2012-04-13

Performance Evaluation of Optimal Rate Allocation Models for Wireless Networks

Ryan Michael Padilla
Brigham Young University - Provo

Performance Evaluation of Optimal Rate Allocation Models for Wireless Networks

Ryan M. Padilla

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Daniel Zappala, Chair
Sean Warnick
Dan Olsen

Department of Computer Science

Brigham Young University

August 2012

ABSTRACT

Performance Evaluation of Optimal Rate Allocation Models for
Wireless Networks

Ryan M. Padilla
Department of Computer Science, BYU
Master of Science

Convex programming is used in wireless networks to optimize the sending or receiving rates of links or flows in a network. This kind of optimization problem is formulated into a rate allocation problem, where each node in the network will distributively solve the convex problem and all links or flows will converge to their optimal rate. The objective function and constraints of these problems are represented in a simplified model of contention, interference, and sending or receiving rates. The Partial Interference model is an optimal rate allocation model for use in wireless mesh networks that has been shown to be theoretically superior to other conceptual models. This paper compares the Partial Interference model to three other models of wireless networks using the ns-3 simulator to verify these claims. It discusses where the model works as expected, where the model fails to improve network utility, and the limitations inherent to its use.

Keywords: ns-3, optimization, simulation

ACKNOWLEDGMENTS

Thanks to Dr. Zappala for the constant support and constructive criticism that helped my research progress and meet expectations.

Thanks to Dr. Warnick for the much needed help to have a basic understanding of optimization theory and the background to be able to implement these models in the simulator.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless mesh networks provide a cost effective solution to creating a network. They lack a permanent infrastructure allowing them to be quickly deployed at a relatively low expense compared to their wired counterpart. This is particularly attractive to urban or rural areas where the permanent infrastructure has yet to be created. Mesh networks are also used for smart meters and home automation. Unfortunately, mesh networks have at least one major drawback; they don't provide consistent performance due to 802.11 MAC fairness issues in ad hoc mode and low bandwidth for competing multi-hop TCP flows.

When using a wireless medium, nodes in a network must share the use of that medium. If two nodes try to send at the same time, it is likely that the packets sent will be corrupted at the receivers. As these nodes move away from each other, the likelihood of packet corruption decreases. Contention occurs when two sending nodes are considered to be too close to each other to use the medium at the same time. They should share the medium according to some notion of fairness so both nodes are not sending at the same time. Interference occurs when two sending nodes are outside contention range of each other so they don't share the medium, but still cause packet corruption at a receiver.

The 802.11 MAC Protocol defines how and when nodes in a wireless network can access the wireless medium. It forces nodes to share the medium if they are within carrier sensing range of each other. A node is considered to be within carrier sensing range of another node if the signal strength of transmissions at the receiving node is above some threshold. The 802.11 MAC allows random access to the wireless medium and uses a back-off algorithm to determine how transmitting nodes should recover from collisions and packet loss at the MAC layer. Unfortunately, the 802.11 MAC performs poorly when the combined sending rate of all flows exceeds the network capacity. The 802.11 MAC Protocol does not provide a consistent notion of fairness when operating in ad hoc mode, which is a common operating mode of mesh networks [7]. Access to the wireless medium is not consistently distributed among competing nodes, leading to unfair bandwidth distribution.

TCP can also cause unfair bandwidth allocations among competing flows. TCP operates under the assumption that lost packets are due to congestion in the network, which may not always be true in a wireless mesh network. This can lead to throughput levels that are well below the optimal value TCP should converge to. Shi et al. has shown that two hop TCP flows operating in a mesh network can be starved when competing with other one hop flows [12]. This is due to the combined effects of the 802.11 MAC and TCP congestion control algorithms trying to recover from collisions and loss events. Competition between multi-hop and single-hop flows invariably occurs in a mesh network, leading to severe unfairness between two or more TCP flows.

One way to resolve these problems is to calculate a fair rate for each link in the network and restrict the link to this rate. A common approach is to formulate an optimization problem to maximize network utility, subject to constraints that model contention and interference among network links [2, 4, 6, 7, 14]. Similar methods can be used to optimize flow rates. These models have the potential to provide optimal performance and fairness for a wireless network, but often have not been evaluated with accurate simulations or experiments on a real network.

In this thesis, we evaluate two common models that have been used to characterize the effects of contention and interference. The first model is a Binary Contention model, which defines contention so that for any pair of nodes, if one contends with the other, both are considered to contend with each other. This model, when formed into an optimization problem, optimizes the sending rate of each link so that links which contend with each other must share the available bandwidth according to some notion of fairness. The Binary Contention (BC) model has three different variants, based on how interference is treated. The first variant, called the Interference Ignored (II) model, ignores the effects of interference. The second variant is called the Interference as Contention (IC) model. It has the same requirements and construction as the II model, but changes the definition of contention. Contention between nodes is now defined so that for any pair of nodes, if one of the nodes is within contention or interference range of the other, both are treated as though they contend. Under this model, nodes will be sharing the available bandwidth with a much larger set of nodes. The third variant is known as the Adaptive Contention (AC) model. This model is a hybrid of the II and IC models. It computes the expected utility of both the II and IC models and uses the allocations of the model that is supposed to have the better utility.

The second model, called the Partial Interference (PI), explicitly models interference in addition to contention. This model optimizes the receiving rate of all nodes in the network, where the receiving rate is a function of both the sending rate and the amount of interference in the network. The PI model has been shown by Wang et al. to always perform as well or better than the three variants of the BC model [14]. However, these results were derived using numeric models in Matlab. Later work by Wang in an experimental setting showed that the relationship between interference and the sending rate of the interferer is not linear. Because the PI model is based on this assumption, the predicted receive rates were not always correct [13].

In this thesis we generalize Wang's experimental work by evaluating the BC and PI models using the ns-3 packet level simulator. We show that the PI model does not always outperform the BC variants as expected. These discrepancies arise due to several modeling assumptions that are not valid. First, the PI model is based on the assumption that there is a linear relationship between interference and the interferer sending rate. That is, as the interferer decreases its rate, the interferee should have a linear increase in receive rate. We show this does not always occur in the simulator. Second, the PI model is based on the assumption that interferers are independent. In other words, if two interferers cause 20% packet loss when sending individually, then when both are sending they should cause 40% packet loss. The simulator shows that interference drastically increases with multiple interferers causing the PI model to be inaccurate. The PI model was constructed based on the results of Niculescu, who showed these assumptions were valid in his mesh testbed [8].

Despite these flawed assumptions, our results show that the PI model still provides some benefits relative to BC. The PI model prevents links that experience high levels of packet loss due to interference from being starved by greatly reducing the rate of the interfering node. Our results also demonstrate that high network load can cause control traffic to be starved. When this occurs, links are not able to coordinate rates and rate allocations will not be made or changed. It is possible that network capacity decreases as contention increases, which may account for this problem.

It is imperative that models used in optimization protocols are thoroughly designed, proven, and tested in a cohesive and transparent process so there will be a high level of confidence in the optimal solutions that are found. Thoroughly testing a new model in a mesh network testbed is difficult. The simulations done in this work illustrate the challenges associated with deploying a distributed rate allocation protocol. Questions regarding contention and network capacity, the interference-to-rate relationship, and the effects on packet loss of multiple interferers must be further validated in a real network.

Chapter 2

Related Work

Nandagopal et al. present one of the seminal papers on optimization techniques applied to network fairness and resource allocation [7]. They show how continuous, concave, and differentiable utility functions form a convex problem that is a representation of some notion of fairness for rate allocations between nodes in a network. They develop a framework that shows how a network graph can be constructed and then transformed into a resource allocation graph that represents transmission constraints on the utility function.

The representation of resource constraints provides the model of when nodes can communicate with each other. It is constructed to account for contention and to ignore interference. The graph is constructed as follows: initially, a graph of the topology is created where vertices represent nodes in the network and edges represent contention between those nodes, where contention is defined as being within carrier sensing range of another node (see Figure 2.1). Interference is ignored. This graph is then formed into a link contention graph where vertices represent one-way links (sender and receiver) and edges represent contention between those links (see Figure 2.2). Maximal cliques are then calculated based on the contention graph. Each maximal clique in the graph represents a collection of links that contend with one another. In Figure 2.2 there are three cliques: AB, BC, CD; BC, CD, DE;

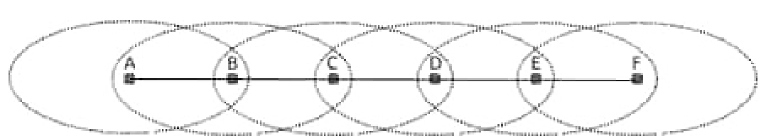


Figure 2.1: No two neighbors can simultaneously transmit

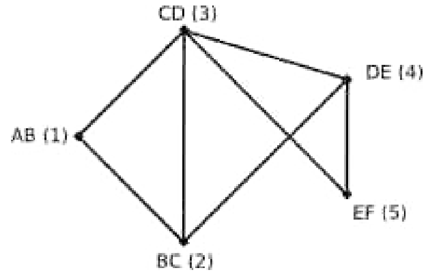


Figure 2.2: In each clique, only one node can transmit

and CD, DE, EF. Because of this contention, only one of the links (or the node that controls that link) in a clique can actively send at a time.

Chen et al. considers the problem of unfairness when using the congestion control algorithms of 802.11 MAC and TCP [4]. A general framework is laid out to implement future TCP and MAC designs that are based on optimization techniques. They formulate a utility maximization problem separated into its primal and dual problem. Constraints are derived from link contention among nodes, ignoring interference. Using Lagrangian relaxation and duality theory, a distributed algorithm is implemented. This paper differs from Nandagopal et al. in that a simulation in the ns-2 simulator is used to show how individual nodes in the network converge to the optimal rate.

The two previous papers discuss the general optimization problem,

$$\begin{aligned} \mathbf{Q} : \max f(s) = \sum_{l \in L} U(s_l) \\ \sum_{l \in L(j)} s_l \leq \epsilon_j, \ \forall j \in C \\ s_l \geq 0, \ \forall l \in L \end{aligned} \tag{2.1}$$

where U is some utility function and s_l is the sending rate of link l. It is constrained so each for each clique, j, the sum of link sending rates for each link in j is less than the clique capacity of j. The sending rate of each link l is also constrained to be greater than or equal to zero.

Bensaou and Fang uses the same optimization approach as Chen et al. to show how a fair MAC protocol could be implemented in practice [2]. The problem is formulated as a general utility maximization problem as shown in Equation 2.1, and using Lagrangian relaxation and duality theory, a distributed algorithm, known as the cooperative gradient-based algorithm (CGA) is created. The constraints on the problem differ from previous authors in how cliques (or contention) are defined. Two nodes are said to contend if they are within two hops of each other. This model loosely takes interference into account by assuming that any node within two hops of a sending node must interfere or contend with its packets. The sending node should, therefore, share air time with all nodes within two hops. Interference is loosely approximated and treated as contention.

