Congestion Control in CSMA-Based Networks with Inconsistent Channel State

Violeta Gambiroza and Edward W. Knightly*
Department of Electrical and Computer Engineering
Rice University
Houston, TX 77005, USA
Email: {violeta, knightly}@rice.edu

ABSTRACT

In this paper, we study the performance of utility maximization congestion control over multihop CSMA-based networks. We consider decoupled vs. joint design of congestion control and medium access and consider unmodified MAC protocols such as IEEE 802.11. Networks employing such MAC protocols incur flow *starvation* both without congestion control and with existing TCP-based congestion control. We develop a framework to study key issues in such networks that are not incorporated by prior models, yet are critical to the performance of congestion control algorithms. We study the role of data transmission capacity that is location dependent and, even more, unknown. We show that for the case of *consistent* channel state, a single globally optimal data transmission capacity does not exist. Moreover, for the case of *inconsistent* channel state that arises due to the carrier sense mechanism itself, a data transmission capacity that provides convergence to perfectly fair rates does not exist, i.e., the congestion control algorithm converges to incorrect rates. We study the impact of inter-node collaboration within a contention region, and show that collaboration can alleviate these problems and ensure convergence to fair rates. Finally, we compare the performance of congestion control in a collaborative network with the performance of TCP, and show that TCP starves some flows, whereas congestion control with collaboration removes starvation, provides significantly better fairness, and achieves 17% higher aggregate throughput.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Design, Performance

*This research is supported by NSF ITR Grants ANI-0331620 and ANI-0325971, and by the Cisco ARTI program.

WICON '06, Aug 2–5, 2006, Boston, MA.

Keywords

CSMA, CSMA/CA, Congestion Control, Starvation

1. INTRODUCTION

Flows can experience severe unfairness and even starvation in multihop wireless networks employing variants of CSMA protocols [1, 2, 3, 4, 5]. The origin of this behavior is inconsistent channel state, i.e., in multihop topologies, different nodes have different observable views of the system state. These different views can lead to starvation of a sending node when, for example, the sending node cannot infer another flow's transmission status, resulting in repeated backoffs for the starving flow. Models were developed in [2, 3] that incorporate these effects and predict each flow's throughput.

Because this problem is most severe when all flows are fully backlogged, use of *congestion control* (such as reduction of the input rates of dominating flows) has the potential to alleviate starvation. For example, throttling the input rates of dominant flows yields sufficient spare capacity for otherwise-starving flows. However, prior work on congestion control does not incorporate inconsistent state, starvation, and other critical CSMA-related behaviors, and therefore does not address how MAC related problems manifest in congestion control.

In this paper, we study utility-maximization congestion control in networks with unmodified CSMA-based MAC protocols. Our problem formulation of an "unmodified" MAC is a practical constraint prohibiting changing the MAC protocol (e.g., IEEE 802.11) in any way. This contrasts with joint design in which congestion control and medium access are designed together, e.g., [6]. We study the effects of incorrect feedback information due to channel inconsistency associated with CSMA on the performance of utility maximization congestion control. We study scenarios with single hop flows that can be seen as a part of more complex multihop scenarios. Our contributions are as follows.

First, we present a framework for congestion control algorithm design that incorporates key issues encountered in CSMA-based multihop wireless networks. In particular, data transmission capacity is a critical input to utility maximization based congestion control algorithms. While transmission capacity in *wireline* networks is immediate, the scenario we consider has collisions, channel arbitration overhead, etc., all of which depend on spatial location. Thus, the data transmission capacity in the scenario we consider is *unknown*. The framework also incorporates the roles of packet service order, which can deviate significantly from first-come first-serve due to MAC behavior, and state observation, including inference of channel state that is inconsistent among nodes sharing spectral resources.

Second, we analyze topologies with unknown data transmission capacity, yet *consistent* channel state. In this way, we isolate the impact of unknown capacity from the issue of inconsistent state observation. We find the optimal data transmission capacity that provides the best throughput and fairness. Unfortunately, we show that the optimal data transmission capacity is topology dependent, i.e., even in networks with a homogeneous propagation environment, there is no globally optimal value that provides the best performance in all topologies. The key issue is that the source nodes may have consistent channel state, and at the same time this state can be incorrect. For example, source nodes can measure the same average fraction of busy time that is not the true fraction of busy time of the contention neighborhood they belong to.

Next, we study the role of unknown data transmission capacity in topologies with *inconsistent* channel state. If all flows are backlogged, the service order in these topologies can be close to strict priority, with one flow taking most of the bandwidth [3]. The key issue is that due to carrier sense, there is asymmetry in knowledge of the channel state among the senders. Thus, senders with better knowledge of the channel state know when to contend for the channel while senders with less information contend randomly, and suffer tremendously in terms of service order and thus throughput. We show that in such topologies, there is no data transmission capacity that provides perfect fairness, despite the fact that flows belong to the same clique. We also show that this is the case regardless of the magnitude of the difference in channel state. In other words, even if there exists a small but constant difference in channel state, the congestion control algorithm converges to incorrect rates.

Finally, we study the impact of node collaboration on the performance of the congestion control algorithm in multihop topologies. A network with collaboration provides a mechanism to support inter-node collaboration in order for nodes in the same contention region to obtain the correct channel state. We show that it is critical to performance to obtain per-flow information instead of aggregate information, since aggregate information can lead to convergence to a false stable operating point. We also compare the performance of such collaborative congestion control with TCP and show that TCP fails to provide fairness and leads to starvation of some flows. In contrast, the congestion control algorithm that maximizes the sum of utility functions in a network that enables collaboration among nodes prevents starvation, achieves fairness, and achieves 17% higher aggregate throughput than TCP.

The reminder of this paper is structured as follows. Section 2 provides background on the utility maximization framework for congestion control in wireless networks. Section 3 presents the framework for algorithm design in a CSMA environment as well as performance metrics and our simulation environment. Section 4 provides a performance evaluation for topologies with *consistent* channel state and no inter-node collaboration, and topologies with *inconsistent* channel state and no inter-node collaboration is studied in Section 5. We study collaboration in Section 6. Finally, related work is discussed in Section 7 and Section 8 concludes the paper.

2. BACKGROUND

We first review utility maximization congestion control in the commonly considered case of a wireless network with a perfect collision-free channel and a single transceiver utilizing a single channel. (Section 7 presents a complete discussion of prior work.) Such a wireless network consists of overlapping contention neighborhoods. We define a contention neighborhood as a subset of links that satisfy two properties: first, no two links from the subset can be active simultaneously, and second, there is no other link in the network such that by adding it to the subset, the first property is preserved. The contention neighborhood defined as above includes both transmission and interference ranges. Note that a single link can belong to multiple contention neighborhoods. Let C_l be the capacity of link l when active in isolation, and L_n the set of all links in contention neighborhood n. Denote set F as the set of all flows in the network, and L_n^f as the set of links in contention neighborhood n traversed by flow f. Assume that each flow f transmits at rate x_f. Since flows within the same contention neighborhood cannot transmit simultaneously, for each contention neighborhood n we can write

$$\tau_n = \sum_f \sum_{l \in L_n^f} \frac{x_f}{C_l} \le 1. \tag{1}$$

We make the following three observations. First, the above inequality provides only a necessary condition for the feasibility of rates x_f (see [7]). However, if the contention graph [8] is perfect [9], the above condition is also a sufficient feasibility condition (see [10], also cited in [7, 6]). Second, for a perfect contention graph and the IEEE 802.11 MAC, we have that $\tau_n < 1$ due to contention, overhead such as RTS/CTS exchange, SIFS, DIFS etc. Finally, observe that $\frac{x_f}{C_l}$ is the fraction of time needed for amount x_f of flow f to be transmitted over link l with capacity C_l.

Next, we define the congestion control algorithm as an optimization problem such that the goal is to maximize the sum of utility functions. Later, we study the performance of this congestion control algorithm over wireless networks with IEEE 802.11 DCF MAC.

Assume that each flow f obtains a utility $U_f(x_f)$ when it transmits at rate x_f. As in [11], we assume that U_f is strictly concave, continuously differentiable and increasing, and that utilities are additive. Our objective is to determine rates x_f such that [6, 11]

$$\max_{x_f \ge 0} \sum_f U_f(x_f) \tag{2}$$

$$\text{s.t. } \tau_n \le 1, \forall n. \tag{3}$$

Since the objective function is strictly concave and the feasible region is convex, we know that a unique maximizer exists. However, the solution to this maximization problem assumes that sources know all utilities, which is impractical, and requires coordination among possibly all sources. Thus, a distributed algorithm can be devised by considering a relaxation of the original problem [11]:

$$\max_{x_f \ge 0} \Lambda(x) \tag{4}$$

where

$$\Lambda(x) = \sum_f U_f(x_f) - \sum_n \int_0^{\tau_n(x)} p_n(v)dv, \tag{5}$$

$\tau_n(x) = \sum_f \sum_{l \in L_n^f} \frac{x_f}{C_l}$ is the fraction of time contention neighborhood n is active (i.e., there is an ongoing transmission), and $p_n(x)$ is a price function that can be viewed as the price charged by contention neighborhood n for transmission. We assume that $p_n(x)$ is a continuous, non-negative, increasing function of x, and is not identically zero.

