}} \) computes as

\[
\kappa_{\text{sinr}}(n', n, I) = \begin{cases} 
1 & \text{The signal of } n' \text{ can be decoded at node } n \text{ under a set } I \text{ of interfering nodes} \\
0 & \text{otherwise.}
\end{cases}
\]

for some threshold \( \beta_{\text{sinr}} \) and \( P_n^t \) as the aforementioned thermal noise perceived at node \( n \). Of course, for any interference model it must be given that \( \kappa(n', n, \emptyset) = 1 \Leftrightarrow n' \in \mathcal{D}_n \). Based on the notion of \( \kappa \) we define the transmission capacity \( \omega : \mathcal{N} \times \mathcal{N} \rightarrow [0, 1] \) between two nodes \( n \) and \( n' \) as

\[
\omega(n', n) = \sum_{\gamma \in \psi(n', n)} \kappa(n', n, I_{\gamma}).
\]

Here, \( I_{\gamma} \) denotes the set of nodes transmitting during channel \( \gamma \), or \( I_{\gamma} = \{ n' \in \mathcal{N} \mid \gamma \in \psi'(n') \} \). The set of nodes whose signals can be decoded correctly at node \( n \) even in the case of concurrent transmissions can then be written as

\[
\mathcal{U}_n = \{ n' \in \mathcal{N} \mid \omega(n', n) > 0 \}.
\]

### 2.3 Neighbors \( \mathcal{V}_n \) and Schedule Graph

In our model we particularly want to account for acknowledgement based medium access protocols. We therefore define \( \mathcal{V}_n \), the set of all neighbors of \( n \) as follows:

\[
\mathcal{V}_n = \{ n' \in \mathcal{N} \mid n' \in \mathcal{U}_n \land n \in \mathcal{D}_n \}.
\]

The set \( \mathcal{V}_n \) includes all nodes \( n' \) whose signals can be decoded correctly at node \( n \) under concurrent transmissions while being able to correctly receive the acknowledgement sent back to \( n' \). Note that equation 7 models the acknowledgement as an infinite small packet not occupying the medium.

Based on the notion of neighbors, the so called schedule graph is now simply defined as a directed and weighted graph \( G_{T}(\mathcal{N}, \mathcal{E}) \), where \( \mathcal{N} \) corresponds to the set of nodes and \( \mathcal{E} \) denotes the set of directed edges with

\[
\mathcal{E} = \{ (n', n) \in \mathcal{N} \times \mathcal{N} \mid n' \in \mathcal{V}_n \}.
\]

The subscript \( T \) indicates the number of channels used, \( T = |\Gamma| \). The weight of an edge \((n', n) \in \mathcal{E} \) is simply given by \( \omega(n', n) \), using the aforementioned transmission capacity function (Equation 5).

It follows directly from the definition of a schedule graph \( G_{T}(\mathcal{N}, \mathcal{E}) \) that for any path \( n_0, n_1, \ldots, n_k \) with \( n_i \in \mathcal{N}, \forall i \leq k \) and \((n_i, n_{i+1}) \in \mathcal{E}, \forall i < k \) there is also a corresponding schedule of channel assignments \( \psi(n_0, n_1), \psi(n_1, n_2), \ldots, \psi(n_k, n_0) \) in a way that node \( n_0 \) is able to consecutively transmit data to node \( n_k \) at a rate strictly greater than zero. We will make use of this property later on to reason about the achievable capacity of the underlying physical network.

## 3. Throughput Capacity

Throughout this section, an ad hoc network is represented by its schedule graph \( G_{T}(\mathcal{N}, \mathcal{E}) \) and the corresponding weight function \( \omega \). Capacity is then defined over a set \( \Psi \) of communication pairs:

\[
\Psi \subseteq \{ (n', n) \in \mathcal{N} \times \mathcal{N} \mid n' \neq n \}.
\]
More precisely, we say that a schedule graph \( G_T(N, E) \) with a communication pattern \( \Gamma \) has a throughput capacity of \( \lambda_{n, n} \) if a communication pair \((n', n) \in \Gamma \) can expect an end-to-end throughput of \( \lambda_{n, n} \) bits per second.