This algorithm allows links (and by extension the nodes that make up those links) to individually calculate their optimal rate based on a local price. Rates are then sent to all other contending nodes and the local price of each of the contending nodes is updated. A separate link discovery protocol is also defined and implemented so each node can formulate an accurate network topology and calculate maximal cliques. CGA and the link discovery protocol were implemented in a real ad hoc network and tested to show rate convergence.

Each of the previous authors establish the use of optimization protocols as a practical and desirable solution to the unfairness issures that arise in a mesh network. However, the proposed solutions do not explicitly address the harmful effects of interference in a mesh network and how unfairness issues in this context can be overcome.

Wang et al. introduces the PI model to better model interference and its effects on signal propagation between transmitting nodes [14]. Contention is described as two nodes within carrier sensing range of each other, while interference is describe as two non-contending nodes that transmit simultaneously and cause packets to become corrupted. A new optimization problem,

$$\mathbf{P} : \max_{s} f(r) = \sum_{l \in L} U(r_l)$$

$$\sum_{l \in L(j)} s_l \leq \epsilon_j, \ \forall j \in C$$

$$s_l \geq 0, \ \forall l \in L$$

$$r_l = d_l s_l \prod_{i \in I(l)} (1 - a_{il} s_i), \ \forall l \in L$$

is formulated taking the effects of interference into account. The problem now optimizes the receive rate, rather than the sending rate, using the model described by Niculescu [8] to predict receive rates. It is shown how this new problem can be formulated into a convex, polynomial time, distributed algorithm that can be implemented. The PI model then is compared against the previously described II, IC, and AC models and shown to always perform as well or better than these other models in Matlab simulations.

It is important to note that all the models previously described optimize link rates between nodes, rather than flows between applications. As a result, the optimal rates that are calculated may not be realizable in practice. This limitation must be accepted because comparisons are being made against the PI model, which has only been formulated as a link-based solution. The PI model is formulated as an optimization of links because it is not known how to formulate the flow based optimization version of this model as a convex problem. Current solutions are, therefore, based on links rates, recognizing this limitation.

Ripplinger has developed the First Principles Model [11]. It calculates the effects of interference and contention using Uniform Random Sets to characterize when and how often nodes are able to carrier sense and interfere with other nodes. This new model, given certain limiting conditions, is shown to reduce to each model previously described. This model is formulated into an optimization problem that is, in many instances, non-convex. Because the problem is intractable, it is intended for off line use to calculate an upper bound on performance. This upper bound can be used as a benchmark to show how much information is captured within the simpler convex solutions previously presented.

Chapter 3

Simulation Setup

Simulations are a valuable technique in testing protocols. They allow scenarios that cannot practically be implemented in the real world to be easily tested in a simulated environment. They are also significantly faster, cheaper, and provide a very controlled environment. It is not uncommon for wireless mesh testbeds to be running experiments in the presence of some external interference, which is simply due to the complex digitized world we live in. A simulator can remove all external noise and inconsistencies. However, the simulator must be accurately configured to represent the real world and the models the simulator uses must be both accurate and correctly applied. Simulations, in short, can be a powerful technique to validate optimization protocols and their effects in various situations.

When an optimization problem is implemented in a simulator, there are two types of models that must be considered. There is the model that the optimization problem is based on and there are the models the simulator uses to calculate when and whether or not a packet is received based on signal strength and interference. Wireless signal propagation and the corresponding effects of interference is a complex problem. The ns-3 simulator is a packet-level simulator that contains models of interference and signal propagation that have been shown to closely model the results of experiments done in wireless mesh networks [1, 10, 9]. In using the ns-3 simulator to implement an optimization protocol, the focus will be on the models used in the optimization problem and it is assumed the default settings in the models of ns-3 accurately reflect reality in a free space environment with no obstacles.

We use the ns-3 simulator to implement and test several optimal rate allocation models to help validate previous work in this domain. Our design runs in user space on top of the current 802.11 MAC protocol. We implement the II, IC, AC, and PI models and determine under what conditions the highest network utility is achieved.

3.1 Assumptions and Requirements

We perform the same experiments done by Wang et al., but we perform them in the ns-3 simulator using a user space implementation. We focus on comparing network models and their performance in a simulated environment. To this end, we ignore the details of link discovery and assume the link contention graph is universally known before the experiment starts. This means the network topology is static and universally known as well. To ensure that the contention graph is as accurate as possible, we require that each node transmit at a constant bit rate that is the same as the broadcast rate. This ensures the contention graph is accurate for all transmissions. If channel rates were not static, a separate contention graph must be created for each rate because contention and interference change at each channel rate.

These assumptions are reasonable assuming the network is a fixed mesh network and a short session of testing can be run prior to the actual experiments to establish carrier sensing range, interference, and link capacities. In testing, I will assume that the link capacity between node A and node B is the same as between node B and node A to limit the number of tests that must be performed. Because a simulator is a very controlled environment, it is reasonable that this form of testing is accurate.

In implementing the different optimization models, certain limitations have been discovered. The optimization protocols require that if nodes A and B are said to contend, then they must be able to communicate with each other so that sending rate information can be exchanged. This becomes problematic if node A is within carrier sensing range and outside transmission range of node B and no other communication path is available. Because the 802.11 MAC typically has a contention range that extends beyond transmission range, it is possible to have contending nodes that cannot communicate. If we simply describe contention as being transmission range, then allocations can be assigned that will exceed network capacity, leading to potential unfairness issues that were described earlier. We, therefore, limit most network topologies to those where nodes within contention range of each other have a communication path either directly or through some other node.

The optimization protocol was designed with the assumption that any node in the network could start sending at any time. This assumption led to the creation of a contention graph that accounts for every possible link in the network. Cliques are then formulated based on this contention graph. Because any link in the network could become active at any moment, all nodes with links that are in the same clique must coordinate rate information, even if a node is not sending anything. This, in turn, forces an interfering node to communicate with an interferee if it is in contention range, even if the interferee never sends a packet. This assumption has an effect on testing that we will describe later in this paper.

3.2 Rate Controller

We implement a rate controller above the MAC layer and below the IP layer that enforces the rate allocations assigned by each model. In a real network environment, packet interception below the IP layer has been implemented by Buck et al., making our analogous implementation in the simulator a practical realization [3].

The rate controller limits packets based on a timed interval that corresponds with the limit assigned to the associated link. Each node has a rate controller to enforce the rates assigned to the one or more links a node is a part of. When a packet arrives at the rate controller, it is given a timestamp, which is based on the rate of the link, that indicates when the packet should be given to the MAC layer. All packets are stored in a priority queue ordered by time-stamp. Whenever the rate allocation assigned to a link changes, the entire queue must be traversed and all packets apart of that link must have their time-stamps updated to reflect the new rate. In this way rate limits are always applied immediately.

3.3 Optimization Protocols

We implement an optimization protocol based on the designs presented by Bensaou and Fang [2], Low and Lapsley [6], Nandagopal et al. [7], and Wang et al. [14]. The implementation differs in concept from previous implementations in that it is explicitly designed to reside in user space above 802.11. This approach works because 802.11 is potentially unfair only when links are attempting to acquire an unfair bandwidth allocation. The optimization algorithms, derived from their respective models, coupled with a rate controller, will prevent this from occurring.

Dent et al. shows that the carrier sensing range that determines contention can be adjusted, which may help alleviate the problem of contention range being larger than transmission range [5]. In the 802.11 MAC, if signal strength is above a certain threshold, it tries to receive the packet, otherwise it checks a separate threshold to determine if signal strength is high enough for the wireless medium to be considered busy. The default configuration of the 802.11 MAC causes the contention range to be significantly larger than the transmission range. In order for the contention range and the transmission range to be the same, a node must be able to receive all packets it attempts to receive when there are no interfering nodes and all signals below this threshold must be ignored. This setup requires a controlled environment where each node has the same transmission power and the same receive thresholds. If these conditions are not met, it will be possible to have uni-directional links and to have instances where contention range extends beyond transmission range, allowing the optimization protocol to potentially fail.

Wang et al. perform Matlab simulations comparing the Partial Interference (PI) model to the II, IC, and AC models [14]. In practice, we find that the IC and AC models are impossible to test using the topologies described in their paper without some kind of omniscient controller. The topologies cause interferers to not have any communication path to interferees, preventing the optimization protocol from functioning. We, therefore, perform the experiments of Wang et al. using only the PI and II models. We then make a direct comparison with the findings of Wang et al. regarding the PI model. In order to compare the PI model to the IC and AC models, we create similar topologies to those of Wang et al., but include a chain of nodes between the interferers and the interferees so bi-directional communication is possible. Unfortunately this failed to work as expected and will be discussed later.

The PI model, though constructed from the same basic framework described by Nandagopal et al. [7], differs from previous models in that it requires an interference map in addition to a contention graph. We create the interference map based on the methodologies presented by Niculescu [8]. However, we use unicast instead of broadcast to measure interference among links because we want an exact measure of interference rather than an approximation. In each of the experiments we use the PI model to predict the receive rate of the link being interfered with. We then compare this with the actual receive rate to validate how well the PI model predicts packet reception in the face of interference. Therefore, the interference map will be calculated for every experiment performed, regardless of whether the PI model is actually being used to assign rate allocations.

Chapter 4

Experiments

We perform two sets of experiments using the ns-3 simulator to validate model performance. The first set modifies the carrier sensing and receive thresholds so an interferer can get close enough to some receiver to cause 100% packet loss. These could be potentially duplicated in a real network by changing the wireless driver. This set of experiments is designed to validate work done by Wang et al. In the second set of experiment we maintain the default settings of the simulator as much as possible under the assumption that these settings reflect how a network in free space would function. This provides a baseline for how the models perform in a controlled environment without the influence of external interference.

The following settings of the ns-3 simulator are used and are common to all tests. Any settings not mentioned are set at the default value. The TCP segment size is set to 1460 so that packets sent at the link layer will be 1500 bytes. The channel rate is changed so all transmissions are sent at 6 Mbps. We use the AdhocWifiMac with QoS turned on and static routing. QoS is used so control packets of the optimization protocol will be sent with a higher priority than other traffic. Normal traffic is sent using UDP so variations in the sending rate will occur due to allocation changes made by the optimization protocol and not congestion control. Control traffic sent by the optimization protocol is sent using TCP to take advantage of reliability that is built into TCP and is not rate limited like other flows. These network settings are easily duplicated in a real network.