It has been shown in [6] that under the above assumptions, $\Lambda(x)$ defined by Equation (5) is strictly concave. From concavity, it directly follows that the above relaxation defined with (4) has a unique maximizer. We calculate the optimal rates solving the following differential equations,

$$\frac{\partial \Lambda}{\partial x_f} = 0,\ f \in F. \tag{6}$$

Thus we obtain

$$U'_f(x_f) - \sum_n \sum_{l \in L_n^f} p_n(\tau_n(x)) \frac{1}{C_l} = 0. \quad (7)$$

Applying the gradient method to (4)-(5), the following congestion control algorithm can be defined

$$\dot{x}_f(t) = k_f(U'_f(x_f) - q_f(t)), f \in F, \quad (8)$$

where k_f is a positive constant and $q_f = \sum_n \sum_{l \in L_n^f} p_n(\tau_n(x)) \frac{1}{C_l}$. Note that $q_f(t)$ is the feedback that the source of flow f observes. The price $p_n(\tau_n)$ can be interpreted as a measure of the congestion in contention neighborhood n when the total channel busy time is τ_n [11]. Therefore, $q_f(t)$ represents the congestion in all congestion neighborhoods traversed by flow f.

It has been shown in [11] that for any initial condition $x_f(0)$, the congestion control algorithm (8) converges to the unique solution of the relaxation defined in (4).

We note that the algorithm defined by Equation (8) provides an approximate solution to the original problem (2)-(3). However, by choosing the appropriate price function $p(\cdot)$, the original problem (2)-(3) can be solved exactly [12]. We discuss the choice of the price function in the next subsection.

3. FRAMEWORK AND PERFORMANCE METRICS

The utility-based optimization framework was originally designed for wireline networks [11]. The same framework has been applied to study congestion control over wireless networks as described in Section 2 [6]. Here, we describe decoupled vs. joint congestion control and MAC design, and describe the issues and challenges such designs encounter in wireless networks in which the MAC protocol is a CSMA variant such as the IEEE 802.11 distributed coordination function (DCF) [13].

3.1 Joint vs. Decoupled Design

As described above, utility maximization congestion control is based on feedback characterized by the price function $p_n(t)$. The price is a function of $\tau_n(t)$, which can be described as the fraction of time contention neighborhood n is busy due to transmissions. Therefore, congestion control solutions can be classified according to the way the congestion price is realized.

In *joint design* of congestion control and medium acces, the congestion price is generated directly from the multiple access scheme. An example of this approach is [6]. In *decoupled design*, the MAC algorithm is strictly defined and cannot be modified. Thus, the price function is specified in accordance with this unmodified MAC. Ideally, the price function in this case is chosen such that the approximation defined by Equation (8) provides the same solution as the original problem (2)-(3).

In this paper we consider *decoupled design* and study CSMA-based MAC protocols such as IEEE 802.11 that cannot be modified. We use the following price function,

$$p_n(x) = (\tau_n(x) - 1 + \epsilon)^+ / \epsilon^2. \quad (9)$$

In [11, 12] it has been shown that as $\epsilon \to 0$, the approximation is arbitrarily close to the exact solution of the original problem defined by (2)-(3). Note that a node within a contention neighborhood n needs to obtain the value of τ_n. As we discuss below, obtaining the correct τ_n represents a challenge in multi-hop wireless networks.

We also consider a hop-by-hop scheme in which the feedback is realized based on nodes' perception of congestion that is advertised to an upstream node. In the next section, we argue that a node's perception of congestion in CSMA-based networks is not necessarily correct, and in subsequent sections we study how feedback based on such incorrect congestion inferences affects the performance of the congestion control algorithm.

Finally, without loss of generality, we assume that the utility U_f of a flow f is defined as $U_f(x_f) = w_f \log x_f$, which enforces weighted proportional fair resource allocation.

3.2 Algorithm Design Framework

Channel State in Multihop Networks. In a multihop topology in which all nodes are not within radio range, different nodes can have different views of the channel. In particular, even if a set of nodes are within a single *contention neighborhood*, they do not necessarily have the same views of the channel. For example, in Figure 1(a), the two depicted flows share a contention neighborhood, yet the sender of flow one does not observe flow two's transmission. On the other hand, the sender of flow two observes flow one, such that the two senders have different channel state.[1]

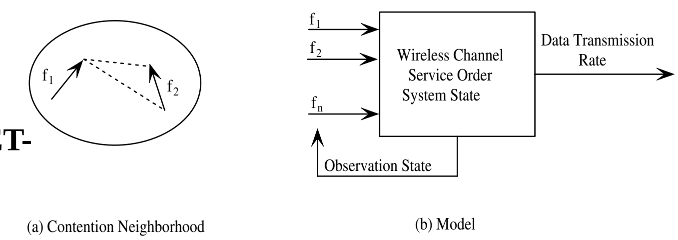


Figure 1: System Model Illustration

This inconsistency in channel state is not addressed in the utility-based optimization framework presented in Section 2 and the congestion control algorithm from Equation (8). In particular, in that formulation, all source nodes need to have the same information about the channel state expressed as the channel busy time of contention neighborhood n (τ_n) to achieve convergence to the fair rates. Therefore, we study the impact of inconsistent channel state on the performance of congestion control algorithms.

Data Transmission Capacity. The actual capacity available for data transmission within a contention neighborhood depends on many factors such as the number of competing flows, node locations, and the propagation environment. Moreover, due to the random nature of the MAC protocol, a portion of resources (e.g., time, bandwidth) cannot be used for data transmission. For example, during collision resolution, there are time instances in which the channel is idle even though flows are backlogged.

Consequently, because of these dependencies on numerous dynamic factors, the data transmission capacity cannot be determined in advance. Thus, the constraint defined by Equation (3) cannot be determined in advance. Therefore, we define the Efficiency Index $0 \leq \gamma_n \leq 1$, which can be viewed as the *net* data transmission capacity, expressed in time, of contention neighborhood n. Then, instead of the original constraint defined by (3), we consider the constraint $\tau_n \leq \gamma_n, \forall n$. Observe that γ can have different values depending on the MAC layer, e.g., a MAC layer that uses time division multiple access (TDMA) would result in $\gamma = 1$. However, the IEEE 802.11 MAC layer will always result in $\gamma < 1$. Moreover, we show that the IEEE 802.11 MAC layer in different topologies

[1] In our terminology, the channel state at a particular location refers to both physical-layer state such as SNR as well as MAC-layer state such as carrier-sense state.

results in different γ. Therefore, the price function (9) becomes

$$p_n(x) = (\tau_n(x) - \gamma_n + \epsilon)^+/\epsilon^2, \tag{10}$$

where γ_n is the Efficiency Index of contention neighborhood n. Because of its critical role in describing system resources, throughout this paper we study the performance of the congestion control algorithm as a function of γ. We show that γ plays a critical role in topologies with inconsistent channel state.

Service Order. Because nodes that share a single contention neighborhood contend for the same channel, all of the local queues in the contention neighborhood can be considered as a *distributed* queue for that contention neighborhood, as depicted in Figure 1(b). The service order of this distributed queue is in general *not* FCFS and can diverge from FCFS as far as strict priority. Moreover, the actual service order is unknown and depends on the topology, number of nodes, nodes' transmission and carrier sense ranges, etc.

For example, it is well established in the literature that 802.11 suffers from severe unfairness and starvation in the presence of hidden terminals [1] and information asymmetry [1, 8, 14], for uncontrolled UDP or TCP traffic [4]. The service order in topologies with hidden terminals or information asymmetry is very close to strict priority, i.e., one flow achieves a throughput equal to its input rate, while the other flows, even if backlogged, achieve only a fraction of the remaining resources. While it may be possible to design a congestion control algorithm to achieve fairness in a network with strict priority service, our scenario imposes a further design issue. In particular, the service order here is unknown in advance, is topology dependent, and spans orders ranging from FCFS to strict priority.