Important to note the computation of throughput capacity is the routing function \( \eta : N \times N \longrightarrow \mathcal{P}(E) \). Hence, for a given source-destination pair \((n', n) \) the resulting route simply consists of the set\(^2\) of edges included in the sequence \( e_0, e_1, \ldots, e_{k-1} \), with \( e_i = (n_i, n_{i+1}) \in E \) and \( n_0 = n' \) and \( n_k = n \).

We now want to analyze the expected throughput \( \lambda \) of a communication pair \((n', n) \in \Gamma \). Since both the network and its graph representation \( G_T(N, E) \) are random, the resulting throughput per node pair can also be considered as random. Based on this, the approach we follow is of a probabilistic nature. For any node pair \((n, n') \in \Gamma \), we model throughput capacity as a random variable \( \zeta_{n, n'} : \mathcal{P}(\Gamma) \longrightarrow [0, \infty] \) and then compute the expected value \( E[\zeta_{n, n'}] \) of \( \zeta_{n, n'} \) with \( E[\zeta_{n, n'}] = \lambda_{n, n} \). Consider the fact that in a schedule graph, a path between two nodes also reflects a schedule of channels. Throughput capacity is a metric reflecting a schedule of channels. Throughput capacity is achieved when considering all the traffic \( \Upsilon \) taking place in the network and its graph representation \( G_T(N, E) \) of order the edges in a route is not important for the computation of \( \lambda \) we prefer the set notion which simplifies further treatment.

### 3.1 Computing \( \lambda_{n, n} \) using Monte-Carlo methods

One can compute \( E[\zeta_{n, n'}] \) given the common density function \( p(\zeta_{n, n'}) \) for the random variables \( \zeta_{n, n'} \). However, finding the density function \( p(\zeta_{n, n'}) \) is not trivial. In fact, the problem can be viewed as an extension to the traditional connectivity problem where one tries to find the probability of whether a given node distribution and transmission range results in a connected network. In this paper we do not pursue an analytical treatment of \( E[\zeta_{n, n'}] \) but rather use a Monte-Carlo estimator. For this purpose we first generalize our model \( \zeta_{n, n'} \) to reflect also the average throughput capacity \( \zeta = \frac{1}{|\Gamma|} \sum_{(n', n) \in \Gamma} \zeta_{n, n'} \) that can be expected in the network. In fact, due to the linearity of the expected value one can easily verify that \( E[\zeta_{n, n'}] = E[\zeta] \), namely

\[
E[\zeta] = E\left[ \frac{1}{|\Gamma|} \sum_{(n', n) \in \Gamma} \zeta_{n, n'} \right] = \frac{1}{|\Gamma|} \sum_{(n', n) \in \Gamma} E[\zeta_{n, n'}] = E[\zeta_{n, n'}].
\]

Hence, the expected throughput capacity \( \lambda_{n, n} \) can be approximated using the monte carlo method:

\[
\lambda_{n, n} = E[\zeta_{n, n}] = E[\zeta] = \frac{1}{|\Gamma|} \sum_{(n', n) \in \Gamma} E[\zeta_{n, n'}] \approx \frac{1}{|\Gamma|} \sum_{(n', n) \in \Gamma} \frac{1}{k} \sum_{i=0}^{k-1} E[\zeta_{n', n'}|X = X^*_i] \approx \frac{1}{|\Gamma|} \sum_{(n', n) \in \Gamma} \frac{1}{k} \sum_{i=0}^{k-1} \zeta_{n', n'}|X = X^*_i}
\]

Or in other words, we approximately compute the expected value of \( \zeta \) for a given set of parameters by sampling over \( k \) realizations of the underlying random network, with \( X^*_i \) as a concrete set of node placements in the area \( A \).

### 4. CAPACITY OF STATIC NETWORKS

To validate the model, we compute the throughput capacity of two simple, static scenarios. Static in the sense that the network topology as well as the communication pattern is fixed. The throughput of such fixed network configurations can be seen as the conditional expected value \( E[\zeta|X] \) availability between \( n' \) and \( n \) in \( G_T(N, E) \) and therefore \( \zeta_{n, n} \) can be seen as a direct function of the connection probability between the two nodes. Or one can say that the capacity of an ad hoc network is related to the connectivity of its corresponding schedule graph \( G_T(N, E) \). This might be of interest when analyzing capacity in sparse and partially disconnected random networks, but also in mobile scenarios where the movement of the nodes often leads to temporarily broken paths.

