Computing Throughput Capacity for Realistic Wireless Multihop Networks

Patrick Stuedi Gustavo Alonso
Department of Computer Science
Swiss Federal Institute of Technology (ETH Zurich)
8092, Zurich, Switzerland
{studiep, 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* $G_T(\mathcal{N}, \mathcal{E})$. 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 $G_T(\mathcal{N}, \mathcal{E})$ 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 throughput capacity is computed for various forms of network configurations and the results are compared to simulation results obtained with ns-2.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Wireless Networks - Multihop, Capacity

General Terms

Algorithms, Performance

Keywords

Wireless Multihop Networks, Capacity

MSWiM'06, October 2–6, 2006, Torremolinos, Malaga, Spain.

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* $G_T(\mathcal{N}, \mathcal{E})$ 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 $G_T(\mathcal{N}, \mathcal{E})$. 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.

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 [7], 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 [6], bound attenuation functions [5] and multiple channels [9]. 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 [3, 15] 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 [16] but without considering multihop networks.

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.
- By linking throughput capacity of multihop wireless networks with the connection probability in a *schedule graph* $G_T(\mathcal{N},\mathcal{E})$, 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* $G_T(\mathcal{N},\mathcal{E})$. Examples of physical properties are node locations or perceived signal strengths.

In a schedule graph, $\mathcal{N}$ is the set of nodes in the network and $\mathcal{E}$ denotes a set of directed edges between the nodes, such that the existence of a sequence of nodes $n_0, n_1, ...n_k$ – with $n_i \in \mathcal{N}, \forall i \leq k$ and $(n_i, n_{i+1}) \in \mathcal{E}, \forall i < k$ – states that there is also a schedule of channel assignments $\psi(n_0, n_1)$, $\psi(n_1, n_2)$, $...\psi(n_{k-1}, n_k)$ in a way 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 deduce the achievable capacity of the underlying wireless network. In this section, we first define some common properties in order to gradually develop the graph representation by assigning three sets $\mathcal{D}_n \supseteq \mathcal{U}_n \supseteq \mathcal{V}_n$ of nodes to each node n, with $\mathcal{D}_n \subseteq \mathcal{N}$. 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 notations 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 $\mathcal{N}$, a node distribution δ, a signal propagation ϑ, a channel assignment ψ and an interference model κ. We assume $x_n \in \mathcal{R}^2$ to be the coordinate[1] of node n, identifying the node's position with respect to an area $\mathcal{A}$, and we consider the set $\mathcal{N}$ of nodes as being distributed in $\mathcal{A}$ according to some probability function $\delta : \mathcal{A} \longrightarrow [0, 1]$. Throughout this paper, we use $\mathcal{P}(\cdot)$ to refer to the collection of all possible subsets of a set.

[1]The model could also be applied to $\mathcal{R}^3$

2.1 Decodables $\mathcal{D}_n$

Let us start by defining how signals are propagated. First, each node n is supposed to transmit with a signal power $P^{\text{t}}_n \in [0, \infty[$. For a certain signal propagation ϑ, $P_{n\leftarrow n'} = \vartheta(P^{\text{t}}_{n'}, |x_{n'} - x_n|) \in [0, P^t_{n'}]$ denotes the power of the received signal at node n perceived due to a transmission of node n'. In the simplest case, ϑ 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 a path loss exponent ρ and d_0 as a reference distance for the antenna far-field. A more sophisticated model is the log normal shadowing radio propagation [11]:

$$\vartheta_{sh}(p, d) = p \cdot (d/d_0)^{-\rho} \cdot 10^{X/10} \quad (1)$$

where X is a Gaussian random variable with zero mean and standard deviation σ, and ρ is the aforementioned path loss exponent. In case of σ equal 0, there is no random effect and $\vartheta_{sh} \equiv \vartheta_{pl}$. We now define a set D_n as

$$\mathcal{D}_n = \{n' \in \mathcal{N} \mid P_{n\leftarrow n'} \geq \beta_D\} \quad (2)$$

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

2.2 Senders $\mathcal{U}_n$

Transmissions from a node n' to another node n are bound to a set of predefined channels $\psi(n', n)$, where $\psi : \mathcal{N} \longrightarrow \mathcal{P}(\Gamma)$ and Γ is the set of all available channels. We further use $\psi^*(n) = \bigcup_{n' : n\in\mathcal{D}'_n} \psi(n, n')$ 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 is 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 $\kappa : \mathcal{N} \times \mathcal{N} \times \mathcal{P}(\mathcal{N}) \longrightarrow \{0, 1\}$ with

$$\kappa(n', n, I) = \begin{cases} 1 & \text{The signal of } n' \text{ can be decoded at node } n \text{ under a set } I \text{ of interferering nodes} \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