Static routing for wireless networks over multiple hops is not implemented in ns-3. We perform preliminary experiments to determine connectivity, contention, and interference in the topology being tested. We use this information and Dijkstra's shortest path algorithm to create static routing paths for the topology being tested and update the routing tables of each node accordingly.

The default settings of the simulator, which match typical 802.11 configurations, cause the carrier sensing range and the interference range to overlap. Carrier sensing range, which determines contention, extends to 224 meters. Transmission range is 114 meters with full throughput being achieved at a maximum range of 100 meters. We require that the nodes in a link, which are being tested for loss due to interference, be 100 meters apart. This allows full throughput at the link and allows interference to the cause the largest amount of packet loss because the signal between sender and receiver is relatively weak. Using a link distance of 100 meters, interference becomes high enough to cause 100% packet loss when an interfering node is 168 meters from the receiving node. This means an interferer must be within contention range of its interferee in order to cause 100% packet loss.

In setting up the constraints of the model, Wang et al. briefly states that a clique capacity of 0.85 is used. This is because it is assumed that applications running in the network will never use all of the channel capacity due to overhead at the link layer. This is problematic when using the PI model because receive rates are calculated using an interference factor that is measured in terms of application layer throughput. Receive rates are calculated using the following model:

$$r_l = d_l s_l \prod_{i \in I(l)} (1 - a_{il} s_i). \tag{4.1}$$

d_l is the inherent loss of the link, s_l is the sending rate of the link, s_i is the sending rate of the interferer, and a_{il} is the interference factor. The interference factor represents the percentage of packets lost due to interference and is measured in terms of application capacity, not channel capacity. Because it would be very difficult to measure loss at the link layer due to interference, we set clique capacity to be 1.0 and use clique capacity to mean capacity at the application layer rather than channel capacity. Because of this difference, we compare the calculations of the PI model with the results of Wang et al. using a clique capacity of 0.85

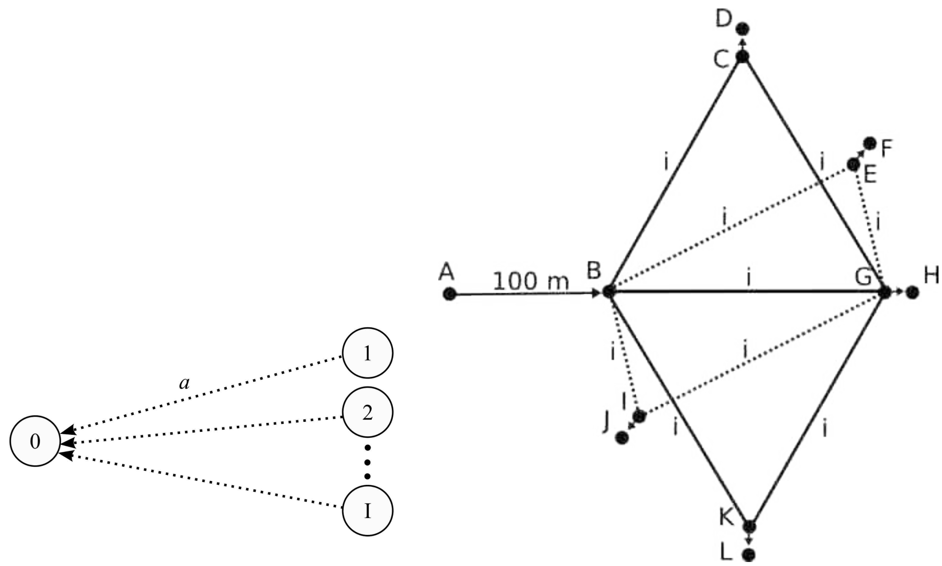
to verify the model is correctly implemented and we perform all experiments using a clique capacity of 1.0 to check model performance.

4.1 PI vs II model

We try to replicate the results of Wang et al. in the simulator, but find that the contention graphs used to test the models cannot be duplicated in a network with the default settings of the simulator under the assumptions that were used to build the optimization protocol. These contention graphs are shown in Figure 4.1(a) and Figure 4.6(a). The contention graphs require interfering nodes to be within contention range of the receiver, but outside transmission range. This fact led us to modify the settings of the simulator in order to change the contention range. Specifically, we change the EnergyDetectionThreshold in the YansWifiModel from -95 to -96 and the CcaMode1Threshold in this same model from -98 to $1.79769 * 10^{308}$. These changes have no effect on the transmission range and reduce the contention range from 224 meters to 162 meters. When duplicating the contention graphs of Wang et al. we use these modified settings.

4.1.1 N Interferers

Wang et al., in their tests, require that interferers not interfere with each other, yet still provide equal amounts of interference at the interferee link. In practice, this is difficult, if not impossible, to accomplish, so we relax the constraint such that interferers are only required to not contend with each other. We minimize the effects of interference among interfering links by setting the distance between nodes in an interfering link to one meter. This causes the signal strength between these nodes to be as strong as possible. In practice, interference at the interfering links causes at most 2% packet loss. The constraints imposed by the contention and interference ranges make it impossible to perform the tests with ten interferers. Figure 4.1(a) is the contention graph used by Wang et al., where a is the percentage of packet loss at link 0 due to interference from a single link. Figure 4.1(b) is the topology used to



(a) Original contention graph (b) Actual topology tested

Figure 4.1: Multiple Interferers: Contention graph and actual topology

approximate Figure 4.1(a), where i is the distance of each interferer from receiver B to cause packet loss equivalent to a.

There are three constraints that determine how many interferers can interferer with packets received at some node: the interfering nodes must be 163 meters from each other so they don't contend, the interferers must be 163 meters from the sending node of the interferee link so interference is possible, and interferers must be at least 168 meters from the receiver so up to 100% packet loss can be induced. Given these constraints, we can place up to six interferers so that they all have maximum interference with one link.

In the order to test the models at all levels of interference, we vary the distance, i, between interfering nodes and the receiving node to various points between 168 and 450 meters to cause anywhere from 0 to 100% packet loss. This allows us to see how the number of interferers, at all possible interference levels, affects the models. Figure 4.2 shows how interference changes as a function of distance when there is one interferer.

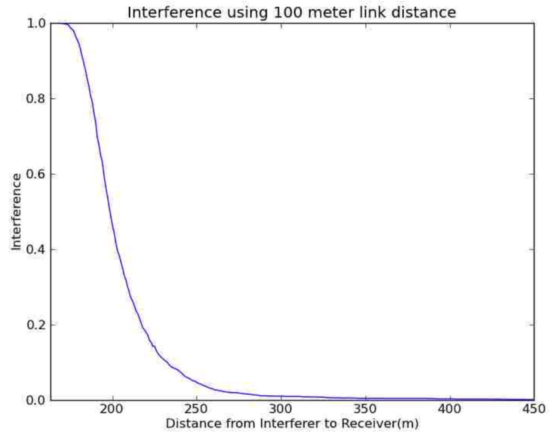


Figure 4.2: Interference as a function of distance

The PI model uses the objective function

$$f(r) = \sum_{i \in L} ln(r_l). \tag{4.2}$$

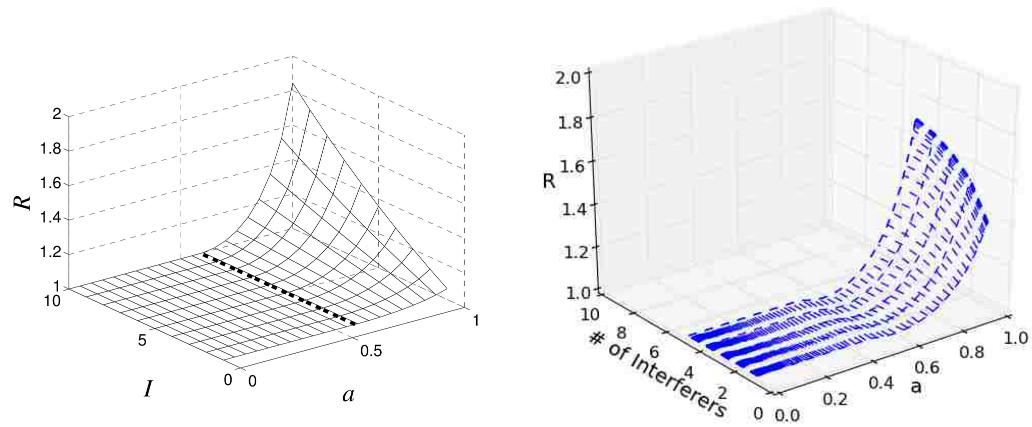
The performance function

$$P(r) = e^{f(r)/|L|}, \tag{4.3}$$

as presented by Wang et al., is used to transform the range of $f(r)$ so meaningful comparison between models can be made. A ratio, R, is used to compare the PI model to some other model, where the PI model is R times better than its counterpart. R is given by

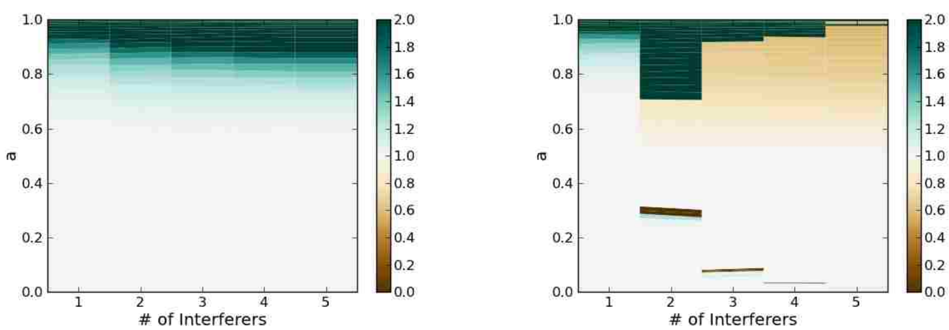
$$R = P(r^*)/P(r'). \tag{4.4}$$

Figure 4.3 shows a comparison between the utility improvement graphs of the theoretical topology of Wang et al. and the topology used in the simulations. The graphs show the expected utility improvement if the receive rate predictions of the PI model are correct in the multiple interferer case. A value of one indicates that the PI and II models perform identically. Clique capacity is set to 0.85 to so a numerically meaningful comparison can be made. However, because a clique capacity of 0.85 means different things in each of our



(a) Theoretical topology (b) Simulated topology

Figure 4.3: Clique Capacity 0.85: Utility improvement graphs in the multiple interferer case.