In the next sections we show how \( \lambda_{n, n} = E[\zeta_{n, n}] \) can be computed using monte carlo methods.
of the random variable $\zeta$ under a concrete node placement $X$. For a fixed channel assignment $\psi$, $E[\zeta|X]$ simply computes as $E[\zeta|X] = \frac{1}{|T|} \sum_{(n',n) \in T} \zeta_{n',n} \big|_{X=X^*_n}$, where $X^*_n$ is the given set of coordinates of the nodes. For both examples we will consecutively derive $E[\zeta|X]$ by going through the basic steps of section 2 and 3.

The first network topology we consider consists of three nodes being distance $d$ apart from each other, as shown in Figure 1. To simplify the analysis, we use a more primitive interference model $\kappa_{\text{protocol}}$ than the one proposed in Equation 4:

$$\kappa_{\text{protocol}}(n', n, I) = \begin{cases} 
1 & \forall n' \in I : P_{n \rightarrow n'} < P_{n \rightarrow n}
\end{cases} \quad (15)$$

Let us further assume $n' \in D_n$ for all $n' \neq n$. According to $\kappa_{\text{protocol}}$, the set of senders $U_n$ is modelled in a way that a node $n'$ belongs to $U_n$ if, and only if, no other concurrent transmission with a signal stronger than $P_{n \rightarrow n'}$ is received by node $n$. Hence, the graph $G_T$ only depends on how the different channels are assigned to the nodes. We now want to illustrate the outcome of $E[\zeta|X]$ for three possible channel assignments. We keep track of all states and sets of the network model for each of the three channel assignments in Table 2.

In all the configurations we assume shortest path routing and only assign channels to edges that are also used when considering the traffic pattern $Y$. In the case of one common channel $\psi(n) = \psi(n')$ for all nodes $n, n'$, no transmission can correctly be decoded at any receiver (Equation 15) which leads to $U_n = \emptyset, E = \emptyset, S_{n,n'} = 0, \zeta_{n,n'} = 0$ and finally to $E[\zeta|X] = 0$. In the presence of two separate channels ($T = 2$), two directed links can be established (among the potential 6). Along with a communication pattern $Y = \{(A,B),(B,C),(C,A)\}$ (see Table 2), $E[\zeta|X]$ is $W \cdot 1/3 \cdot (0 + 1/2 + 0) = 1/6 \cdot W$. In case transmissions are spread over three channels, $G_T(N,E)$ becomes fully connected and $E[\zeta|X]$ equals $W \cdot 1/3 \cdot (1/3 + 1/3 + 1/3) = 1/3 \cdot W$. Adding a fourth edge, e.g., to the transmission between node $A$ and $B$, does not increase the capacity any further. This is because the increase in the bottleneck $(B_{A,B} = 2)$ is compensated by the increase in the total amount of used channels ($T = 4$).

The situation is slightly different for the scenario in Figure 2 since node $B$ acts as a router and some of its bandwidth is consumed by traffic sent from $A$ to $C$. The case $T = 1$ is trivial and comparable with the corresponding case in the triangle scenario. Assigning two channels to the four edges results in two established links ($U_A = \{B\}$ and $U_C = \{B\}$). Considering the traffic pattern $Y$, this is sufficient for one path to be established $S_{B,C}$ and the resulting capacity $E[\zeta|X]$ computes to $1/6 \cdot W$. Two of the three routes can be established if 3 channels are used ($T = 3$), which results in $E[\zeta|X] = 2/9 \cdot W$. The capacity of the given traffic pattern can be further improved by assigning one channel per transmission pair. All the routes can be established with a bottleneck of $B_{n,n'} = 1, \forall(n, n') \in T$ and $E[\zeta|X]$ equals $1/4 \cdot W$. Obviously, the same channel assignment results in a different capacity if another traffic pattern is used, like e.g. $Y = \{(A,C),(B,C),(C,A)\}$. Since the link between node $B$ and $C$ is used twice, the values for $B_{A,C}$ and $B_{B,C}$ reduce to $1/2$. For such a traffic pattern, a 5-channel