State Observation and Sharing. To deploy the congestion control algorithm defined by (8), each source node needs to obtain the fraction of "busy" time $\tau_n(x)$ for all contention neighborhoods n it belongs to. Moreover, in some realizations, each source node needs to obtain the capacity of the links between the source and destination. In our formulation, a source node can locally observe multiple metrics, such as channel occupancy time, flow throughput, or its input rate. Thus, we evaluate three different metrics to obtain the needed fraction of busy time, namely direct measurement of channel busy time, the sum of the ratios of measured throughputs over measured capacities, and the sum of measured input rates over measured capacities.

There are multiple ways a source node can obtain the fraction of busy time. In this paper, we classify all approaches according to whether or not nodes *collaborate*. Without collaboration, a source node uses only information available locally to infer the shared state of the channel, whereas with collaboration, nodes within a contention neighborhood exchange messages in order to obtain the same correct channel state.

3.3 Example

To illustrate this design framework, consider the scenario shown in Figure 2. In this example, nodes A, B, C and D use their own view of the channel to calculate the congestion feedback. As described above, a node's local view of the channel is not necessarily correct. For example, node B (depending on its spatial location) may be unaware of D-E transmissions. This leads to node B's perception of congestion being incorrect, and thus node's B feedback being incorrect.

As described in Section 2, the congestion control mechanism adjusts source rates based on the congestion sources perceive. Note that the way the congestion control algorithm is defined allows for some flexibility in the choice of parameters used to describe congestion. For example, the joint MAC and congestion control algorithm proposed in [6] uses only channel busy time τ_n to describe congestion. However, unlike this paper, [6] derives the price function directly from the MAC algorithm, and does not consider IEEE 802.11 media access. Similarly, [7] uses the number of transmissions in a time interval to describe congestion, and studies IEEE 802.11 media access. Here, in order to ensure that the algorithm in (8) solves the original problem (2)-(3) exactly, we define the price function by (10), thus we describe congestion of a particular contention neighborhood n using two parameters, namely channel busy time τ_n and Efficiency Index γ_n. In subsequent sections, we show that both parameters are critical in scenarios which do not employ inter-node collaboration to realize channel state.

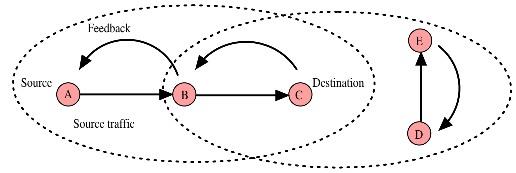


Figure 2: Illustration of Feedback Based on Inaccurate Information

3.4 Performance Metrics and Simulation Environment

3.4.1 Performance Metrics

To characterize the performance of a congestion control algorithm, we use throughput, short-term fairness index and long-term fairness index. As a measure of long-term fairness, we define the Kullback-Leibler (KL) Fairness Index using KL distance, similar to [15]. KL distance (or relative entropy) is a distance from a "true" probability distribution, p, to a "target" probability distribution, q, and is denoted as $D(p\|q)$. It is a measure of "the inefficiency of assuming that the probability distribution is q when the true distribution is p" [16]. Thus, if we define vectors Θ and $\tilde{\Theta}$ as vectors of achieved and ideal fair shares respectively, then we can define the KL Fairness Index as follows:

$$\begin{aligned} D(\Theta\|\tilde{\Theta}) &= D([\theta_1, \theta_2, ..., \theta_N]\|[\tilde{\theta_1}, \tilde{\theta_2}, ..., \tilde{\theta_N}] \\ &= \sum_{i=1}^{N} \theta_i log_2 \frac{\theta_i}{\tilde{\theta_i}}, \end{aligned} \tag{11}$$

where N is the number of flows in the network. Note that KL fairness index 0 indicates perfect fairness.

We then use the sliding window technique to measure short-term fairness. In other words, for a given window size we calculate the fairness index within that window. Then, as we slide the window, one element at the time, we obtain a series of fairness indices. Finally, after sliding the window through the entire sequence, we find the average of all fairness indices.

3.4.2 Simulation Environment

Our evaluation of congestion control algorithms uses extensive simulation experiments. Unless explicitly mentioned, all simulation experiments use the configuration described here. As the simulation platform we use *ns-2* version 2.27. Most parameters are default values: the channel rate is constant and has 2 Mb/sec capacity, the channel propagation model is the two-way ground reflection model, and the transmission range is 250 m. The MAC protocol is IEEE 802.11 DCF. At the beginning of a simulation, all source nodes start transmitting UDP traffic with 1000 byte packets at the full link rate. Rates are updated periodically every 10 msec. The buffer size at each node is 50 packets, and the simulation time is 250 sec. We set all weights to be 1, while the parameter ϵ in the price function we set to 0.005.

4. CONGESTION CONTROL WITH CONSISTENT STATE

Previous work either ignores the issue of unknown data transmission capacity, or acknowledges it but does not study what the ideal data transmission capacity should be. Here, we first study data transmission capacity in terms of the Efficiency Index γ in topologies with consistent state and no inter-node collaboration. We show that γ is critical to performance as it significantly impacts short-term fairness and controls throughput. Moreover, we show that, unfortunately, there is no single γ that is optimal in all topologies, i.e., the γ that provides high throughput and ideal fairness is topology dependent.

In this section, we consider topologies with the property that all source nodes have consistent state, in other words on average they measure the same channel busy time. We name these topologies as symmetric topologies. Observe that the source nodes may have consistent channel state, and at the same time this state can be incorrect. In other words, source nodes can measure on average the same fraction of busy time that is not the true fraction of the busy time of the contention neighborhood they belong to. Therefore, we classify symmetric topologies in two classes, fully connected topologies and symmetric incorrect state topologies. All source nodes in fully connected topologies have the same correct channel state, whereas source nodes in symmetric incorrect state topologies have the same, however, incorrect, channel state.

4.1 Fully Connected Topologies

We first address fully connected topologies in which all source nodes are able to locally and independently measure the same correct fraction of the busy time. While fully connected topologies have been studied in the literature (e.g., see [17]), we study them here as a baseline to illustrate the effect of the Efficiency Index γ on the throughput and fairness properties of the utility maximization congestion control algorithm. A fully connected topology for two flows is shown in Figure 3.

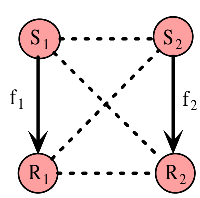


Figure 3: Fully Connected Topology

Using the utility maximization problem defined by (2) and assuming a perfect MAC, the congestion control algorithm is an approximation to the following problem

$$\max_{x_1,\ldots,x_n \geq 0} \sum_{i=1}^{n} w_i \log x_i \qquad (12)$$

$$\text{s.t.} \sum_{i=1}^{n} \frac{x_i}{C_i} \leq \gamma. \qquad (13)$$

Using Equation (7) we have that x_i, $i = 1, ..., n$ are solutions to the following system of equations:

$$\frac{w_i}{x_i} = \frac{(\sum_{i=1}^{n} \frac{x_i}{C_i} - \gamma + \epsilon)^+}{C_i \epsilon^2}, i = 1, ..., n \qquad (14)$$

When $w_i = 1$, $C_i = C$ and $\epsilon \rightarrow 0$ we have

$$x_i = \frac{\gamma C}{n}, i = 1, ..., n.$$

We make the following observations. First, the allocated rates of all n flows are the same so that fairness is achieved. Next, the allocated rates are a linear function of the Efficiency Index γ, indicating that the total throughput can be controlled as a linear function of γ. Therefore, the optimal γ that provides the maximum throughput and the best fairness in fully connected topologies is $\gamma_{opt}^{FC} = 1$. Next, we use simulations to validate these results and to study the throughput and fairness properties of fully connected topologies.

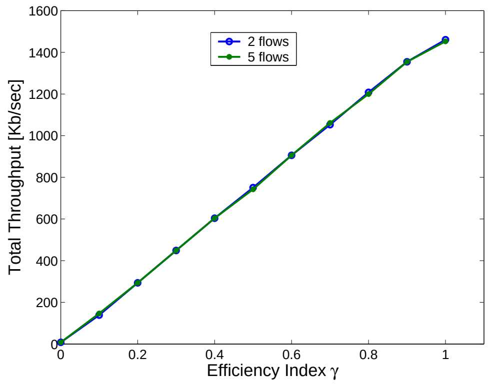
Throughput. Figure 4(a) depicts total throughput vs. Efficiency Index γ for two- and five-flow fully connected topologies. For ease of comparison, we report the results for total throughput, and note that the per-flow throughput for fully connected topologies is the total throughput divided by the number of flows. In all simulations, the algorithm converged so that the sum of utilities is maximized. Both curves show the linear dependency on γ as predicted by the above analysis, and that the throughput achieves a maximum of 1.46 Mb/sec for $\gamma = 1$. Moreover, the curves are nearly identical because we consider the total throughput, and not per-flow throughput.