As an example of an interference model [11], the well known *signal to interference plus noise* model κ_{sinr} computes as

$$\kappa_{sinr}(n', n, I) = 1 \iff \frac{P_{n\leftarrow n'}}{P^*_n + \sum_{n''\in I} P_{n\leftarrow n''}} > \beta_{sinr} \quad (4)$$

for a threshold β_{sinr} and P^*_n as the aforementioned thermal noise perceived at node n. Of course, for any interference

	Notation	Semantic
Parameters for $G_T(\mathcal{N},\mathcal{E})$	$\mathcal{N}$	set of nodes in the network
	δ	node distribution function
	ϑ	signal Propagation function
	ψ	channel assignment function
	κ	interference model
$G_T(\mathcal{N},\mathcal{E})$ internal	X	set of coordinates x_n for each node n
	Γ	set of channels
	P_n^{t}	transmission power of node n
	$P_{n \leftarrow n'}$	signal power perceived at node n due to the transmission of node n'
	P_n^*	thermal noise perceived at node n
	$\mathcal{D}_n$	set of nodes that can be decoded at node n without noise
	$\mathcal{U}_n$	set of nodes that can be decoded at node n under noise
	$\mathcal{V}_n$	set of nodes that can be considered as neighbors of n
	$\mathcal{E}$	set of directed edges in a *schedule graph*
Parameters for λ	$G_T(\mathcal{N},\mathcal{E})$	schedule graph
	$\omega(n',n)$	weight function indicating the number of channels used on a link
	Υ	set of source destination pairs
	η	routing function
λ internal	Π	set of path participating in communication
	$B_{n',n}$	lowest number of channels between any two neighbors along a path
	T	used channels, $T = \|\Gamma\|$ in $G_T(\mathcal{N},\mathcal{E})$
	$\zeta_{n',n}$	achievable throughput capacity along a path
	λ	expected throughput capacity

Table 1: Notation

model it must be given that $\kappa(n', n, \emptyset) = 1 \iff n' \in \mathcal{D}_n$. Based on the notion of κ we define the transmission capacity $\omega : \mathcal{N} \times \mathcal{N} \longrightarrow [0, |\Gamma|]$ between two nodes n and n' as

$$\omega(n', n) = \sum_{\gamma \in \psi(n', n)} \kappa(n', n, I_\gamma). \tag{5}$$

Here, I_γ denotes the set of nodes transmitting during channel γ, or $\mathcal{I}_\gamma = \{n' \in \mathcal{N} \mid \gamma \in \psi^*(n')\}$. 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

$$\mathcal{U}_n = \{n' \in \mathcal{N} \mid \omega(n', n) > 0\}. \tag{6}$$

.

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:

$$\mathcal{V}_n = \{n' \in \mathcal{N} \mid n' \in \mathcal{U}_n \wedge n \in \mathcal{D}_{n'}\}. \tag{7}$$

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 acknowledgment 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

$$\mathcal{E} = \{(n', n) \in \mathcal{N} \times \mathcal{N} \mid n' \in \mathcal{V}_n\}. \tag{8}$$

The subscript T indicates the number of channels used. 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 $n_0, n_1, ...n_k$ – 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 $\psi(n_0, n_1)$, $\psi(n_1, n_2)$, $...\psi(n_{k-1}, n_k)$ 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 when deducing 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 ω. Capacity is then defined over a set Υ of communication pairs:

$$\Upsilon \subseteq \{(n', n) \in \mathcal{N} \times \mathcal{N} | n' \neq n\}. \tag{9}$$

More precisely, we say that a *schedule graph* $G_T(\mathcal{N},\mathcal{E})$ with a communication pattern Υ has a throughput capacity of $\lambda_{n',n}$ if a communication pair $(n', n) \in \Upsilon$ can expect an end-to-end throughput of $\lambda_{n',n}$ bits per second.

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

[2]Since we assume no loops and the order of the edges in a route is not important for the computation of λ, we prefer the set notion which simplifies further treatment.