| Used Channels $T$ | $\psi(n)$ | $U_n$ | $Y$ | $B_{n,n'}$ | $E[\zeta|X]$ |
|------------------|-----------------|-----------------|-----------|-----------------|-----------------|
| 1                | $\psi(B,A) = \psi(C,B) = \psi(A,C) = \emptyset$ | $\emptyset$ | (A,B) | $0$ | $0$ |
| 2                | $\psi(B,A) = \psi(C,B) = \psi(A,C) = \emptyset$ | $\emptyset$ | (B,C) | $B_{A,B} = 0$ | $1/6W$ |
| 3                | $\psi(B,A) = \psi(C,B) = \psi(A,C) = \emptyset$ | $\emptyset$ | (C,A) | $B_{C,A} = 0$ | $1/3W$ |
| 4                | $\psi(B,A) = \psi(C,B) = \psi(A,C) = \emptyset$ | $\emptyset$ | (A,B) | $B_{A,B} = 2$ | $1/3W$ |

Table 2: States for the triangle scenario

Figure 1: The triangle scenario

Figure 2: The chain scenario
thus cation patterns, the shortest path may create hotspots, and shorter the routes the higher the throughput capacity. For one that implements simple best effort channel assignment \( \eta \) of a network. During this work, we use the shortest-path number of channels is favorable for the overall throughput.

eral, it seems that a combination of short paths and a small channel assignment not only depends on the node location but also on the communication pattern \( \Upsilon \). Depending on \( \Upsilon \) and \( \eta \), the number of channels maximizing throughput may be above or below what would be necessary for full network connectivity. Therefore, an optimized channel assignment certainly has to take the routing into account [9]. In general, it seems that a combination of short paths and a small number of channels is favorable for the overall throughput of a network. During this work, we use the shortest-path algorithm from Floyd and Warshall [7] as routing function \( \eta \). For the channel assignment \( \psi \), we use two algorithms:\footnote{Note that the capacity model proposed in this paper operates on an abstract channel assignment and serves as a framework to actually evaluate the effect of various channel assignments on throughput capacity.}

we refer to that one as RandomEdge – and one that implements a fairly optimal channel assignment, called Greedy\(^+\). The RandomEdge channel assignment (Algorithm 1) assigns a set of maximum \( T \) channels in a round robin manner modulo \( T \) to all transmission pairs \((n, n')\) with \( n' \in D_n \). At each round one transmission pair is picked on a random basis. The Greedy\(^+\) channel assignment aims at assigning channels in a conflict free way. Traditionally, greedy algorithms operating on graphs are meant to traverse the nodes according to some iterator function while always assigning the lowest channel to a node that is not yet used within the node’s neighborhood. However, as we want to assign channels to edges rather than to nodes, we cannot directly apply the greedy algorithm. Additionally, each edge may have multiple channels assigned. We therefore propose to assign channels to edges in such a way that each edge receives the first channel which has not already been assigned to one of its neighbors, where a neighbor of an edge \((n', n)\) is defined to be any edge including either \( n' \) or \( n \). This accounts for the fact that a node may neither be able to transmit data to several nodes within the same channel nor to simultaneously decode signals from nodes transmitting in the same channel. Unfortunately, such an edge-based greedy channel assignment may not lead to the desired result, as shown in Figure 3. Both node \( A \) and \( C \) transmit in channel 0, causing a conflict at node \( B \). In [12, 15], the authors propose to overcome this problem by extending the neighborhood of an edge to include all interfering edges within a two-hop distance. While the algorithm works well assuming that only direct neighbors (\( D_n \)) interfere with each other, it fails under more complex interference models. As an example, using the SINR interference model, two nodes in distance 2d may interfere with a transmission at distance \( d \) if their interference is accumulated. The solution we adopt is to build a so-called conflict mesh that replaces the notion of a neighborhood in the traditional greedy algorithm. A conflict mesh consists of two types of sets: \( C_e \) includes all the conflicting nodes for a given node pair \( e := (n, n) \) with \( n' \in D_n \), and \( C_c \) includes all the conflicting node pairs for a given node