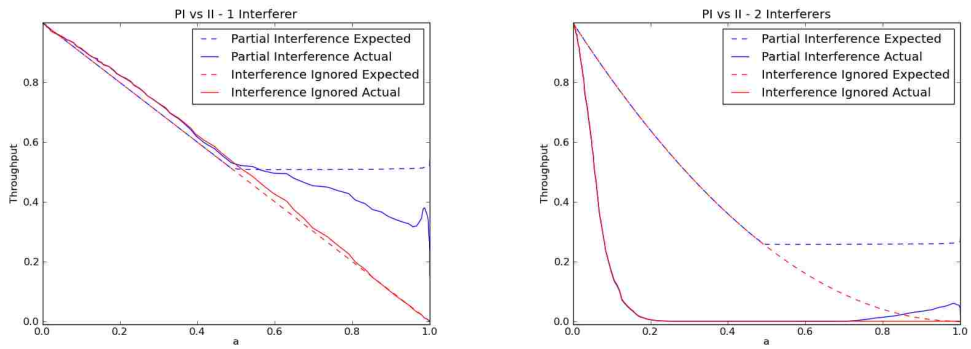


(a) PI model predicted (b) Actual

Figure 4.4: Clique Capacity 1.0: Utility improvement with multiple interferers

tests, comparing results of what actually happened would be meaningless. We make this comparison to show the differences between the theoretical topologies and those used in the simulator. As can be seen, if a clique capacity of 0.85 represented full application throughput, the PI model is expected to perform even better using the more realistic topology.

When the tests are performed with the clique capacity set to 1.0, the PI model performs slightly worse than expected in the one interferer case. In this case, the PI model always performs as well as the II model, but when interference is above 0.6, the performance gains are slightly below what is expected. Figure 4.5(a) shows how the rate at the interferee link doesn't remain as high as expected. The interferer is decreasing its sending rate, but the interferee is not getting the expected increase in throughput. This is an indication that



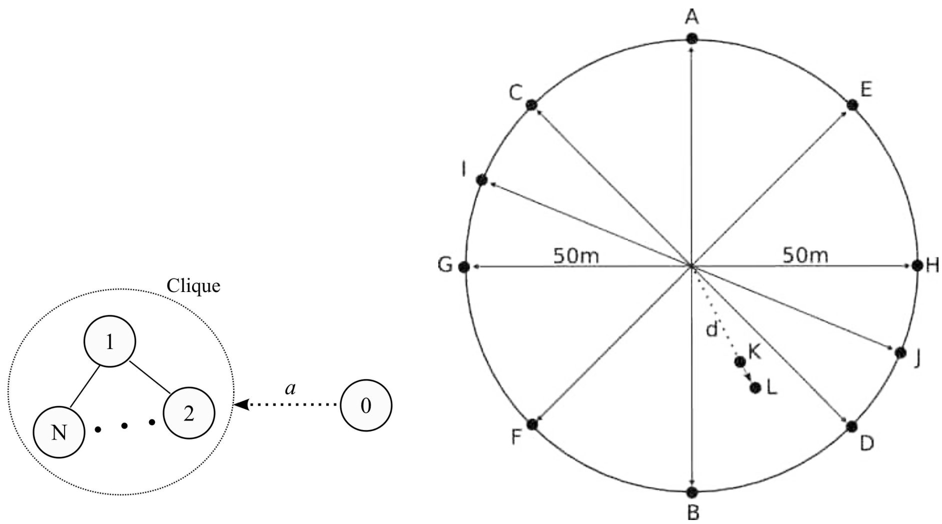
(a) Interferee Link with 1 Interferer

(b) Interferee Link with 2 Interferers

Figure 4.5: Clique Capacity 1.0: Rate of interferee with multiple interferers

the interference-to-rate relationship is not entirely linear. In other words, decreasing the interferer's rate doesn't always give the gains at the interferee that are expected.

When there are multiple interferers causing interference, the PI model does not correctly characterize receive rates at the interferee link. In the simulator, interferers are not independent and so, interference is not additive. This contradicts work done by Niculescu [8], where this property is shown to hold in his mesh test bed. The simulator shows interference is drastic in this case; its effects on the interferee link in the two interferer case is shown in Figure 4.5(b). In Figure 4.4(b), when the interference is above 0.5, the PI model starts to perform worse than the II model because it's decreasing the rate of interferers and the decrease is having no effect at the interferee. Eventually the utility abruptly goes to infinity because the rate of the interferers drop enough that a few packets make it through. Before this point both models cause the throughput at the interferee link to be at 0. The utility of 0 is $-\infty$, but since both models have a utility of $-\infty$, on the utility comparison graph it is shown as 1. This means that once any number of packets get through for one of the models, if the other model is still causing 0% throughput at the interferee, the utility of the former is infinitely higher. This explains the anomalies when interference is below 0.5 for the multiple interferer cases where both model are giving the exact same allocations to all the



(a) Theoretical contention graph (b) Actual topology tested

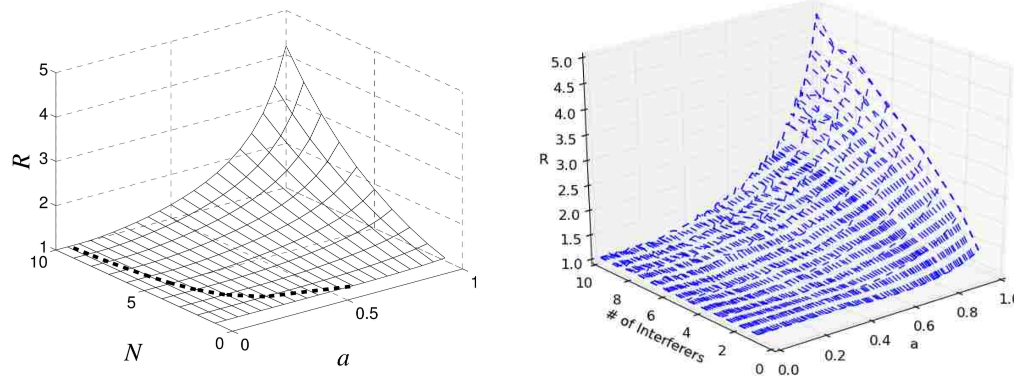
Figure 4.6: Multiple Contenders: Contention graph and actual topology

nodes. Throughput is already at 0 for the interferee link in those cases and one model just happened to get a few packets through causing the utility to go to ∞ for that model.

4.1.2 N Contenders

Wang et al. uses the topology in Figure 4.6(a) to see how the PI model compares when there are a large number of contenders and a single interferer. Figure 4.6(b) shows how we approximate this by creating a circle of nodes, where each node is 100 meters from the node directly across from it. Each node sends packets to the node directly across from it allowing us to create 10 contending links. Node K is on a plane that is perpendicular to the plane of the circle of nodes. K sends packets to L, which is always one meter from K. K is d meters from the center of the circle, where d is a variable distance that allows K to be anywhere from 168 to 450 meters away from all nodes in the circle.

The topology is different from Figure 4.6(a) in that there are many potential links that are not used, but must be considered by the PI model when calculating sending rates. For example, there is a link from node A to B that is used, but there is also a potential link from node A to C. When node K is calculating its optimal sending rate, the PI model does



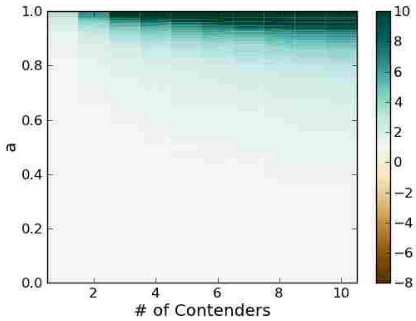
(a) Theoretical topology (b) Simulated topology

Figure 4.7: Clique Capacity 0.85: Utility improvement graphs with multiple contenders.

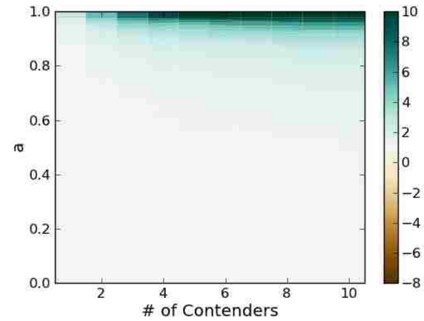
not check if the link from A to C is active. It simply assumes that all links that are interfered with will be sending at the full rate and acts accordingly. This means the PI model should be more conservative in allocating bandwidth to interferers if there are a large number of interferee links that are not being used.

Figure 4.7 shows a comparison between the utility improvement graphs of the theoretical topology of Wang et al. and the topology used in the simulations when testing the multiple contender case. These results are based on the PI model predicted receive rates. We perform these tests with all ten contenders and, as can be seen, the results are very similar to what is expected. The utility comparison graph for our topology would results in a slight increase in utility over the theoretical topology if a clique capacity of 0.85 made sense in our tests.

We test the performance of the multiple contender topology in 4.6(b) using one interferer with a clique capacity of 1.0. The PI model has the same problem as in the multiple interferer case when there is one contender. The PI model does not perfectly predict receive rates when there is one interferer and interference is above 0.6. Figure 4.9 shows that as the number of contenders increases, the PI model actually becomes more accurate and then begins to over-predict the expected receive rate by a slightly larger and larger margin. This over-prediction is reflected in the utility graphs in Figure 4.8 where the utility is not as high

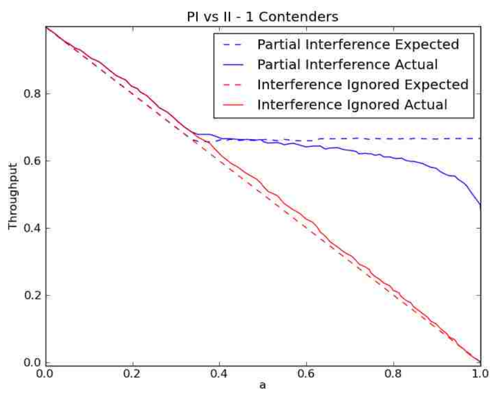


(a) PI model predicted

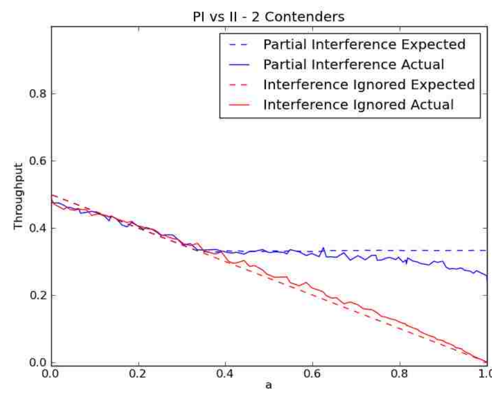


(b) Actual

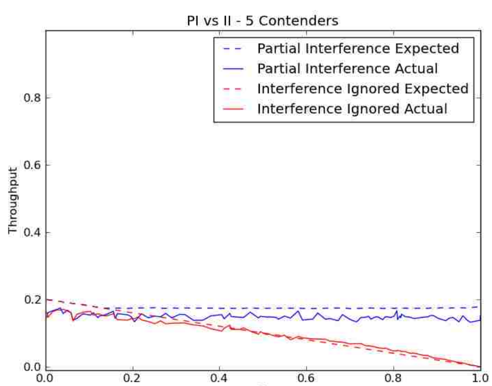
Figure 4.8: Clique Capacity 1.0: PI to II utility with multiple contenders, one interferer