We now want to analyze the expected throughput λ of a communication pair $(n', n) \in \Upsilon$. Since both the network and its graph representation $G_T(\mathcal{N}, \mathcal{E})$ are random, obviously 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 $(n, n') \in \Upsilon$, we model throughput capacity as a random variable $\zeta_{n',n} : \mathcal{P}(\Upsilon) \longrightarrow [0, \infty[$ to 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 concave metric, meaning that the available throughput for a certain source destination pair is always determined by the node with the lowest bandwidth, the so called bottleneck. Hence, let $B^*_{n',n} = \min_{e \in \eta(n',n)} \omega(e)$ be a random variable indicating lowest number of channels available between two nodes along the path from n' to n. One can easily verify that the resulting throughput capacity along the path can not be bigger than $W \cdot B^*_{n',n}/T$, where W is the maximum transmission rate equal to all nodes and $T =\mid \Gamma \mid$ is the number of channels used in total.

The throughput capacity may be further diminished when considering all the traffic Υ taking place in the network. For this purpose let us define a so called load function $\mu : \mathcal{E} \longrightarrow [0, N]$, indicating to what extent a certain edge $e \in \mathcal{E}$ is shared among other ongoing traffic, or more formally:

$$\mu(e) = \sum_{\substack{i=\eta(n',n) \\ (n',n)\in\Upsilon}} 1_i(e) \tag{10}$$

where $1_i : \mathcal{E} \longrightarrow \{0, 1\}$ is the set membership function. If we want to take all ongoing traffic into account we therefore have to consider μ while computing $B^*_{n',n}$, or

$$B_{n',n} = \begin{cases} 0 & \eta(n',n) = \emptyset \\ \min_{e\in\eta(n',n)} \frac{\omega(e)}{\mu(e)} & \text{otherwise.} \end{cases} \tag{11}$$

Based on the definition of $B_{n',n}$ we now claim that the achievable throughput $\zeta_{n',n}$ for a communication pair $(n', n) \in \Upsilon$ in a *schedule graph* $G_T(\mathcal{N}, \mathcal{E})$ can be modelled as

$$\zeta_{n',n} = \frac{W \cdot B_{n',n}}{T}. \tag{12}$$

In a simplified setup, where each node is only allowed to transmit within one single channel, $B_{n',n}$ refers to the path availability between n' and n in $G_T(\mathcal{N}, \mathcal{E})$ 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* $G_T(\mathcal{N}, \mathcal{E})$. 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.

3.1 Computing $\lambda_{n',n}$ using Monte-Carlo methods

One could 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 $\zeta = \frac{1}{|\Upsilon|}\sum_{(n',n)\in\Upsilon} \zeta_{n',n}$ 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

$$\begin{aligned} E[\zeta] &= E[\frac{1}{|\Upsilon|} \sum_{(n'',n''')\in\Upsilon} \zeta_{n'',n'''}] \\ &= \frac{1}{|\Upsilon|} \sum_{(n'',n''')\in\Upsilon} E[\zeta_{n'',n'''}] = E[\zeta_{n,n'}]. \end{aligned} \tag{13}$$

Hence, the expected throughput capacity $\lambda_{n',n}$ can be approximated using the Monte-Carlo method:

$$\begin{aligned} \lambda_{n,n'} &= E[\zeta_{n,n'}] = E[\zeta] = \frac{1}{|\Upsilon|} \sum_{(n'',n''')\in\Upsilon} E[\zeta_{n'',n'''}] \\ &= \frac{1}{|\Upsilon|} \sum_{(n'',n''')\in\Upsilon} \int_{\mathcal{R}^{2N}} E[\zeta_{n'',n'''}|X = X^*] f(X^*) dX^* \\ &\approx \frac{1}{|\Upsilon|} \sum_{(n'',n''')\in\Upsilon} \frac{1}{k} \sum_{i=0}^{k-1} E[\zeta_{n'',n'''}|X = X^*_i] \\ &= \frac{1}{|\Upsilon|} \sum_{(n'',n''')\in\Upsilon} \frac{1}{k} \sum_{i=0}^{k-1} \zeta_{n'',n'''}|_{X=X^*_i} \end{aligned} \tag{14}$$

Or in other words, we approximately compute the expected value of ζ 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 $\mathcal{A}$.