<table>
<thead>
<tr>
<th>Used Channels ( T )</th>
<th>( \psi(n) )</th>
<th>( \Upsilon )</th>
<th>( B_{n,n'} )</th>
<th>( E[\xi])</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>( \psi(A, B) = \psi(C, B) = {0} )</td>
<td>( \psi(B, C) = {1} )</td>
<td>( \psi(B, A) = \emptyset )</td>
<td>( \psi(C, B) = {2} )</td>
</tr>
<tr>
<td>3</td>
<td>( \psi(A, B) = {0} )</td>
<td>( \psi(B, C) = {1} )</td>
<td>( \psi(B, A) = {2} )</td>
<td>( \psi(C, B) = {3} )</td>
</tr>
<tr>
<td>4</td>
<td>( \psi(A, B) = {0} )</td>
<td>( \psi(B, A) = {1} )</td>
<td>( \psi(B, C) = {2, 3} )</td>
<td>( \psi(C, B) = {4} )</td>
</tr>
<tr>
<td>5</td>
<td>( \psi(A, B) = {0} )</td>
<td>( \psi(B, A) = {1} )</td>
<td>( \psi(B, C) = {2, 3} )</td>
<td>( \psi(C, B) = {4} )</td>
</tr>
</tbody>
</table>

Table 3: States for the chain scenario
6. CAPACITY OF LARGER NETWORKS

In this section we analyze throughput capacity of various types of communication patterns and network topologies.

Algorithm 1 RandomEdge Channel Assignment
Input: The maximum number of channels $T$.
Output: Channel assignment $\psi$ and number of used channels.

1: $\mathcal{O} := \mathcal{E}$;
2: $i := 0$;
3: while $\mathcal{O} \neq \emptyset$ do
4: $e := \text{ANY} \{ e \in \mathcal{E} \}$;
5: $\mathcal{O} := \mathcal{O} \setminus \{ e \}$;
6: $\psi(e) := i$;
7: $i := i + 1$ MOD $T$;
8: end while
9: if $|\mathcal{N}| < |\mathcal{T}|$ then
10: return $|\mathcal{N}|$;
11: else
12: return $|\mathcal{T}|$;
13: end if

Algorithm 2 Algorithm to create a conflict mesh
Input: Nodes $\mathcal{N}$, decodables $\mathcal{D}_n$, load function $\mu$, interference model $\kappa$.
Output: Conflicts $C^D_e, C^N_n, \forall e := (n',n) \text{ with } n' \in \mathcal{D}_n$.

1: for all $n \in \mathcal{N}$ do
2: for all $n' \in \mathcal{D}_n$ do
3: if $\mu((n',n)) > 0$ then
4: $e := (n',n)$;
5: $C^D_e := C^D_e \cup \{ n \}$;
6: $C^N_n := C^N_n \cup \{ e \}$;
7: $\mathcal{T} := \emptyset$;
8: $L := \text{sort}(\mathcal{N}\setminus\{n, n'\})$ such that $n' \prec n$;
9: for $n' \in L$ do
10: if $n' \neq n$ then
11: $\mathcal{T} := \mathcal{T} \cup \{n'\}$;
12: if $\kappa_{\text{inter}}(n,n',\mathcal{T}) \neq 1$ then
13: $C^D_e := C^D_e \cup \{ n' \}$;
14: $C^N_n := C^N_n \cup \{ e \}$;
15: end if
16: end if
17: end for
18: end for
19: end for
20: end for

To simplify the notation we will refer to $E[\gamma]$ as $\lambda$ for the rest of the paper. For each analyzed configuration we also provide results taken from simulations with ns-2 [19] under the very same topology and communication setup. The idea is to compare real 802.11 multipath throughput (ns-2 simulation) with the information theoretical throughput (calculated with our model) under a given schedule and channel assignment. Throughout this section, we use a path loss radio propagation as defined by $\psi_{pl}$ and a SINR based interference model, $\kappa_{\text{inter}}$, as described in Equation 4. Since we use $\psi_{pl}$, the threshold for a node $n$ to be part of $\mathcal{D}_n$ only depends on the distance between the two nodes. We have fixed this threshold to be $250m$. To avoid mixing up capacity measurements with routing issues, packets within ns-2 simulations are forwarded using pre-computed shortest path routes. We further have set the MAC data rate in ns-2 to 1Mbit since operating 802.11 at higher rates results in drastically reduced efficiency and makes the measurements differ.