Fairness properties. Next, we study fairness properties of fully connected topologies. We find that the long term fairness index is always 0 (i.e., perfect fairness) and does not depend on γ. However, the short-term fairness index depends on both γ and the window size. Figure 4(b) depicts the short-term KL fairness index as a function of window size for two different values of $\gamma = 0.9$ and 1. Observe that when $\gamma = 1$, the window size needs to be more than double the window size for the case when $\gamma = 0.9$ to achieve the same short-term fairness of 0.05. At the same time, the throughput for $\gamma = 1$ is 1.46 Mb/sec, whereas the throughput for $\gamma = 0.9$ is 1.35 Mb/sec. Finally, we note that the short-term fairness properties for $\gamma < 0.9$ are similar to those for $\gamma = 0.9$, therefore, we omit them from the figure.

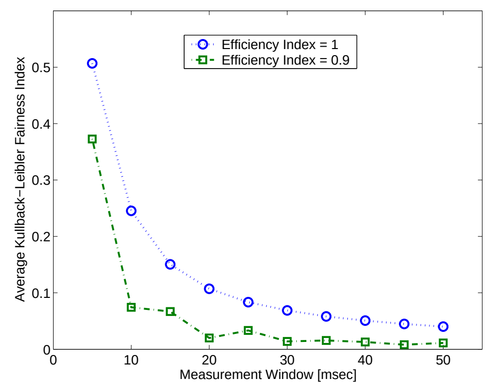
Figures 4(a) and 4(b) point out the tradeoff between throughput and short-term fairness. For example, by setting $\gamma = 1$ maximum throughput can be achieved, yet short-term fairness deviates considerably from the perfect value. Thus, to achieve a satisfactory short-term fairness, one needs to "sacrifice" throughput. This problem is even more pronounced in symmetric incorrect state topologies that we investigate next.

4.2 Symmetric Incorrect State Topologies

In this section, we study Symmetric Incorrect State (SIS) topologies without inter-node collaboration. The key issue in these topologies is that, due to symmetry, all source nodes on average measure the same fraction of the channel busy time, yet this measurement does not correspond to the actual value of the contention neighborhood's busy time. These topologies occur when receivers are in transmission range of one another while senders are out of range of senders and receivers of the other flows. Thus, the local inference of the channel state and busy time at each sender does not reveal the actual channel busy time. At the same time, due to the symmetric geometric relationship among flows, flows measure the same incorrect channel busy time on average. Observe that [17] cannot predict the throughput for this topology since it applies only to fully



(a) Total Throughput vs. Efficiency Index



(b) Short-Term Fairness Index

Figure 4: Fully Connected Topology: Simulation Results

connected topologies. The throughput properties of incorrect state two-flow topologies in which flows are continuously backlogged have been studied in [3].

Figure 5: Symmetric Incorrect State Topology

One such topology is presented in Figure 5. Observe that senders are in transmission ranges of their respective receivers only. Here, we study the impact of incorrect, but on average the same, channel state on the throughput and fairness properties of SIS topologies in the presence of congestion control without collaboration.

Since the senders use the information available locally to infer the channel busy time, the congestion control algorithm is not the approximation to the original problem as stated in Equations (12)-(13), but the following problem instead

$$\max_{x_1,\ldots,x_n \geq 0} \sum_{i=1}^{n} w_i \log x_i \tag{15}$$

$$\text{s.t. } \frac{x_i}{C_i} \leq \gamma \,, \forall i = 1, ..., n. \tag{16}$$

This is due to the fact that sources S_i, $i = 1, ..., n$ are in ranges of their respective receivers only, and thus are unable to locally measure the busy time of the other flows. Using (8), the x_i, $i = 1, ..., n$ are the solution to the following system of equations

$$\frac{w_i}{x_i} = \frac{(\frac{x_i}{C_i} - \gamma + \epsilon)^+}{C_i \epsilon^2}, i = 1, ..., n. \tag{17}$$

Again, when $w_i = 1$, $C_i = C$, and $\epsilon \to 0$ we have

$$x_i = \gamma C, \; i = 1, ..., 1.$$

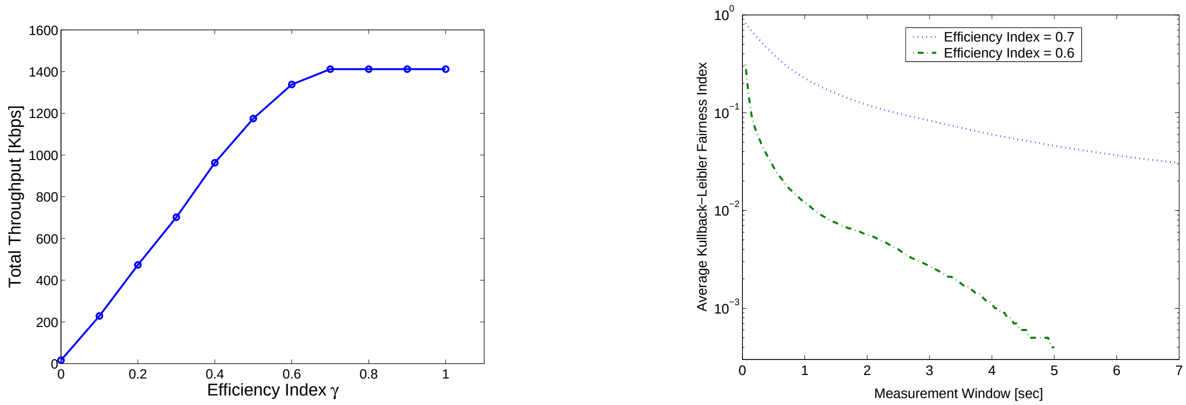
Assuming a perfect MAC, the above solution will be feasible only for $\gamma \leq \frac{1}{n}$, and the maximum throughput is achieved for $\gamma = \frac{1}{n}$. Thus, further increasing γ results in an infeasible solution, and no throughput increase. Moreover, because the allocated rates are the same, fairness is achieved for $\gamma < \frac{1}{n}$. However, for $\gamma > \frac{1}{n}$, the same allocated rates and the symmetric geometric relationship between the flows provide long-term fairness but not short-term fairness. Therefore, the optimal Efficiency Index is $\gamma_{opt}^{SIS} = \frac{1}{n}$. Finally, observe that $\gamma_{opt}^{SIS} \neq \gamma_{opt}^{FC}$, hence there is no single globally optimal γ_{opt}. Next, we use simulations to study throughput and fairness properties in the two-flow SIS topology from Figure 5.

Throughput. Results for total throughput vs. the Efficiency Index γ are shown in Figure 6(a). In all simulations the algorithm converges and the sum of utilities is maximized. We observe that the dependency between throughput and γ for SIS topologies consists of two segments. In the first segment for $\gamma \leq 0.6$, throughput increases linearly as γ increases. Due to MAC imperfections, the actual value of the "knee" is approximately 0.6. The slope of the increase is twice the slope for the case of the fully connected topology. In the second segment for $\gamma > 0.6$, the throughput is constant at 1.41 Mb/sec, and does not depend on γ. Therefore, in order to maximize throughput, the ideal setting of the Efficiency Index is $\gamma > 0.6$.

Finally, observe that the maximum throughput is lower than for the case of fully connected topologies. The main reason is the increased contention that we explain as follows. Assume that there is an ongoing transmission between S_1 and R_1 in Figure 5. Since S_2 is unaware of this transmission it can possibly send an RTS message to R_2. At the same time, R_2 is aware of the transmission and does not reply with a CTS message. Consequently, S_2 increases its contention window and enters backoff. At the same time, S_1 is more likely to acquire the channel in the next transmission attempt, and consequently, to hold the channel for long periods of time. This increase of the contention window leads to a decrease in throughput, while the long periods of time in which the channel is occupied by a single flow leads to serious short term unfairness that we explore next.

Fairness Properties. Figure 6(b) reports the short-term fairness index vs. window size for two values of the Efficiency Index γ. We observe that short-term fairness properties are extremely different for $\gamma = 0.6$ and for $\gamma = 0.7$. When $\gamma = 0.6$ a satisfactory fairness index of 0.05 can be achieved at time scales less than 300 msec, whereas when $\gamma = 0.7$ this same index can only be achieved at much longer time scales of 4.7 sec. On the other hand, the total throughput when $\gamma = 0.6$ is 1.33 Mb/sec, while the throughput when $\gamma = 0.7$ is 1.41 Mb/sec, a 6% difference. This characterizes the strong tradeoff between short-term fairness and throughput loss. We note that short-term fairness properties for $\gamma < 0.6$ and $\gamma > 0.7$ are similar to those shown in Figure 6(b) for $\gamma = 0.6$ and $\gamma = 0.7$ respectively, and are therefore omitted from the figure. Finally, the long-term fairness index is always 0 (i.e., perfect fairness) and does not depend on γ.