4. MAXIMIZING $E[\zeta]$

The main objective of the model is to provide a tool to explore the interaction between different physical and logical properties of ad hoc networks with regard to throughput maximization. From the definition of $\zeta_{n',n}$ in Equation 12, one can see the various trade-offs involved. First of all, one would like to maximize $B_{n',n}$, the channel bottleneck on the path. For random networks, with random communication patterns, it is easy to verify that $B_{n',n}$ decreases with increasing path lengths because the same traffic is propagated to an increasing number of nodes ($\mu(e)$ grows). Hence, the shorter the routes the higher the throughput capacity. For some other than random network topologies and communication patterns, the shortest path may create hotspots, and thus $B_{n',n}$ decreases, which results in a low throughput capacity. Besides $B_{n',n}$, throughput capacity is also heavily affected by the number of used channels. Obviously, in order to maximize throughput one wishes to keep the num-

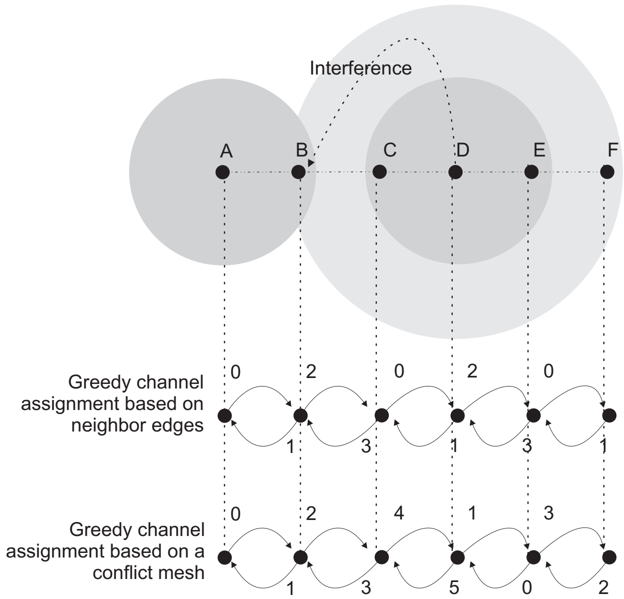


Figure 1: Result of a *Greedy* channel assignment in the chain topology

ber of channels low, not too low of course, otherwise the network becomes disconnected. In reality things are slightly more complex. The optimal channel assignment not only depends on the node location but also on the communication pattern Υ. Depending on Υ and η, 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 [2]. 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 [4] as routing function η. For the channel assignment ψ, we use two algorithms[3]: one that implements simple best effort channel assignment – 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 $n' \in \mathcal{D}_n$. 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

Algorithm 1 RandomEdge Channel Assignment

Input: The maximum number of channels T^*
Output: Channel assignment ψ and number of used channels

$\mathcal{O} := \mathcal{E}$;
$i := 0$;
while $\mathcal{O} \neq \emptyset$ **do**
    $e := ANY\{e \in \mathcal{E}\}$;
    $\mathcal{O} := \mathcal{O} \setminus \{e\}$;
    $\psi(e) := i$;
    $i := (i+1)\ MOD\ T^*$;
end while
if $|\mathcal{N}| < T^*$ **then**
    return $|\mathcal{N}|$;
else
    return T^*;
end if

an edge $(n', n), n' \in \mathcal{D}_n$ 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 1. Both node A and C transmit in channel 0, causing a conflict at node B. In [8], 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 ($\mathcal{D}_n$) 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: $\mathcal{C}_e^{\mathcal{D}}$ includes all the conflicting nodes for a given node pair $e := (n', n)$ with $n' \in \mathcal{D}_n$, and $\mathcal{C}_n^{\mathcal{N}}$ includes all the conflicting node pairs for a given node $n \in \mathcal{N}$. The algorithm used to construct these sets for each node $n \in \mathcal{N}$ and each node pair (n', n) in the case of κ_{sinr} as the interference model is shown in Algorithm 2. In principle, the algorithm operates in two phases: first it assigns a conflict with node n for each of its decodables $n' \in \mathcal{D}_n$, and vice versa[4] (line 5 and 6 of Algorithm 2). In a second step the algorithm assigns a conflict with the node pair (n', n) for each interferer of n, and vice versa (line 13 and 14). The set of interferers of a node n in the case of κ_{sinr} is determined by gradually testing the interference model κ_{sinr} with an increasing set of interferers, starting with the node n' contributing the lowest signal power $P_{n \leftarrow n'}$. After the algorithm has finished, the sets $\mathcal{C}_e^{\mathcal{D}}$ and $\mathcal{C}_n^{\mathcal{N}}$ include all the conflicts of any node and node pair in the network. Assigning channels to nodes is then done in a *Greedy* fashion, except that instead of the neighborhood, the pre-computed conflicts are taken into account (see Algorithm 3[5]). The result of such a *Greedy*$^+$ channel assignment in a chain topology is shown in Figure 1. The *Greedy*$^+$ indeed finds a conflict free schedule also under *SINR* based interference. Please note that *Greedy*$^+$ will not be able to assign the minimum number of channels necessary for a given network configuration $\mathcal{N}, \mathcal{D}_n$ and a given load function μ.