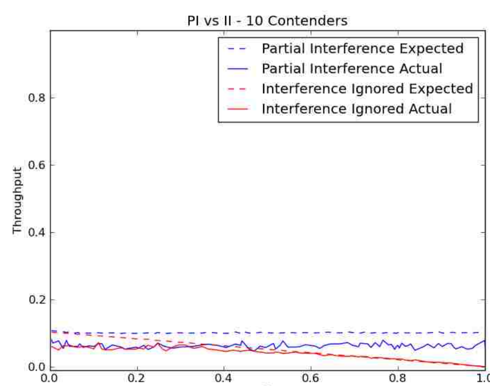
(a) Interferee Link with 1 Contender



(b) Interferee Link with 2 Contender



(c) Interferee Link with 5 Contender



(d) Interferee Link with 10 Contender

Figure 4.9: Clique Capacity 1.0: PI to II rates with multiple contenders, one interferer

as expected. It is possible that capacity of the clique which contains all the contenders is decreasing as the number of contenders increases. This would explain why the PI model over-predicts receive rates as the number of contending links increases.

4.2 IC and AC Models

A direct comparison to the theoretical results presented in Wang et al. is impossible for the IC model. The IC model treats all interfering nodes as though they contend, however, in the optimization model all contending nodes are required to communicate with each other. The topologies described by Wang et al. prevent interferers from being able to communicate with their interferees. Instead, we create a modified topology to try to allow communication between the interferers and interferees. We use the same topology shown in Figure 4.1(b), but add a string of nodes from the interferers to interferees. These intermediate nodes are used to only exchange control traffic. Unfortunately, interference and contention levels are too high using the modified settings of the simulator for control traffic to consistently travel between all nodes in a clique. We, therefore, do not test the IC model using the topologies of Wang et al. Likewise, we do not test the AC model, because it is dependent on the IC model.

As a side note, it should be mentioned that the IC model does not cause interference to be treated as contention at the MAC layer. When two nodes contend at the MAC layer, they cannot transmit at the same time. If one is transmitting, the other will wait until its predecessor is done. The IC model simply sets the rates of the links as if they were contending. If no contention is detected at the MAC layer, these nodes can still transmit at the same time and cause interference. By lowering the rate of the contending nodes, the IC model reduces the likelihood that interference will occur, but cannot explicitly prevent it. If the IC model were to tune the CcaMode1Threshold so that interfering nodes were forced to contend at the MAC layer, interfering links would be prevented from causing interference. Because interference is not prevented, it can have harmful effects on the control traffic of the optimization protocol.

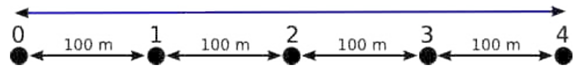


Figure 4.10: Chain topology

It is important to note that the AC model is dependent on the PI model. The AC model uses expected utility to determine whether the II or the IC model will perform better. This expected utility is calculated using the PI model. This is done because the utility of a model is based on the receive rates throughout the network. Any node using the AC model has no way of knowing what the receive rates will be when trying to determine if the II or IC model is better. For this reason, the AC model uses the PI model to predict receive rates and then chooses the model that is supposed to perform better based on these predictions. Some of the underlying assumptions of the PI model have been shown to be flawed, and so, in turn, the AC model will not always be using the model that will, in actuality, perform better.

4.3 General Mesh Testing

We perform a set of tests using mesh topologies to give an indicator of how well the four models perform in realistic settings. The simulator is reconfigured to so the contention threshold is at its default value.

4.3.1 Chain Topology

The chain topology is shown in Figure 4.10. It is a common topology in mesh networks where flows must traverse multiple hops from one end of the chain to the other. Each node in 100 meters away from its neighbor, which is the maximum transmission distance where full throughput is achieved. As shown in Table 4.1, interference primarily occurs when receiving packets at nodes 1 and 3. In the interference tables, row links interfere with column links.

Throughput when four flows are active is illustrated in Figure 4.11. We might expect the PI model to perform better because it will account for interference, however, in this case interference is not the limiting factor. All models perform equally because flows can only go as

	$0 \to 1$	$1 \to 0$	$1 \to 2$	$2 \to 1$	$2 \to 3$	$3 \to 2$	$3 \to 4$	$4 \to 3$
$0 \to 1$	0	0	0	0	0	0.5938	0.008	0.0505
$1 \to 0$	0	0	0	0	0	0	0	0.424
$1 \to 2$	0	0	0	0	0	0	0	0.424
$2 \to 1$	0	0	0	0	0	0	0	0
$2 \to 3$	0	0	0	0	0	0	0	0
$3 \to 2$	0.5924	0	0	0	0	0	0	0
$3 \to 4$	0.5924	0	0	0	0	0	0	0
$4 \to 3$	0.0472	0.0077	0.6170	0	0	0	0	0

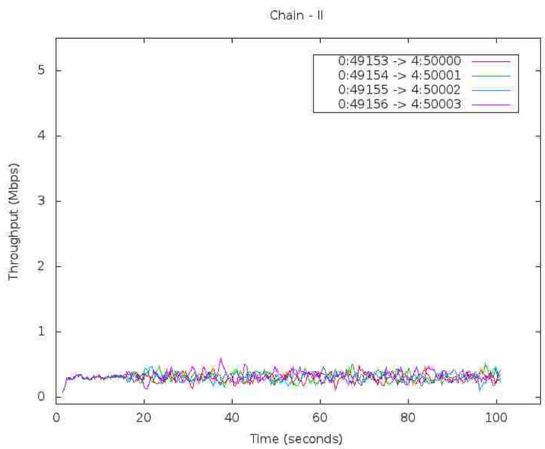
Table 4.1: Link interference in chain topology

fast as the slowest link. The PI, II, and AC models give different allocations to each link while the IC gives the same allocation to all links, but all models have a similar allocation to the bottleneck link. This topology illustrates the limitations link-based formulations encounter when applied to flows that traverse multiple hops.

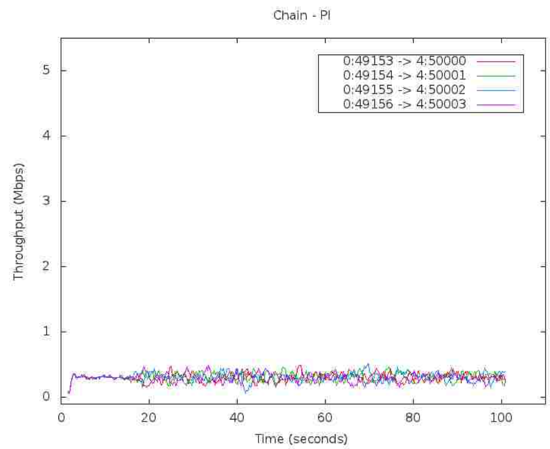
Throughput for two flows, flowing in opposite directions, is illustrated in Figure 4.12. These flows cause every node in the topology to be transmitting throughout the test. In this case, the IC models performs slightly better than the other models. This, again, is attributed to different links having different rate allocations. Because the IC model gives all links the same allocation, it becomes possible for the nodes to better synchronize sending, so throughput is slightly higher. The other models do not promote effective sharing of the medium, due to non-uniform link allocation along the path of the flow.

4.3.2 Man-in-the-Middle Topology

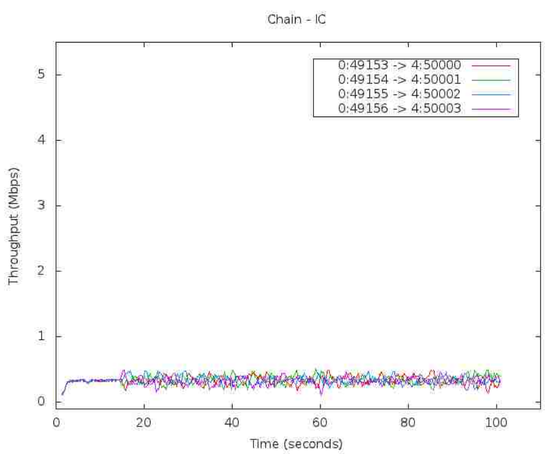
The man-in-the-middle topology, shown in Figure 4.13, is designed to test the effects of two links that contend with a common link, but not with each other. In setting up this topology, it became apparent that this particular topology would only rarely be seen in practice when using an optimization protocol. In order for the outer nodes to contend with the inner nodes and not with each other, the outer nodes must be at the extreme edge of transmission range of the inner node. Nodes must be 225 meters away from each other to not contend and transmission range is 114 meters, with full throughput achieved at 100 meters. Figure 4.13