Thus, in SIS topologies, the optimal Efficiency Index γ_{opt}^{SIS} that



(a) Total Throughput vs. Efficiency Index

(b) Short-Term Fairness Index

Figure 6: Symmetric Incomplete Information Topology: Simulation Results

provides the highest throughput and satisfactory short-term fairness properties is always less than 1. In other words, for the price of a moderate throughput loss, short-term fairness properties can be substantially improved.

5. CONGESTION CONTROL WITH INCONSISTENT STATE

In this section, we consider topologies in which source nodes do not have the same channel state due to their spatial locations, transmission ranges and carrier sense ranges. We refer to topologies in which source nodes have inconsistent channel state as asymmetric topologies, as there is an asymmetry in knowledge of the channel busy time among the senders. Asymmetric topologies with fully backlogged flows have been studied in the literature [1, 3, 14], and severe MAC unfairness was established. Even more, it has been shown that the service order in such topologies is close to strict priority (SP), in which packets of one flow are nearly always served before packets of another flow, i.e., the "low priority" flow only obtains service if the "high priority" flow is not backlogged. Here, we study the impact of this unfairness and unknown data transmission capacity on the performance of the congestion control algorithm in asymmetric topologies with no inter-node collaboration. We show that the asymmetry in channel state is critical to performance and leads to convergence to incorrect rates.

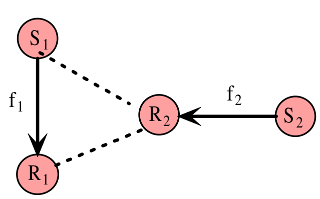


Figure 7: Asymmetric Topology: Transmission Range is Equal to Carrier Sense Range

We classify the asymmetric topologies into two groups according to the difference in the source nodes' knowledge of the channel. The two classes are topologies with equal transmission and carrier sense range and topologies with different transmission and carrier sense range.

5.1 Topologies with Equal Transmission and Carrier Sense Ranges

In previous sections, we studied topologies in which the MAC protocol is able to achieve long-term fairness when all flows are continuously backlogged. However, in asymmetric topologies, due to asymmetry in the channel state, the MAC protocol experiences severe unfairness for continuously backlogged flows. An example asymmetric topology with two flows is shown in Figure 7.

In this example, the sender S_2 is out of range of the sender and receiver of flow f_1, whereas the sender and receiver of flow f_1 are in the range of R_2. Thus, the two flows have different (asymmetric) views of the channel: flow f_1 is aware of flow f_2 whereas the opposite is not true. Consequently, if both flows are backlogged, flow f_1 will have considerably higher throughput than flow f_2 because the sender S_1 is able to sense CTS and ACK sent by R_2, thus knows exactly when to contend for the channel, whereas S_2 cannot sense neither S_1 nor R_1, hence contends randomly. In this case, the system performs close to a distributed strict priority queue, in which flow f_1 has strict priority over flow f_2. Below, we study the throughput and fairness properties of this asymmetric topology in the presence of congestion control without collaboration.

With congestion control and a perfect contention-free MAC, the allocated rates x_1 and x_2 are the approximation to the following system of equations

$$\frac{w_1}{x_1} = \frac{(\frac{x_1}{C_1} + \frac{x_2}{C_2} - \gamma + \epsilon)^+}{C_1 \epsilon^2} \tag{18}$$

$$\frac{w_2}{x_2} = \frac{(\frac{x_2}{C_2} - \gamma + \epsilon)^+}{C_2 \epsilon^2}. \tag{19}$$

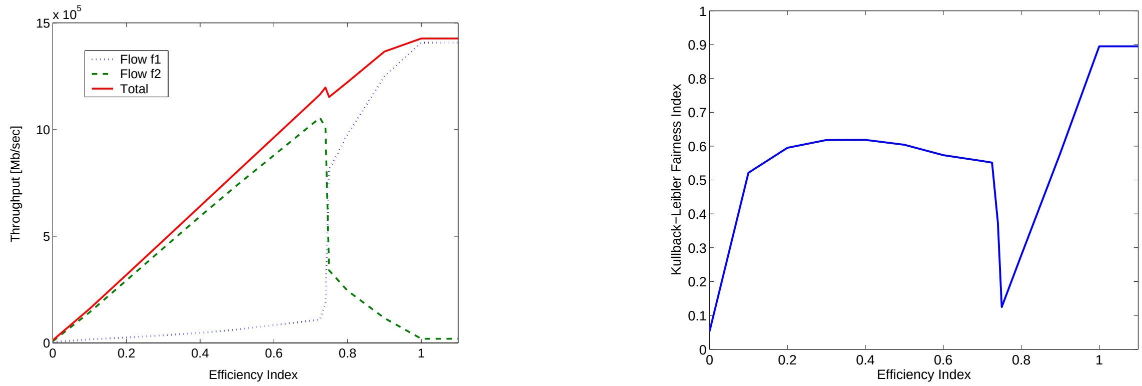
This is due to the fact that flow f_1 is measuring the correct busy time, while flow f_2 is unaware of flow f_1 and is able to locally measure only its own busy time. When $w_1 = w_2 = 1$, $C_1 = C_2 = C$, and $\epsilon \to 0$ we have

$$x_1 = 0 \text{ and } x_2 = \gamma C.$$

Thus, congestion control in asymmetric topologies leads to unfairness, and according to the above analysis, there is no optimal γ. Next, we use simulations to study the throughput and fairness properties of this asymmetric topology.

Throughput. Figure 8(a) depicts the throughput of the two flows as well as the aggregate throughput as γ varies from 0 to 1. For $\gamma = 1$, the system defaults back to the undesirable performance of 802.11 without congestion control.

First, the simulation results match the above theoretical results



(a) Throughput vs. Efficiency Index (b) Long-Term Fairness Index

Figure 8: Asymmetric Topology with Equal Transmission Range and Carrier Sense Range: Simulation Results

only for $\gamma < 0.7$. This discrepancy is due to assumption that $\epsilon \to 0$, whereas in simulations we use $\epsilon = 0.05$. Further decreasing ϵ shifts this transition point towards $\gamma = 1$. Due to the MAC unfairness described above, for any allocated rate $x_1 > 0$ the throughput of flow f_1 is close to the allocated rate, while flow f_2 is able to utilize only the leftover capacity. Next, there is a value of γ, $0.7 < \gamma < 0.8$, such that the rates of the two flows are equal. We name this value as the optimal γ_{opt} for the asymmetric topology with equal transmission and sense ranges. Finally, the aggregate throughput curve has a dip for $\gamma = \gamma_{opt}$. The reason is that when the flows are transmitting at similar rates, some air time is being wasted as flow f_1 is rate limited and flow f_2 is in backoff as it is contending randomly.

Fairness Properties. The fairness properties of asymmetric topologies are illustrated in Figure 8(b), which depicts the long-term fairness index vs. the Efficiency Index γ. We observe that the fairness index varies drastically with γ. There are only two narrow regions in which the fairness index is smaller than 0.5. The first such region is $\gamma < 0.1$: unfortunately, the throughput is low and the performance is unacceptable. The second region is $0.7 < \gamma < 0.8$, near γ_{opt}, with throughput between 1.1 Mb/sec and 1.25 Mb/sec. While a seemingly desirable operating point, unfortunately, we found that the exact value of γ_{opt} is quite sensitive to many parameters such as channel capacity, packet size, etc.

In general, to achieve fairness flow f_1 has to be aware of the demand of flow f_2. Also, there exists an optimal γ_{opt} for the case in which two flows are having different views of the channel. However, the system robustness as a function of γ is low.

5.2 Topologies with Different Transmission and Carrier Sense Ranges

In this section, we study the impact of a small but constant difference in the measured channel busy time on the performance of the congestion control algorithm without collaboration and in asymmetric topologies. We first discuss an example that points out the importance of relying on the Network Allocation Vector (NAV) in random access networks. Then, we describe how the use of NAV results in a small but constant difference in measured busy time, and finally how this impacts the performance of congestion control in asymmetric topologies.