[3]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

[4]Only nodes participating in communication are considered

[5]*Greedy*$^+$ assigns channels such that $B_{n',n} = 1, \forall (n', n) \in \Upsilon$

Algorithm 2 Algorithm to create a conflict mesh

Input: Nodes $\mathcal{N}$, decodables $\mathcal{D}_n$, load function μ, interference model κ
Output: Conflicts $\mathcal{C}_e^{\mathcal{D}}, \mathcal{C}_n^{\mathcal{N}}, \forall e := (n'', n')$ with $n'' \in \mathcal{D}_{n'}$

```
for all $n \in \mathcal{N}$ do
  for all $n' \in \mathcal{D}_n$ do
    if $\mu((n', n)) > 0$ then
      $e := (n', n)$;
      $\mathcal{C}_e^{\mathcal{D}} := \mathcal{C}_e^{\mathcal{D}} \cup \{n\}$;
      $\mathcal{C}_n^{\mathcal{N}} := \mathcal{C}_n^{\mathcal{N}} \cup \{e\}$;
      $\mathcal{I}^* := \emptyset$;
      L := sort($\mathcal{N} \backslash \{n, n'\}$) such that
          $n'' \prec n''' \longleftrightarrow P_{n \leftarrow n''} < P_{n \leftarrow n'''}$
      for $n'' \in L$ do
        if $n'' \neq n \wedge n'' \neq n'$ then
          $\mathcal{I}^* := \mathcal{I}^* \cup \{n''\}$
          if $\kappa_{sinr}(n', n, \mathcal{I}^*) \neq 1$ then
            $\mathcal{C}_e^{\mathcal{D}} := \mathcal{C}_e^{\mathcal{D}} \cup \{n''\}$;
            $\mathcal{C}_{n''}^{\mathcal{N}} := \mathcal{C}_{n''}^{\mathcal{N}} \cup \{e\}$;
          end if
        end if
      end for
    end if
  end for
end for
```

Algorithm 3 *Greedy*$^+$ channel assignment

Input: Set of nodes $\mathcal{N}$, decodables $\mathcal{D}_n$, Conflicts $\mathcal{C}_e^{\mathcal{D}}, \mathcal{C}_n^{\mathcal{N}}$, load function μ
Output: Channel assignment ψ and number of channels used

```
$\Pi := \emptyset$;
for all $n \in \mathcal{N}$ do
  for all $n' \in \mathcal{D}_n$ do
    $e := (n', n)$;
    for $i := 0;\ i < \mu(n', n)$ do
      $\Omega := \emptyset$;
      for all $n'' \in \mathcal{C}_e^{\mathcal{D}}$ do
        $\Omega := \Omega \cup \psi^*(n'')$;
      end for
      for all $e' \in \mathcal{C}_{n'}^{\mathcal{N}}$ do
        $\Omega := \Omega \cup \psi(e')$;
      end for
      $\gamma_i :=$ ANY $\gamma \notin \Omega$;
      Define $\psi(e) := \psi(e) \cup \{\gamma_i\}$;
      $\Pi := \Pi \cup \{\gamma_i\}$;
    end for
  end for
end for
return $|\Pi|$;
```

In fact, finding the minimum number – the so called *chromatic number* – of channels in a network refers to the graph coloring problem which is known to be NP-hard in most cases [8]. However, it is known that *greedily* assigning the colors is a good approximation for the chromatic number in random geometric graphs [7].

5. CAPACITY OF LARGER NETWORKS

In this section we analyze throughput capacity of various types of communication patterns and network topologies. To simplify the notation we will refer to $E[\zeta]$ as λ for the rest of the paper. For each analyzed configuration we also provide results taken from simulations with ns-2 [14] under the very same topology and communication setup. The idea is to compare real[6] 802.11 multihop 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 ϑ_{pl}, and a SINR based interference model, κ_{sinr}, as described in Equation 4. Since we use ϑ_{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 difficult to compare as the per-packet overhead dominates the overall cost. This is due to the fixed length 802.11 preamble used by the hardware for bit synchronization.