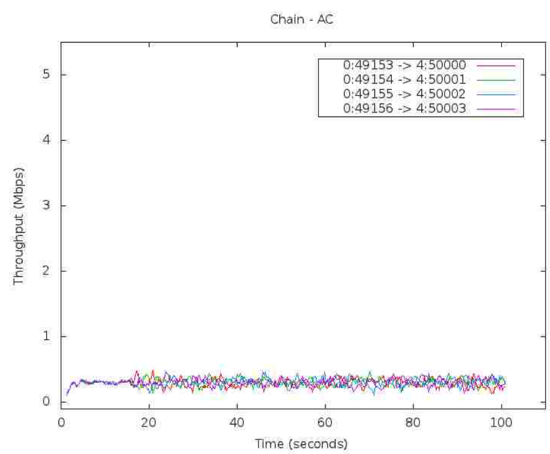
(a) II model



(b) PI model

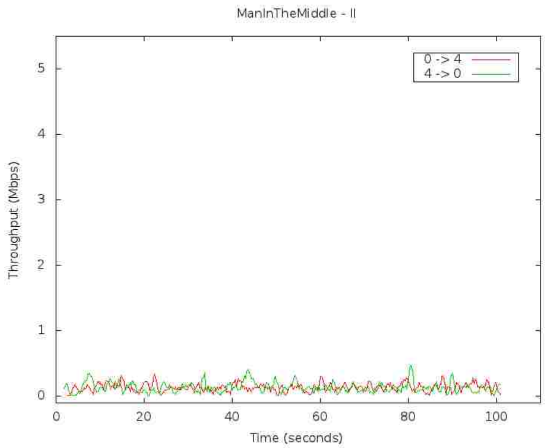


(c) IC model

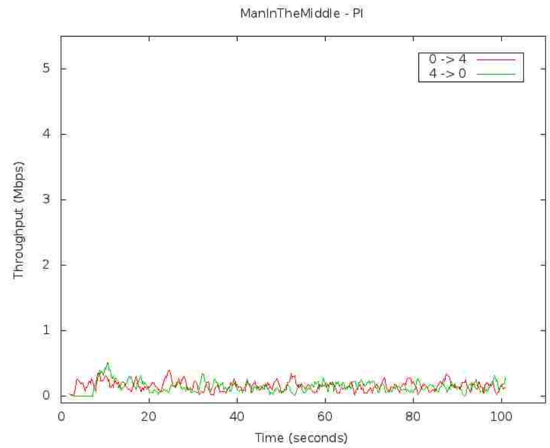


(d) AC model

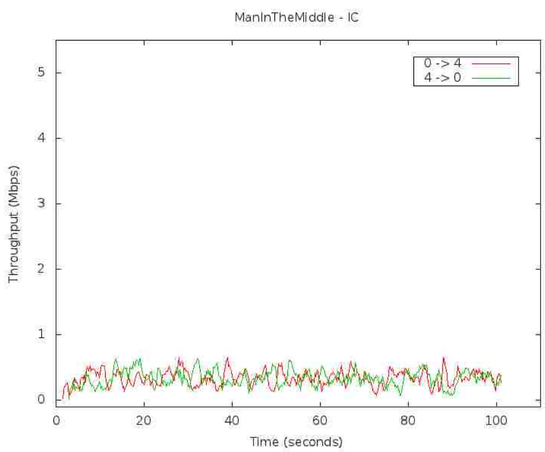
Figure 4.11: Chain topology: Four flows



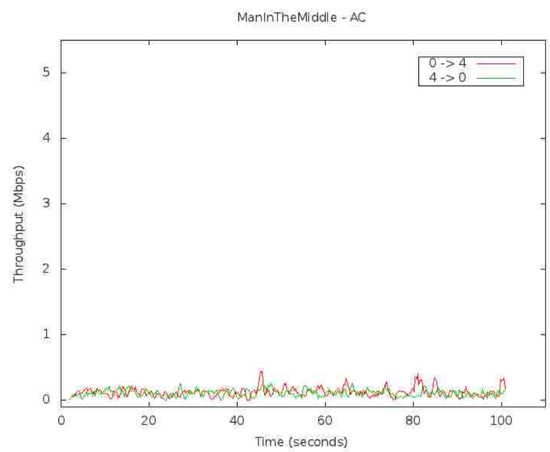
(a) II model



(b) PI model



(c) IC model



(d) AC model

Figure 4.12: Chain topology: Two flows in opposite directions

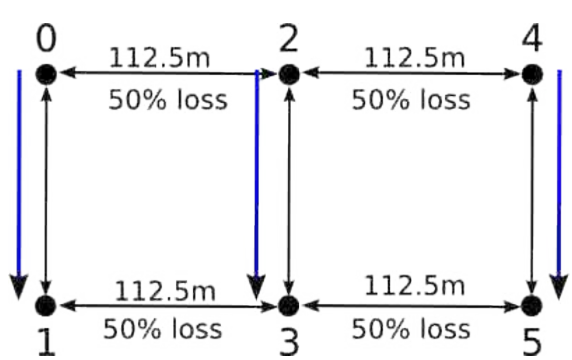


Figure 4.13: Man-in-the-Middle Topology

illustrates how nodes 0 and 4 are 112.5 meters from node 2 making them barely out of contention range of each other. The links from exterior to interior nodes experience a 50% loss due to weak signal strength at such a large distance. An exterior node cannot be any closer or it will contend with its counterpart on the other side of the topology and it can't be moved more than an additional 1.5 meters away from the interior node or the link to that node will disappear. Remember, all nodes that contend with each other must be able to communicate with each other. If this invariant is violated, the optimization protocol can't function.

While the man-in-the-middle topology is an edge case, it does illustrate a more general problem. Because the exterior nodes are barely out of contention range of each other, they cause the absolute maximum amount of interference at interior nodes. Table 4.2 shows the percentage of loss due to interference. When interference is at this extreme value, control traffic from exterior to interior nodes is knocked out, preventing the optimization protocol from functioning. As a temporary fix to allow testing of what the convergence would have been, we force the optimization protocol to go through at least 5 rounds of rate allocation distribution before each node will change the rate of the links they control. This means each node with receive rate allocation information from all other nodes in its clique at least 5 times before setting the actual rate of the links controlled by that node.

Figure 4.14 shows the allocations made for each link. Notice how allocations stop almost completely for the IC and AC models and allocations cease half way through the test for the II model. The PI model is the only one that maintains a somewhat consistent rate allocations throughout the test, even if those allocations are 10 to 20 seconds apart. The associated throughput of the flows using each model is shown in Figure 4.15.

This test illustrates the potential gains of using a model like the PI model that accounts for interference. It was previously shown that the PI model is inaccurate when interference is above 50%, however, the model still provides notable performance improvement. This improvement in performance is necessary for the network to function when interference levels

	$0 \to 1$	$0 \to 2$	$1 \to 0$	$1 \to 3$	$4 \to 2$	$4 \to 5$	$5 \to 3$	$5 \to 4$
$0 \to 1$	0	0	0	0	1	0.0691	1	0.1145
$0 \to 2$	0	0	0	0	1	0.0691	1	0.1145
$1 \to 0$	0	0	0	0	1	0.1172	1	0.0566
$1 \to 3$	0	0	0	0	1	0.1172	1	0.0566
$2 \to 0$	0	0	0	0	0	0	0	0
$2 \to 3$	0	0	0	0	0	0	0	0
$2 \to 4$	0	0	0	0	0	0	0	0
$3 \to 1$	0	0	0	0	0	0	0	0
$3 \to 2$	0	0	0	0	0	0	0	0
$3 \to 5$	0	0	0	0	0	0	0	0
$4 \to 2$	0	0	1	1	0	0	0	0
$4 \to 5$	0	0	1	1	0	0	0	0
$5 \to 3$	0	1	0.0012	1	0	0	0	0
$5 \to 4$	0	1	0.0012	1	0	0	0	0

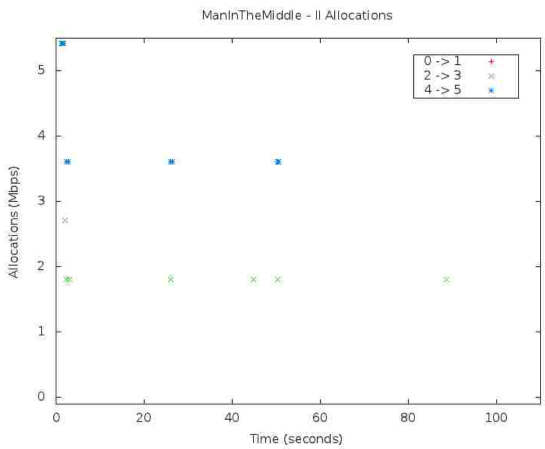
Table 4.2: Link interference in man-in-the-middle topology

are high. There could be nodes that are multiple hops away from a sender that are also just outside of contention range of that sender. These nodes would be free to interfere at potentially harmful levels. For example, if nodes are placed along a straight line at 0, 100, 165, and 225 meters, the node at 225 meters can cause critical amounts of interference for the node at 100 meters when it's receiving.

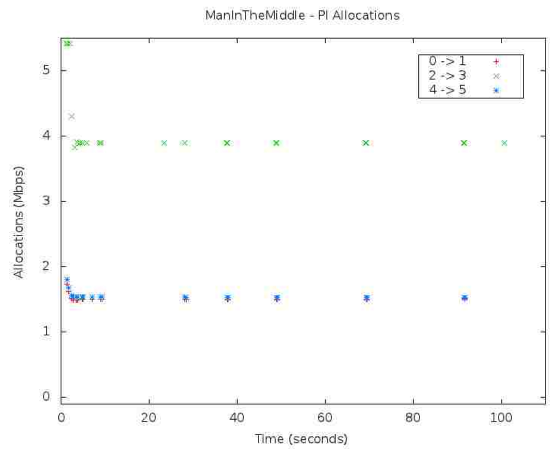
4.3.3 Grid Topology

Figure 4.16 shows the grid topology used. Nodes are 100 meters apart in the horizontal and vertical axes. Because of the density of this network, each node must exchange rate information with at least seven other nodes. Nodes in the middle of the topology must coordinate with all but one of the nodes in the topology. Figure 4.17 illustrates the rate allocations and how often all the nodes are able to coordinate with nodes in their cliques.

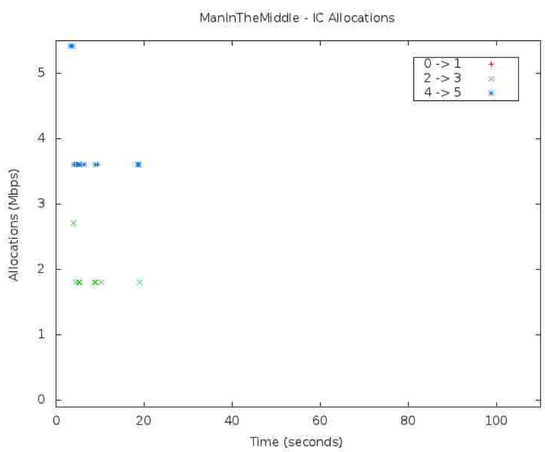
The PI model allows the most consistent communicate among all nodes. The II model performs slightly worse in this regard, and the IC and AC models are not able exchange control traffic once allocations become active. In Figure 4.18, we see that throughput for each of the flows is most evenly distributed using the PI model. The II, IC, and AC models cause