5.2.1 Importance of Network Allocation Vector (NAV)

Consider the topology in Figure 9, in which R_2 is in the carrier sense range of both S_1 and R_1, whereas S_2 is out of the range of both. Thus, when R_2 is transmitting, S_1 and R_1 receive the packet but cannot decode it. Consequently, as per IEEE 802.11 [13] both S_1 and R_1 set their NAVs to the value of EIFS as depicted in Figure 10. IEEE 802.11 also specifies the duration of EIFS frame to be longer than time needed for an ACK frame to be transmitted at the physical layer's lowest mandatory rate. Not relying on the NAV

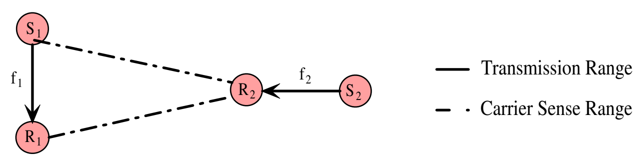


Figure 9: Motivating Example

in such topologies would result in collision, i.e., if S_1 does not set its NAV and starts transmitting there would be a collision at R_2. Consequently, when inferring the channel state locally, we have to assume that the channel is busy whenever the NAV is nonzero. Finally, observe that in the topology in Figure 9, neither S_1 nor R_1 are able to decode the packet from R_2, thus none of them have the exact information about the channel busy time. Next we discuss how this effects the performance of congestion control.

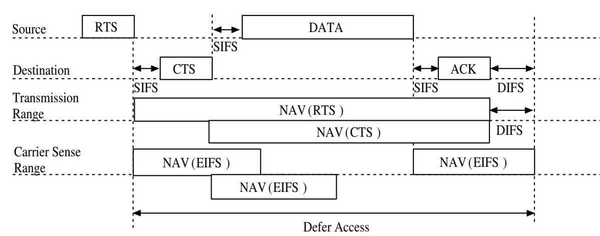
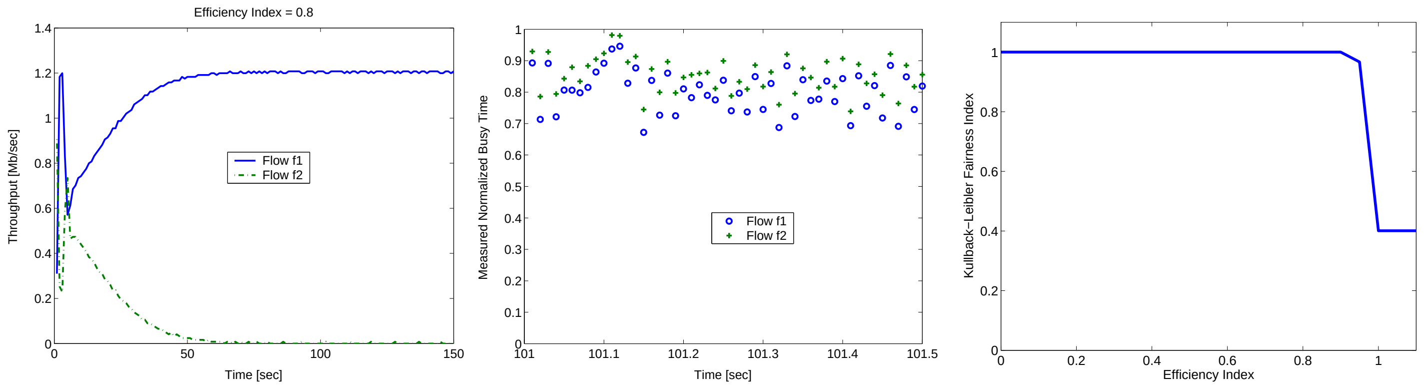


Figure 10: IEEE 802.11

5.2.2 Performance

Here, we study congestion control in topologies in which source nodes measure busy times with a small, constant difference. For that purpose, we use the same example as in Section 5.1, and we set the carrier sense range to be twice the transmission range. Therefore, S_2 is now in carrier sense range of both S_1 and R_1 (and vice versa) as shown in Figure 11. Observe that flows f_1 and f_2 cannot transmit simultaneously. The throughput and fairness properties for this asymmetric topology are discussed below.



(a) Throughput vs. Time (b) Measured Busy Time (c) Long-Term Fairness Index vs. Efficiency Index

Figure 13: Asymmetric Topology with Carrier Sense Range Twice Transmission Range: Simulation Results

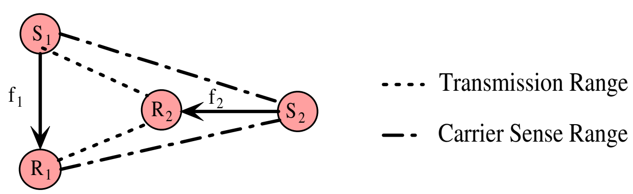


Figure 11: Asymmetric Topology: Carrier Sense Range is Twice Transmission Range

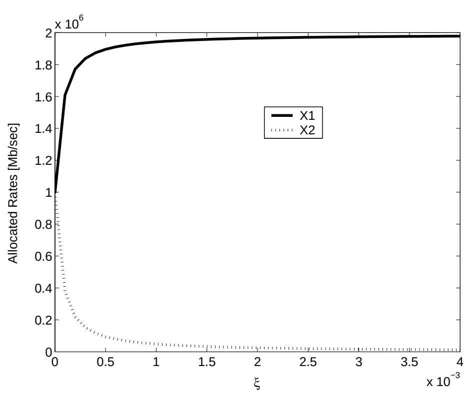


Figure 12: Allocated Rates of Two Flows vs. ξ

For the given topology, the busy time flow f_2 measures is always larger than the busy time flow f_1 measures. The reason is that when either S_1 or R_1 are transmitting, S_2 is able to sense the channel busy, but is not able to decode the packet. Thus, it sets its NAV to the value of EIFS as described above and as shown in Figure 10. Since the duration of EIFS is determined by the time needed for an ACK packet to be transmitted at the physical layer's *lowest* mandatory rate, S_2 overestimates the busy time by some value $\xi > 0$. Thus, the rates the congestion control algorithm will converge to are the solution to the following system of equations:

$$\frac{w_1}{x_1} = \frac{(\frac{x_1}{C_1} + \frac{x_2}{C_2} - \gamma + \epsilon)^+}{C_1 \epsilon^2} \quad (20)$$

$$\frac{w_2}{x_2} = \frac{(\frac{x_1}{C_1} + \frac{x_2}{C_2} + \xi - \gamma + \epsilon)^+}{C_2 \epsilon^2} \quad (21)$$

Figure 12 plots x_1 and x_2 as a function of ξ. The other parameters from the above equations are as presented in Section 3.4.2. We observe that for ξ as small as 0.0001 the rate x_2 converges to a value close to 0.

In other words, as long as there is a very small but constant difference between the measured busy times of the two flows, the rates will converge to incorrect values. Thus, the above analysis suggests that there is no optimal γ for the asymmetric topologies with small, constant difference in measured busy times. Next, we study the throughput and fairness properties via simulations.

Throughput. Figure 13(a) depicts the throughput progression in time of both flows. The results shown are for $\gamma = 0.8$; however, we note that similar performance is obtained for any $\gamma < 0.95$ (i.e., flow f_2 always converges to zero throughput while the convergence value of flow f_1 varies with γ).

The main reason as described above is the constant discrepancy in measured busy times. Figure 13(b) shows a sample of normalized measured busy times of the two flows between 101 sec and 101.5 sec. of simulation time. Observe that flow f_2 always measures slightly larger busy time than flow f_1, which based on the analysis above explains the results.

Fairness Properties. Figure 13(c) depicts the long term fairness index as a function of the Efficiency Index γ for the asymmetric two-flow topology in Figure 11. We observe that the fairness index is 1 for $\gamma < 0.95$ and it decreases to 0.4 for $\gamma = 1$ (i.e., the system performs as pure IEEE 802.11 without congestion control). This is due to the aforementioned convergence to incorrect rates. Hence, for the case of local inference of the channel state, congestion control in asymmetric topologies with carrier sense range twice the transmission range degrades the performance. Therefore, in the next section we study congestion control in the presence of collaboration among nodes.

6. CONGESTION CONTROL WITH COLLABORATIVE INFERENCE OF CHANNEL STATE

The previous sections clearly indicate that relying on local inference of the channel state can result in serious unfairness, thus pointing to the need for collaboration among nodes. Thus, in this section we study congestion control under inter-node collaboration, i.e., inter-node sharing of channel-state information within a contention neighborhood.

We first describe how we realize the collaboration and the type of metrics we are measuring. Then we study fairness and throughput properties of congestion control in the presence of collaboration in more complex topologies.

6.1 Collaboration

The objective of the algorithm that enables collaboration is that each sender collects information about the channel busy time for all contention neighborhoods it belongs to. The design space for such a distributed algorithm is immense, and it includes measurement of the required metric (e.g., busy time, offered load, carried load, capacity), message distribution, and rate computation.