5.1 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.

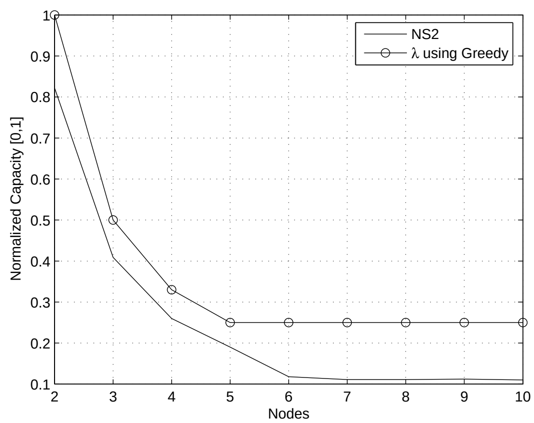
We use *Greedy*$^+$ as the channel assignment algorithm. Since there are no random components involved, λ is a direct function of the channels needed, and computes to 1/4 as the chain grows. From Figure 2a, we see that the value of λ lies above the throughput measured with ns-2, especially when the chain 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 [10], 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 $d(n) \in \mathcal{N} \backslash \{N\}$ to every node $n \in \mathcal{N}$. Figure 2b shows the effect of such a traffic pattern on throughput. The plot shows a quite close match between λ and the measurements obtained with ns2. This is not too surprising since we know from Figure 2a that the throughput of an 802.11 chain tallies 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. 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.

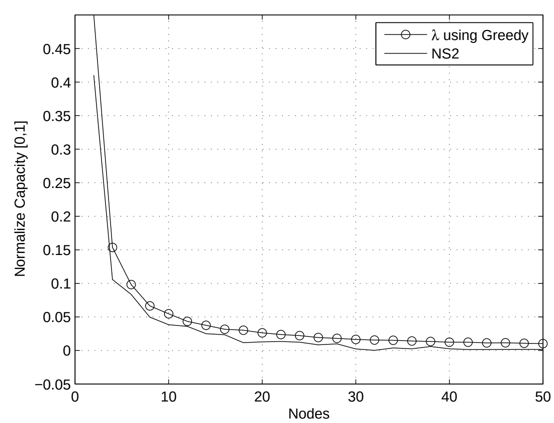
5.2 Grid

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[7] of all nodes constant (neglecting the border nodes), it seems natural that there is an optimum for the number of channels, almost independent of the size of the grid. Figure 3a shows the throughput capacity in a grid of 100 nodes using *RandomEdge* with a varying number of channels. Clearly,

[6]See [13] for a comparison of different network simulators.
[7]The size of the set $\mathcal{D}_n$

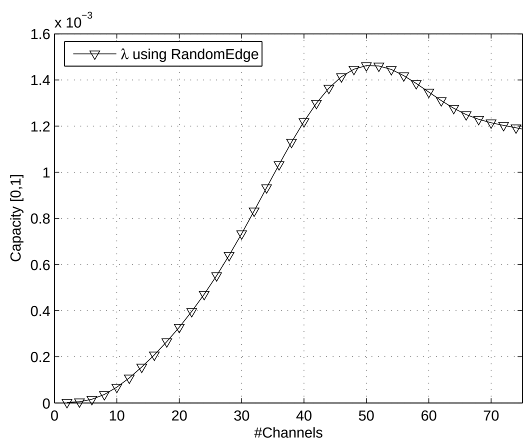


(a) Single end-to-end flow

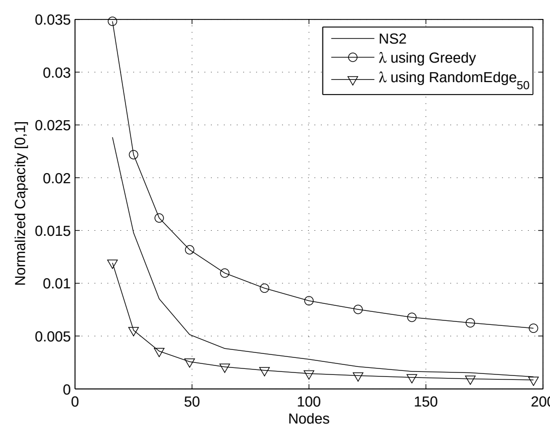


(b) Random communication

Figure 2: Chain Topology



(a) Number of channels vs throughput of *RandomEdge* channels assignment in a grid with 100 nodes



(b) Grid size vs throughput capacity

Figure 3: Grid topology and random traffic

the throughput capacity is maximized when using around 50 channels in total. In Figure 3b, 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 explanation is that the two dimensions 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 30 channels for every grid configuration, which is also far less than the optimum of 50 channels used in *RandomEdge*.