\footnote{Only nodes participating in communication are considered.}

\footnote{Note that Greedy assigns channels in a way that finally $B_{n',n} = 1, \forall (n',n) \in T$.}
6.1 Chain

In a first comparison we look at a configuration of a chain of \( n \) nodes. Each node is 200 meters away from its neighbor. The first node acts as a source of data traffic, the last node is the traffic sink. Data is sent as fast as the MAC allows.

We use Greedy\(^+\) as the channel assignment algorithm. Since there are no random components involved, \( \lambda \) is a direct function of the channels needed, and computes to 1/4 as the chain grows. From Figure 4a, we see that the value of \( \lambda \) lies above the throughput measured with ns-2, especially when the chain becomes large. This is due to the overhead of headers, RTS, CTS and ACK packets but also because in reality nodes fail to achieve an optimal schedule. The results obtained with our model match those presented in [14], where the authors discuss throughput capacity measurements taken from ns-2 simulations with respect to theoretical upper bounds.

As a more realistic scenario, we now investigate random communication patterns in chain topologies. For this purpose, we assign a random destination \( d(n) \in N \setminus \{n\} \) to every node \( n \in N \). Figure 4b shows the effect of such a traffic pattern on throughput. The plot shows a quite close match between \( \lambda \) and the measurements obtained with ns-2. This is not too surprising since we know from Figure 4a that the throughput of an 802.11 chain matches the theoretical limit if the chain length is short. Under a random communication pattern the average path length in a chain is far below the maximum value of \( n - 1 \), for a chain of length \( n \) it is \( \frac{2n-2}{3} \) [14]. Furthermore overlapping communication paths reduce capacity (\( B_{n,n} \) in our model) due to the forwarding load inflicted upon the nodes, especially if the chain becomes large.

6.2 Grid

We look at grid topologies where each node is 200 meters away from its closest neighbor and the nodes communicate using a random communication pattern. Since we keep the transmission radius and therefore also the degree \( d \) of all nodes constant\(^7\), it seems natural that there is an optimum for the number of channels, almost independent of the size of the grid. Figure 5a shows the throughput capacity in a grid of 100 nodes using RandomEdge with a varying number of channels. Clearly, the throughput capacity is maximized when using around 50 channels in total. In Figure 5b, the throughput capacity of a grid with a RandomEdge\(_{50}\) channel assignment is shown together with the results produced by a Greedy\(^+\) assignment and measurements taken from ns-2. Unlike in the chain scenario, there is quite a gap between the ns-2 measurements and our model using Greedy\(^+\). One possible explanation is that the two dimension of the grid and the even node distribution make it yet more difficult for 802.11 to achieve an optimal schedule.

The Greedy\(^+\) channel assignment on the other hand, finds a schedule with around 50 channels for every grid configuration, which is also far less than the optimum of 50 channels used in RandomEdge.

6.3 Random Topology

We consider random topologies of \( n \) nodes distributed uniformly within an area of 1000 \( \times \) 1000 meters. As in the previous topologies, all nodes have \( \beta_2 \) configured such that their transmission range equals 200m. Each node \( n \) acts as a traffic generator and has a random destination assigned, chosen uniformly out of \( N \setminus \{n\} \). Figure 6 shows the throughput capacity \( \lambda \) in contrast with ns-2 simulation measurements. The result approves the trend already observed in the previous configurations of the chain and the grid: randomness improves 802.11 throughput capacity with respect \( \lambda \). This might be particularly the case in random networks where the node density is increasing with an increasing number of nodes. In such network configurations the demand for channels is high due to the high node degree, leaving less room for an optimal channel assignment. Or in other words, the relative costs of using, e.g., 100 channels by 802.11 instead of the 80 used by Greedy\(^+\), is lower than using 50 instead of 30 as it is the case, e.g., in the grid.

7. APPLYING THE MODEL

We now use the model to investigate various tradeoffs in multihop networks with regard to throughput capacity. First, we tackle the problem of finding the optimal transmission range in fixed-traffic networks. Second, we analyze the effect of randomized signal propagation on throughput capacity under different interference models.