Our focus in this paper is not the design of the algorithm itself. Rather, we study the effects of collaboration and the nature of the metric measured on the throughput and fairness properties of the congestion control algorithm. Therefore, we use a data structure to obtain the needed information about the channel state at each sender. We consider that the topology is known and that the data structure is populated periodically every 20 msec, which is also the period of rate calculation. When a source rate needs to be recalculated, the data structure is accessed, and the needed information is fetched. We then use rate limiters to enforce the calculated source rates. In other words, we consider *perfect* collaboration among nodes in a contention neighborhood to study the performance limits of collaboration, and omit factors such as overhead and logistics of information exchange.

As described in Section 2, rate update is done periodically, and one of the metrics required for the rate update is the channel busy time. However, there are multiple ways one can measure the channel busy time. Here, we consider three approaches. In the first approach, each node measures the channel busy time locally using carrier sensing. Thus, in this way each node measures the *aggregate* busy time of the channel, and populates the data structure by this aggregate value. In the second approach, the channel busy time is calculated for each source as the ratio of throughput to capacity. Thus, here each sender measures the number of packets it transmits. We do not measure *capacity* and use the capacity specified in 3.4.2 instead. The data structure is populated then with the ratio *throughput/capacity* for each sender. Finally, in the third approach, each sender measures its offered load, and the data structure is populated with *offered_load/capacity*. This approach best resembles what is done in Section 2.

6.2 Flow-in-the-Middle Topology

Here, we study the flow-in-the-middle topology presented in Figure 14. In this topology flows f_1 and f_3 are out of range of each other and can transmit simultaneously. However, flow f_2 is in the range of both f_1 and f_3 and can transmit only if none of these two are transmitting.

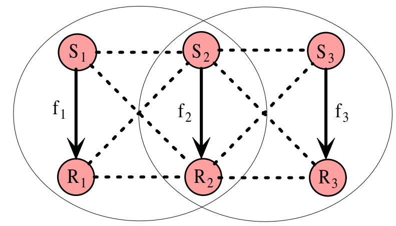


Figure 14: Flow-in-the-Middle Scenario

Continuously Backlogged Flows. If all flows are backlogged, flow f_2 will have considerably lower throughput than the other two flows, whose throughput will be close to the maximum. The reason is that flow f_2 is sensing the channel busy whenever either of the two outer flows are transmitting. Since the transmissions of the outer flows are not synchronized, the busy time flow f_2 is sensing can be quite long, thus the transmission opportunities for flow f_2 are severely limited.

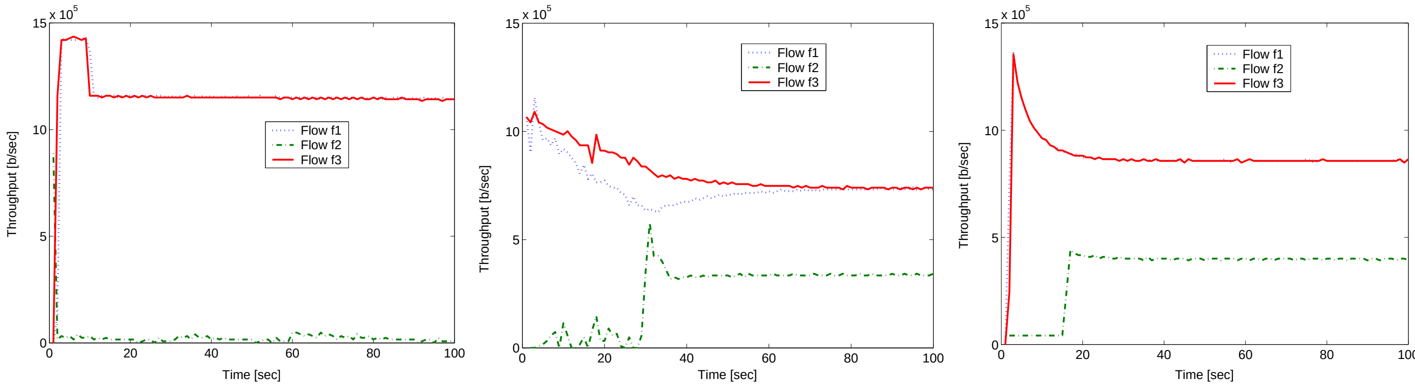
Congestion Control with Collaboration. We use this topology to study congestion control with inter-node collaboration and the choice of the measured metric. Note that in this topology the proportional fair shares for the three flows are 2/3, 1/3 and 2/3 of the available capacity. Flow f_2 has half the fair share since it belongs to two contention neighborhoods, whereas the outer two flows each belongs to one contention neighborhood.

Figure 15 depicts throughputs for the three flows for the cases in which the congestion control algorithm calculates rates based on aggregate busy time, throughput, and offered load (Figures 15(a), (b), and (c) respectively). All three plots represent a single simulation run obtained with the Efficiency Index $\gamma = 0.8$, $\gamma = 0.6$, and $\gamma = 0.6$, respectively.

We choose γ's such that the maximum achievable aggregate throughput per contention neighborhood is the same, thus the performance for the three approaches should be similar. We make the following observations.

- Different values for γ are required when different measures of busy time are used. The reason is that by measuring aggregate busy time, we essentially measure the portion of time that is being used for any transmission (i.e., data packets, control packets). However, by measuring throughput for example (or measuring offered load), we are measuring only transmissions of data packets. Hence, the absolute values of busy times measured in these three different ways are quite different, which is the reason the γ's are different.

- From Figure 15(a) we see that there are instances in which the congestion control algorithm does not converge to the correct rates. This is due to the nature of the measured metric in this case. As aforementioned, the congestion control algorithm in this case calculates the rates utilizing the locally measured aggregate busy time. The actual assigned rates in this simulation run for flows f_1, f_2 and f_3 are 1.15 Mb/sec, 0.63 Mb/sec and 1.15 Mb/sec, respectively. However, due to the unfair MAC, almost none of the packets from flow f_2 are being transmitted, while both of the outer flows achieve throughput exactly as assigned rates. In terms of aggregate busy times, this allocation and the correct one (800 kb/sec, 400 kb/sec and 800 kb/sec) are the same. Thus, in some instances the system gets to this "incorrect" stable state and remains there. We do note that most of the simulation runs generate performance similar to the one presented in Figure



(a) Measuring Aggregate Busy Time (b) Measuring Per-Flow Throughput (c) Measuring Input Rates

Figure 15: Per-Flow Throughput when Different Congestion Metrics are Measured

15(b).

- The convergence time in the case in which throughput is measured is double the convergence time for the case in which offered load is measured. The reason is that with the throughput measurement there is some error introduced by the random nature of the system, resulting in more rate oscillation and longer convergence time.

In general, in topologies with multiple contention neighborhoods, each sender needs to obtain the busy times for all contention neighborhoods it belong to in order for the congestion control algorithm to achieve convergence to the correct rates. Moreover, to ensure convergence to correct rates, these obtained values for busy times need to be per-flow rather than aggregate.

6.3 Topology with Multiple Contention Neighborhoods

In this final topology, our goal is to compare the performance and fairness of TCP and the utility-maximization congestion control algorithm with collaboration. For the comparison we choose the topology presented in Figure 16 because it incorporates multiple issues and sub-topologies studied throughout the paper. In other words, the topology is a combination of symmetric, asymmetric, and flow-in-the-middle topologies.

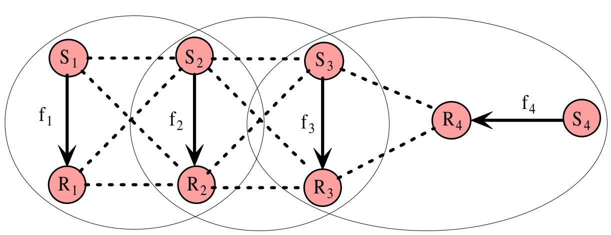


Figure 16: Topology with Multiple Contention Neighborhoods

Figure 16 depicts the throughputs for each flow. In the TCP experiment below, each source generates long-lived TCP-Sack traffic, with all parameters set to their default values. In the utility-maximization congestion control algorithm experiments, collaboration is based on the measured throughput, and the efficiency index is set to $\gamma = 0.6$.