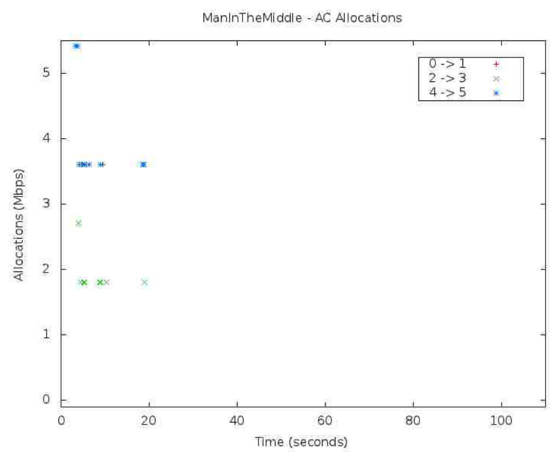
(a) II model: Allocations



(b) PI model: Allocations

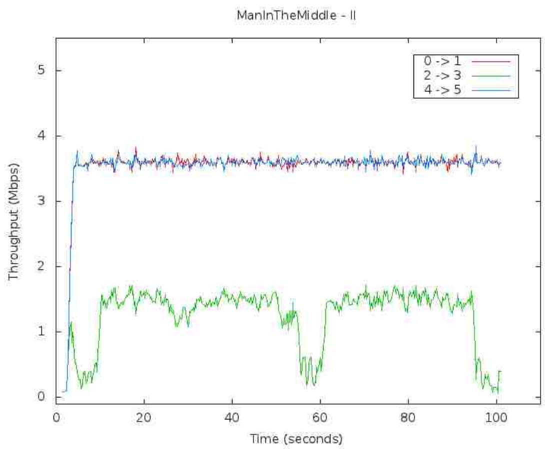


(c) IC model: Allocations

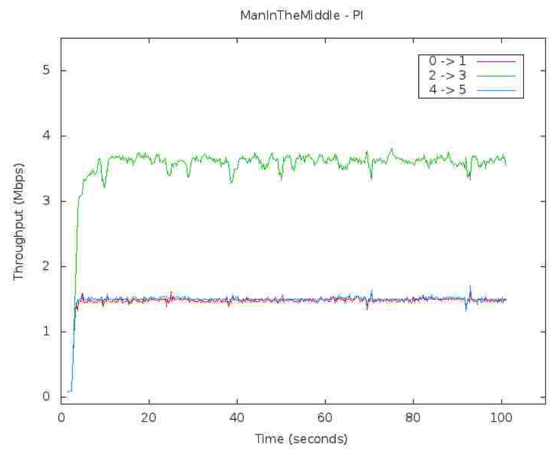


(d) AC model: Allocations

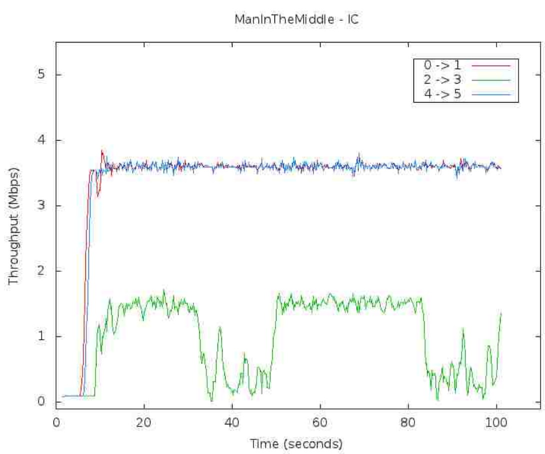
Figure 4.14: Man-in-the-middle: Link allocations



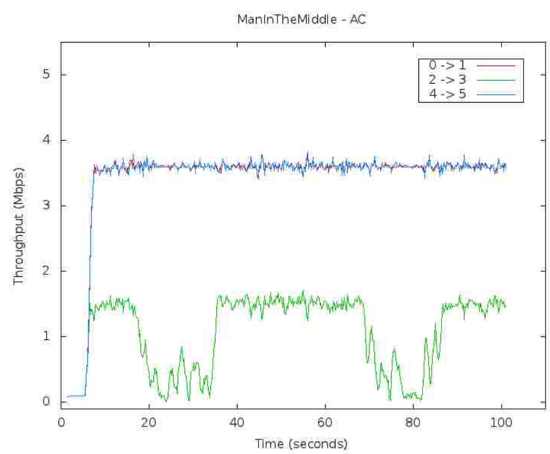
(a) II model: Throughput



(b) PI model: Throughput



(c) IC model: Throughput



(d) AC model: Throughput

Figure 4.15: Man-in-the-middle: Throughput of three parallel flows

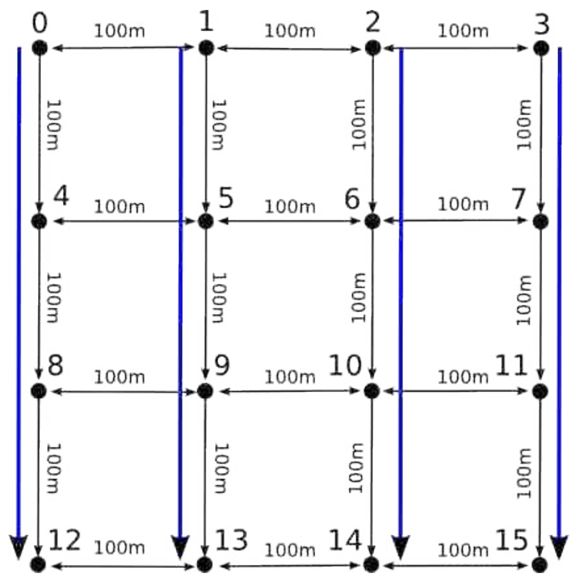
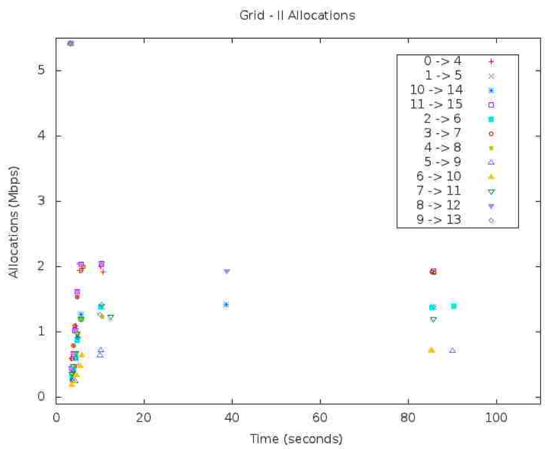


Figure 4.16: Grid topology

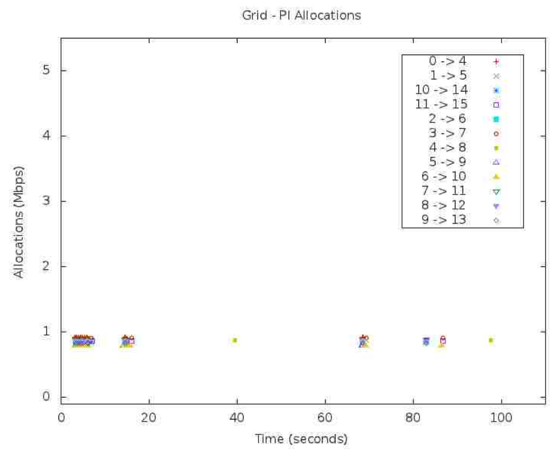
the flows on the edges of the topologies to starve those flows in the center. This topology is a good example of how the PI model promotes fairer bandwidth allocation and better communication throughout the network than other models.

4.3.4 Single vs. Double Hop Flow

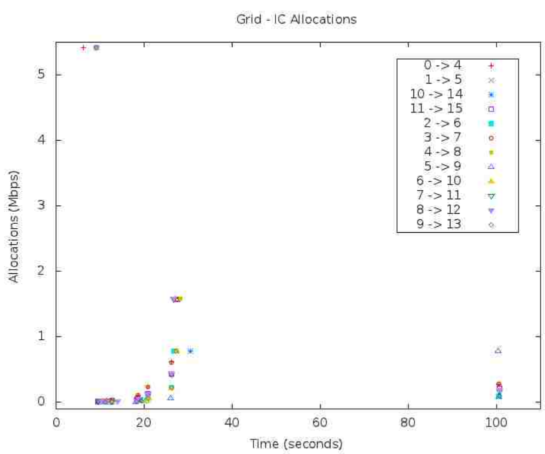
We test the topology in Figure 4.19 to test how optimal rate allocation affects competing flows where one flow traverses a single hop, and the other traverses two hops. Shi et al. discuss how single-hop flows can starve multi-hop flows using TCP in a mesh network [12]. We test each rate allocation protocols in this scenario using UDP flows and show that all models perform identically. Both links contend and there is no interference, so identical allocations will be assigned by each model. Figure 4.20(a) shows how, without rate control a one hop UDP flow completely starves a two hop UDP flow. Figure 4.20(b) shows the throughput gains of an optimization protocol. Notice how in Figure 4.20(c), the allocations to each link are the same. Despite this, the MAC is still unfair. Formulating the problem as an optimization of flows, rather than links, would help alleviate this problem.