7.1 Optimal transmission range

Finding an optimal transmission range is commonly known as the connectivity problem, where we are interested in the minimum transmission range that leads to a connected network. Capacity can be studied in a similar way. Assume a fixed traffic density \( \xi \), i.e., that every node transmits data at a rate \( \xi = W/K \) for some value \( K \), where \( W \) is the maximum transmission rate. Such a traffic density can be modelled by dividing the common channel into subchannels \( \Gamma^0, \Gamma^1, \ldots, \Gamma^{K-1} \). Recall that nodes transmit data only within their assigned channels. Here, transmission is meant to be

\[ \text{Algorithm 3 Greedy}^+ \text{ channel assignment} \]

\begin{align*}
\text{Input:} & \ N, \ \text{decodables} \ D_n, \ \text{Conflicts} \ C^D_n, \ \text{load function} \ \mu \\
\text{Output:} & \ \text{Channel assignment} \ \psi \text{ and number of channels used} \\
1: & \ \Pi := \emptyset \\
2: & \ \text{for all} \ n \in N \ \text{do} \\
3: & \ \text{for all} \ e \in D_n \ \text{do} \\
4: & \ e := (n, n) \\
5: & \ \text{for} \ i := 0; \ i < \mu(n, n) \ \text{do} \\
6: & \ \Omega := \emptyset \\
7: & \ \text{for all} \ n'' \in C^D_n \ \text{do} \\
8: & \ \Omega := \Omega \cup \psi(n'') \\
9: & \ \text{end for} \\
10: & \ \text{for all} \ e' \in C^D_n \ \text{do} \\
11: & \ \Omega := \Omega \cup \psi(e') \\
12: & \ \text{end for} \\
13: & \ \gamma_i := \text{ANY} \ \gamma \notin \Omega \\
14: & \ \text{Define} \ \psi(e) := \psi(e) \cup \{\gamma_i\} \\
15: & \ \Pi := \Pi \cup \{\gamma_i\} \\
16: & \ \text{end for} \\
17: & \ \text{end for} \\
18: & \ \text{return} \ |\Pi| \text{;}
\end{align*}

\footnote{The size of the set} \ D_n \\
\footnote{If we neglect the border nodes}
a node’s own transmission as well as the forwarding load. The question now is at what transmission power (range) the nodes should transmit on average in order to maximize the throughput capacity. Figure 7 shows the throughput capacity as a function of transmission range for three different traffic densities. The network consists of 100 nodes uniformly distributed in an area of 1000 \times 1000 square meters. From Figure 7 we see that the optimal transmission range changes with the traffic density. This is interesting since it stresses that, e.g., topology control should take into account the traffic density as well when looking for optimal transmission power selection. While it has been shown analytically that the maximum throughput capacity of a wireless multihop network is bound by the lowest transmission range R that makes the network connected [9], there is up to our knowledge no work on optimal transmission ranges for networks with a fixed traffic load. The model we propose in this paper supports such analysis, opening up the possibility of exploring in greater detail the relation between transmission range and traffic load under realistic network conditions.

7.2 Effect of randomized radio propagation

It is well known that representing the transmission range as a direct function of the distance does not reflect the reality of radio transmitters. In fact, the received transmission power can be seen as a random variable due to fading effects. The impact of randomized transmission power – also known as the effect of shadowing – has been analyzed with regard to network connectivity [5]. The study extends previous work on connectivity [10] by showing that fading effects increase connectivity if the randomization itself is assumed to be symmetric for both ends of a potential link. On the other hand, if the fading effect is modeled as an independent random variable and communication is acknowledgment-based\(^9\), connectivity is observed to be degraded with increasing randomness [17]. Analyzing these effects with regard to throughput capacity is much more complex since it also influences the interference perceived while receiving data. As a benefit of our model, effects of signal