Results. Observe that TCP is not able to achieve the full available bandwidth. This is due to the congestion control of TCP: when

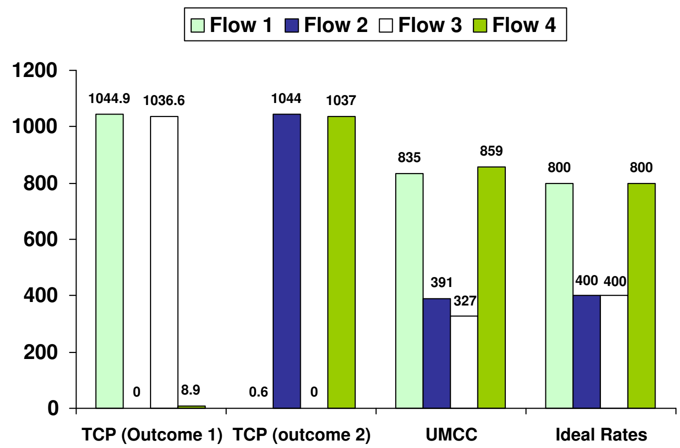


Figure 17: TCP Performance

a TCP flow experiences a loss, it reduces its window by a factor of 2 and is not able to exploit the full available bandwidth. Next, in all experiments we observed only two outcomes. Outcome 1 in which flows 1 and 3 are active while flows 2 and 4 are almost starved, and outcome 2 in which flows 2 and 4 are active while flows 1 and 3 are almost starved. We performed 50 simulation runs of which two thirds of the runs yield outcome 1, and one third yield outcome 2. Observe that although flows 1 and 4 can be active simultaneously, this outcome never occurs. This is due to the information asymmetry embedded in the topology. When flow 1 is transmitting, flow 2 cannot since they are in the same contention neighborhood. At the same time, flows 3 and 4 are both out of the range of flow 1. However, flows 3 and 4 present the same example described in Section 5.1, so that flow 3 obtains almost maximum throughput. We do point out that TCP's inability to fully utilize resources, and its unfairness in wireless networks have been widely studied and addressed in the literature, and solutions have been proposed [5, 4, 18, 19]. We do not consider any of these proposals, and simply study TCP as a baseline for comparison.

The ideal fair shares for the Utility Maximization Congestion Control (UMCC) algorithm with $\gamma = 0.6$ are 800 kb/sec, 400 kb/sec, 400 kb/sec and 800 kb/sec for flows 1, 2 ,3 and 4 respectively. Observe that the achieved shares of the utility-maximization congestion control algorithm with inter-node collaboration are very close to the ideal ones. Moreover, the congestion control algorithm achieves throughput up to 17% higher then TCP, most importantly, without starving any flows.

7. RELATED WORK

Congestion control for wireline networks is a topic of intense research efforts, including study of the utility-based optimization framework for Internet congestion control, e.g., [11, 20, 21].

This same framework has been applied to wireless ad hoc networks, e.g., [7, 8, 6, 22, 23]. In particular in [7, 8], flow contention among link-layer flows is used to propose utility based optimization to achieve MAC layer fairness. In [23] rate allocation is formulated as a utility maximization problem with time constraints, and end-to-end and hop-by-hop schemes that provide convergence to the allocated rates are proposed. In [22], this framework is used to study cross layer design, and a jointly optimal congestion control and power control algorithm is developed. Cross layer design is also studied in [6], which considers joint design of optimal congestion control and MAC protocols.

The impact of MAC unfairness and channel inconsistency has been studied in the context of TCP congestion control. In [24], simulation and testbed measurements are used to study the performance of TCP over the wired-to-wireless and wireless-to-wired networks. TCP's unfairness over multihop wireless networks is studied in [5, 25]; in [5], neighborhood RED is proposed to improve TCP fairness in multihop wireless networks. In [26], throughput analysis of multihop chain networks is presented. Finally, in [4] a fractional factorial experimental design is employed to identify the performance factors that lead to unfairness and starvation of TCP over multihop wireless networks.

This paper differs from such studies in that we study the impact of CSMA/CA based MAC protocols with consistent and inconsistent channel state on the performance of the utility-maximization congestion control schemes. In particular, we explore the impact of the wireless channel, unknown data transmission capacity, unknown service order and unknown and incorrect system state on the performance of congestion control algorithm in environments with and without collaboration. We also demonstrate the requirements of a wireless network in order for congestion control algorithms to provide high performance.

8. CONCLUSIONS

In this paper, we studied the utility maximization approach to congestion control in wireless CSMA-based networks and studied the fundamental challenges arising in multihop networks. We explored the impact of channel state consistency, service order, internode collaboration and unknown data transmission capacity on the performance of the congestion control algorithm. Our work provides a deeper understanding of the performance of utility maximization congestion control over CSMA-based networks and yields new insights that can guide the analysis and design of a distributed congestion control algorithm for CSMA-based networks.

9. REFERENCES

[1] V. Bharghavan, S. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LANs," in *Proceedings of ACM SIGCOMM '94*, London, UK, 1994.

[2] M. Garetto, T. Salonidis, and E. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," Barcelona, Spain, 2006.

[3] M. Garetto, J. Shi, and E. Knightly, "Modeling media access in embedded two-flow topologies of multi-hop wireless networks," in *Proceedings of ACM MobiCom '05*, Cologne, Germany, Sept. 2005.

[4] V. Gambiroza, B. Sadeghi, and E. Knightly, "End-to-end performance and fairness in multihop wireless backhaul networks," in *Proceedings of ACM MobiCom '04*, Philadephia, PA, September 2004.

[5] K. Xu, M. Gerla, L.Qi, and Y. Shu, "Enhancing TCP fairness in ad hoc wireless networks using neighborhood RED," in *Proceedings of ACM MobiCom '03*, San Diego, CA, Sept. 2003.

[6] L. Chen, S. Low, and J. Doyle., "Joint congestion control and media access control design for ad hoc wireless networks," in *Proceedings of IEEE Infocom'05*, Miami, FL, Mar. 2005.

[7] Z. Fang and B. Bensaou, "Fair bandwidth sharing algorithms based on game theory frameworks for wireless ad-hoc networks," in *Proceedings of IEEE Infocom*, Hong Kong, China, Mar. 2004.

[8] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, "Achieving MAC layer fairness in wireless packet networks," in *Proc. of ACM MobiCom'00*, Boston, MA, Aug. 2000.

[9] R. Diestel, *Graph Theory*. Springer-Verlag, 1997.

[10] A. Bar-noy, A. Mayer, B. Schieber, and M. Sudan, "Guaranteeing fair service to persistent dependent tasks," *SIAM Journal on Computing*, vol. 27, no. 4, pp. 1168–1189, Aug. 1998.

[11] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: Shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.

[12] J. T. Wen and M. Arcak., "A unifying passivity framework for network flow control," in *Proceedings of IEEE Infocom'03*, San Francisco, CA, Apr. 2003.

[13] IEEE, "Wireless LAN media access control (MAC) and physical layer (PHY) specifications," June 1999, IEEE standard 802.11.

[14] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly, "Ordered packet scheduling in wireless ad hoc networks: Mechanisms and performance analysis," in *Proceedings of ACM MobiHoc*, Lousanne, Switzerland, June 2002.

[15] C. E. Koksal, H. Kassab, and H. Balakrishnan, "An analysis of short-term fairness in wireless media access protocols," in *Proceedings of ACM Sigmetrics 2000*, 2000.

[16] T. Cover and J. Thomas, *Elements of Information Theory*. John Wiley and Sons, 1991.

[17] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination f unction," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[18] S. Floyd, "Highspeed TCP for large congestion windows," 2003, internet RFC 3649.

[19] K. Tang and M. Gerla, "Fair sharing of MAC under TCP in wireless ad hoc networks," in *Proceedings of IEEE MMT*, Oct. 1999.

[20] S. Kunniyur and R. Srikant, "End-to-end congestion control schemes: Utility function, random losses and ECN marks," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 689–702, Oct. 2003.

[21] S. Low and D. Lapsley, "Optimal flow control, I: Basic

algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, Oct. 1999.

[22] M. Chiang, "To layer or not to layer: Balancing transport and physical layers in wireless multihop networks," in *Proceedings of IEEE Infocom '04*, Hong Kong, China, Mar. 2004.

[23] Y. Yi and S. Shakkottai, "Hop-by-hop congestion control over a wireless multi-hop network," in *Proceedings of IEEE Infocom'04*, Hong Kong, Mar. 2004.

[24] K. Xu, S. Bae, S. Lee, and M. Gerla, "TCP behavior across multihop wireless networks and the wired internet," in *Proceedings of ACM International Workshop on Wireless Mobile Multimedia*, Atlanta, GA, 2002.

[25] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC protocol work well in multihop wireless networks," *IEEE Communications Magazine*, vol. 39, no. 6, June 2001.

[26] A. Kherani and R. Shorey, "Throughput analysis of TCP in multi-hop wireless networks with IEEE 802.11 MAC," in *Proceedings of IEEE Wireless Communications and Networking Conference*, Atlanta, GA, Mar. 2004.