5.3 Random Topology

We consider random topologies of n nodes distributed uniformly within an area of 1000×1000 meters. As in the previous topologies, all nodes have β_D 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 $\mathcal{N} \setminus \{n\}$. Figure 4 shows the throughput capacity λ in contrast with ns-2 simulation measurements. The result confirms the trend already observed in the previous configurations of the chain and the grid: randomness improves 802.11 throughput capacity with respect λ. This might be particularly the case in random networks where the node density augments 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.

6. 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

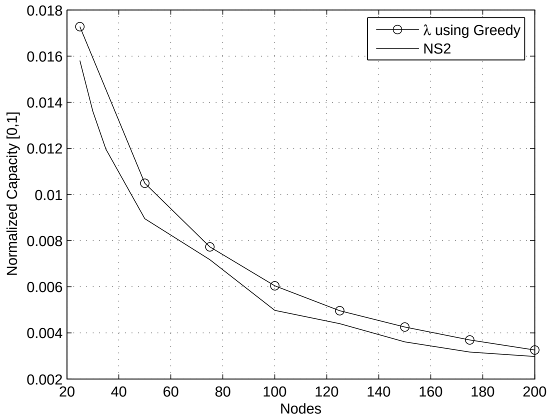


Figure 4: Random topology

(isotropic radio propagation, unidirectional links, perfect scheduling, straight line routing, etc.) and asymptotic bounds, our approach allows for capacity analysis 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. Due to the limited space we couldn't show how the model can be applied to investigate, e.g., optimal transmission ranges or effects of randomized radio propagation. These results can however be obtained in [12].

Acknowledgement

The work presented in this paper was supported (in part) by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322.

7. REFERENCES

[1] Process support for more than web services. http://www.iks.ethz.ch/jopera.

[2] M. Alicherry, R. Bhatia, and L. E. Li. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 58–72. ACM Press, 2005.

[3] P. Bjorklund, P. Varbrand, and D. Yuan. Resource optimization of spatial tdma in ad hoc radio networks: A column generation approach. In *Proc. of IEEE INFOCOM*, March 2003.

[4] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[5] O. Dousse and P. Thiran. Connectivity vs capacity in dense ad hoc networks. In *Proc. of IEEE INFOCOM*, 2004.

[6] J. Gomez and A. Campbell. A case for variable-range transmission power control in wireless ad hoc networks. In *Proc. of IEEE INFOCOM*, March 2004.

[7] P. Gupta and P. Kumar. Capacity of wireless networks. *Information Theory*, 46(2):388–404, 2000.

[8] S. Krumke and M. Marathe. Models and approximation algorithms for channel assignment in radio networks. *Wirel. Netw.*, 7(6):575–584, 2001.

[9] P. Kyasanur and N. H. Vaidya. Capacity of multi-channel wireless networks: impact of number of channels and interfaces. In *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 43–57. ACM Press, 2005.

[10] J. Li, C. Blake, D. S. D. Couto, H. I. Lee, and R. Morris. Capacity of ad hoc wireless networks. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 61–69. ACM Press, 2001.

[11] T. S. Rappaport. *Wireless Communications: Principles & Practice*. Prentice Hall, 2002.

[12] P. Studi and G. Alonso. Computing throughput capacity for realistic wireless multihop networks. In *Technical Report 527, Department of Computer Science, ETH Zurich*, 2006.

[13] M. Takai, J. Martin, and R. Bagrodia. Effects of wireless physical layer modeling in mobile ad hoc networks. In *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, pages 87–94. ACM Press, 2001.

[14] The VINT Project. The NS network simulator.

[15] S. Toumpis and A. Goldsmith. Capacity regions for wireless adhoc networks. In *International Conference on Communications (ICC)*. IEEE, 2002.

[16] M. Zorzi and S. Pupolin. Optimum transmission ranges in multihop packet radio networks in the presence of fading. *IEEE Trans. On Communication*, 43(7), July 1995.