\(\text{\textsuperscript{9}}\) like 802.11
propagation properties can be analyzed by just using an appropriate signal propagation function while computing the graph topology, as explained in section 2. We have studied randomized radio propagation in two different interference models: the signal to interference plus noise model $\kappa_{\text{sinr}}$ as described in Equation 4 and the protocol model $\kappa_{\text{protocol}}$ (Equation 15). Obviously, the two models $\kappa_{\text{sinr}}$ and $\kappa_{\text{protocol}}$ are quite different in terms of interference-sensitivity. While in $\kappa_{\text{sinr}}$ interference is accumulated among many nodes, the $\kappa_{\text{protocol}}$ interference model fails to receive correctly as soon as one interferer exceeds the threshold. Figure 9 shows the effect of shadowing on throughput capacity under the two different interference models for nodes uniformly distributed within an area of $1000 \times 1000$ metres. The corresponding signal propagation model $\theta_{ab}$ is described in Equation 1. For a given propagation distance $d$, the higher the standard deviation $\sigma$, the more the signal is spread around its mean.

What we can see from Figure 9 is that, for both interference models $\kappa_{\text{sinr}}$ and $\kappa_{\text{protocol}}$, the throughput capacity increases with increasing randomization of the signals. This is interesting since for the very same scenario connectivity actually decreases with increasing randomization of the signals, as shown in Figure 8a, matching the study in [17]. The reason for this behavior becomes clearer when we look at the connectivity and capacity curves in more detail. Capacity typically increases with decreasing node density. At least up to some density where the network becomes disconnected one would argue. However, comparing Figures 8b and 9 shows that throughput capacity increases even if the underlying network is partially disconnected. This is because the reduction in used channels ($T$) and the small forwarding traffic ($1/B_{\text{short}}, n$) that is inflicted upon nodes compensates the low connection probability of the nodes by far. That is exactly the behavior we observe in the case of randomized radio propagation. The randomized signal reduces the number of neighbors $\mathcal{N}$ leaving a sparse network where far less channels have to be assigned to the edges. This is shown in Figure 8a, where the number of channels for $\sigma = 0$ grows twice as fast as the for the case of $\sigma = 4$. In general, $\kappa_{\text{protocol}}$
Figure 9: Randomized radio propagation under different interference models

(a) $\kappa_{\text{sinr}}$

(b) $\kappa_{\text{protocol}}$

Figure 10: Random networks and their schedule graphs under different radio propagation models

(a) Random network under $\vartheta_{\text{pl}}$ with $\rho = 2$ and $\sigma = 0$

(b) Schedule graph under $\vartheta_{\text{pl}}$ with $\rho = 2$ and $\sigma = 0$

(c) Random network under $\vartheta_{\text{sh}}$ with $\rho = 2$ and $\sigma = 4$

(d) Schedule graph under $\vartheta_{\text{sh}}$ with $\rho = 2$ and $\sigma = 4$

The influence of shadowing is also illustrated in Figure 10b by means of the schedule graph for $N = 100$. Clearly, the schedule graph for the shadowing case is much less dense than for the pure path loss case. While effects of shadowing
on throughput capacity have been analyzed for unidirectional links [20], there has up to our knowledge not been any work done so far under acknowledgement-based communication, thereby illustrating the potential of the model we propose.

8. CONCLUSIONS

We have presented a probabilistic approach to capacity analysis by linking the throughput of a communication pair in an ad hoc network to the connection probability of the nodes in a so called schedule graph. Contrary to existing work on capacity, based on simplified network models (isotropic radio propagation, unidirectional links, perfect scheduling, straight line routing, etc.) and asymptotic bounds, our approach analyzes capacity under more realistic network configurations. In our model, the effective throughput of a random network is considered as a random variable depending on the node distribution, the communication pattern, the radio propagation, channel assignment, etc. Expected values of that random variable are then computed using Monte-Carlo methods. The modularity of the proposed model makes it a powerful tool to analyze any form of physical and logical interaction with regard to throughput capacity. While the idea of treating throughput capacity as the expected value of a well modelled random variable serves as the basis for this work, the general concept can also be applied to other network properties. In that sense, the paper also suggests a new approach to ad hoc network analysis in cases where pure analytical approaches fall short and protocol specific network simulations are not generic enough. This is of particular evidence against the background of the ever increasing computing power of today’s hardware. For instance, although the computational costs of our model is $O(n^3)$, we were able to compute all the results in our paper within a few minutes using a cluster of 32 machines and JOpera [1] as a grid engine.

9. REFERENCES