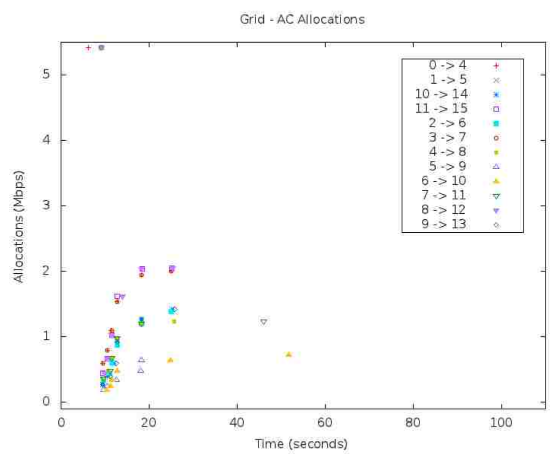
(a) II model: Allocations



(b) PI model: Allocations

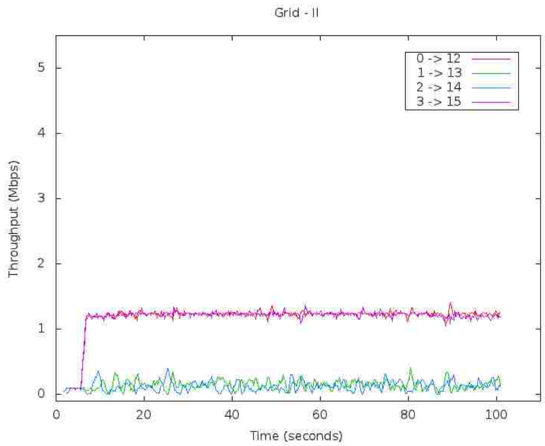


(c) IC model: Allocations

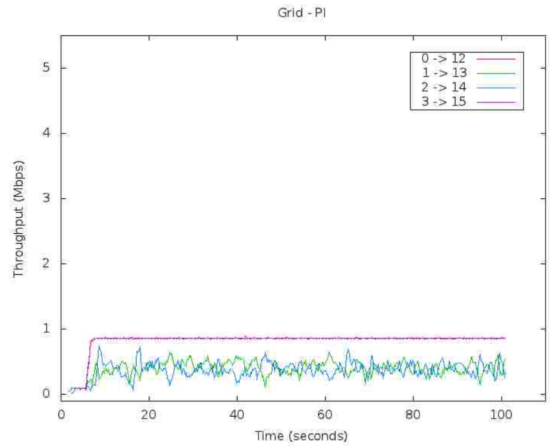


(d) AC model: Allocations

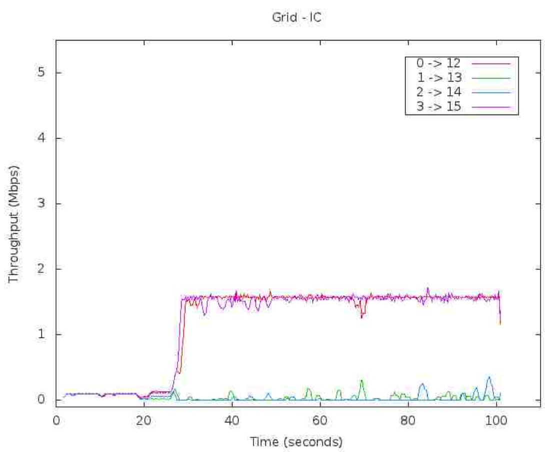
Figure 4.17: Grid: Link allocations



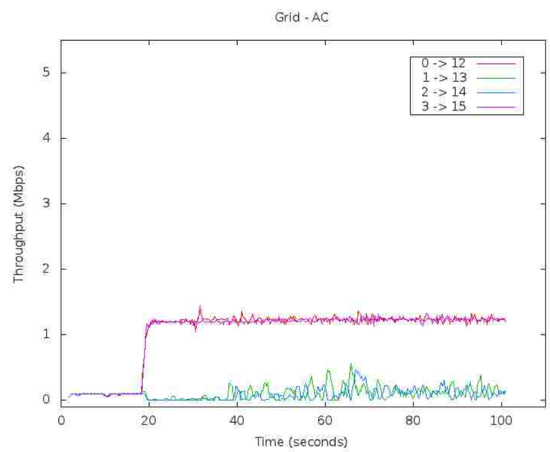
(a) II model: Throughput



(b) PI model: Throughput



(c) IC model: Throughput



(d) AC model: Throughput

Figure 4.18: Grid: Throughput of four parallel flows

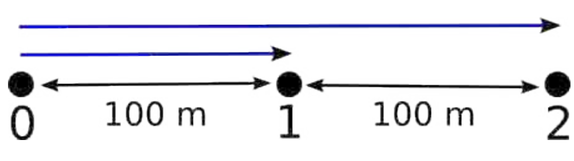
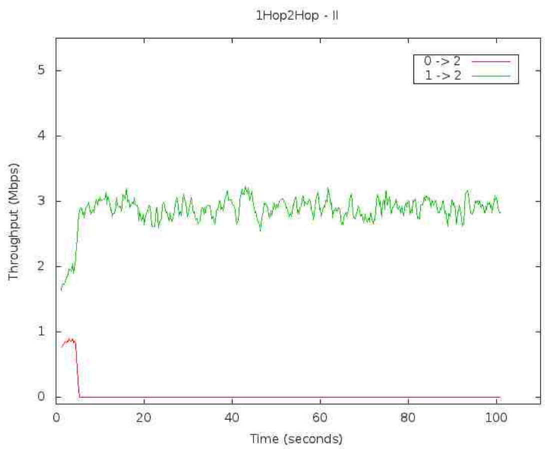
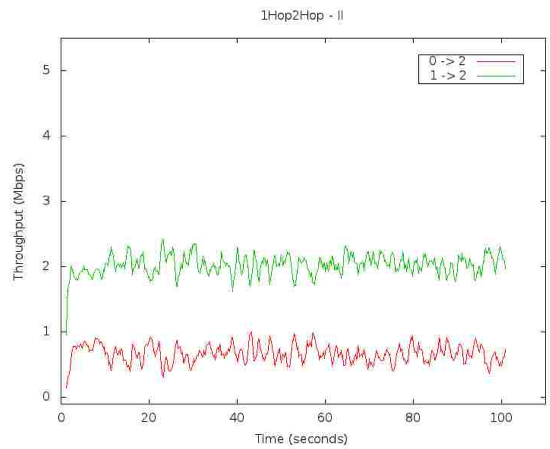


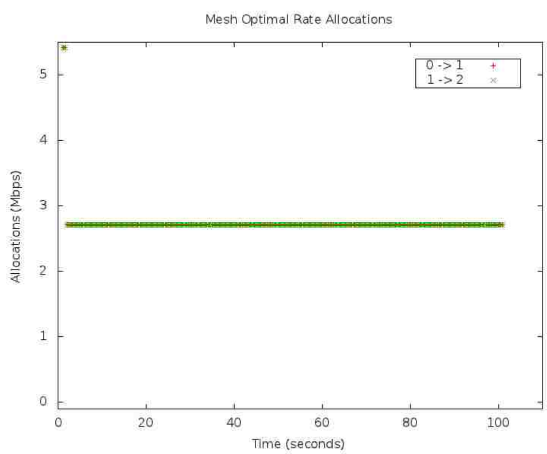
Figure 4.19: Single vs. double hop topology



(a) No Rate Control

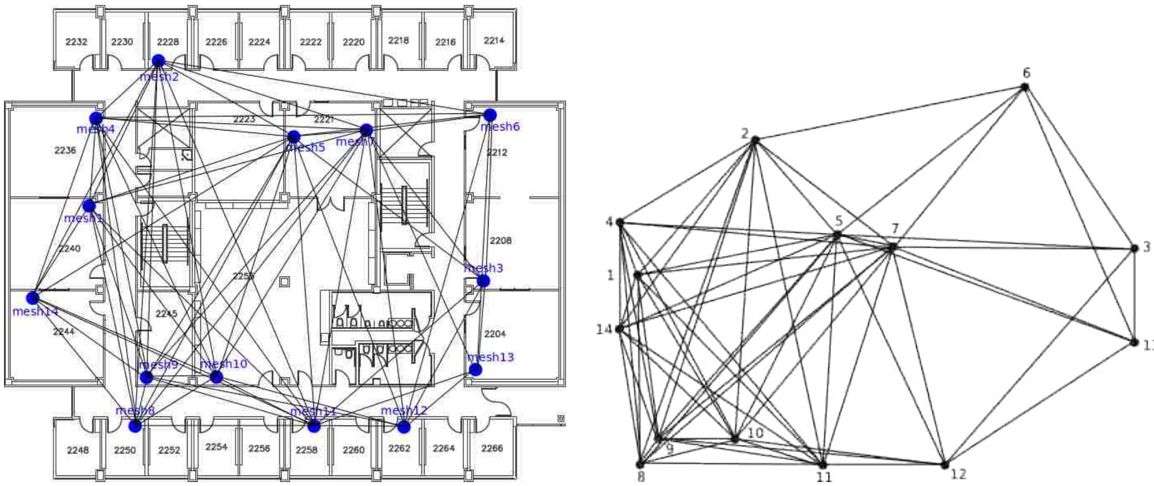


(b) Rate Control



(c) Rate Control

Figure 4.20: Single vs. double hop throughput comparison

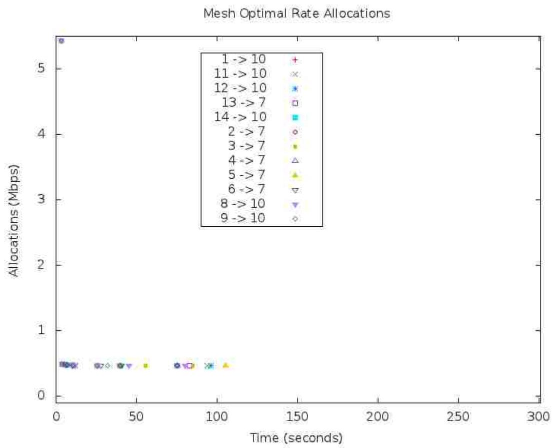


(a) Actual Mesh (b) Simulated Mesh

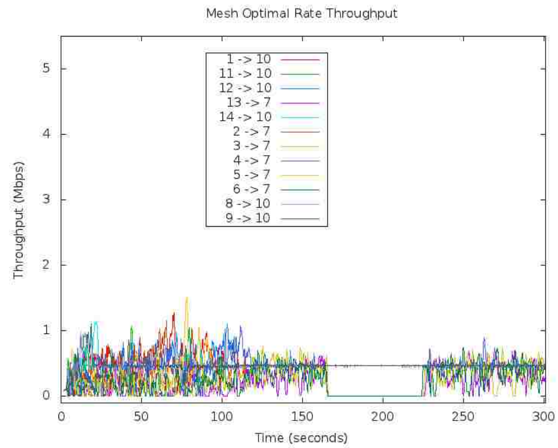
Figure 4.21: Mesh topology

4.3.5 Mesh Network Approximation

Figure 4.21 shows a picture of our mesh network and the mesh topology we created in the simulator. When trying to duplicate the links of our mesh network in the simulator, we find that the simulator causes all the links to contend with each other. We designate nodes 7 and 10 as Internet access points and send all traffic to whichever access point is closer. This causes 12 flows to be active, with 6 flows going to each of the gateways. All flows are one hop. In this scenario, all models perform the same because, like the one-hop to two-hop scenario, all links contend with one another. This forces each node to coordinate its rate with all other nodes in the network. Figure 4.22 shows how as the throughput settles into the given allocations at around 120 seconds, nodes are no longer able to coordinate rate information using their control traffic. However, Figure 4.23 shows that when link capacity is reduced down to 55%, control traffic is able to provide consistent coordination between all nodes in the network. Throughput is also stable and consistent, unlike when links are allowed to send at expected capacity. It is possible that the capacity of the network decreases as the number of contenders increases, due to more overhead as nodes try to gain access to the medium. This may account for the starvation of control traffic when contention is high.

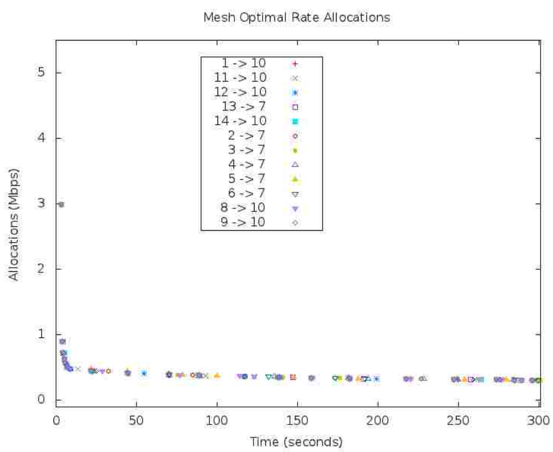


(a) Mesh Allocations

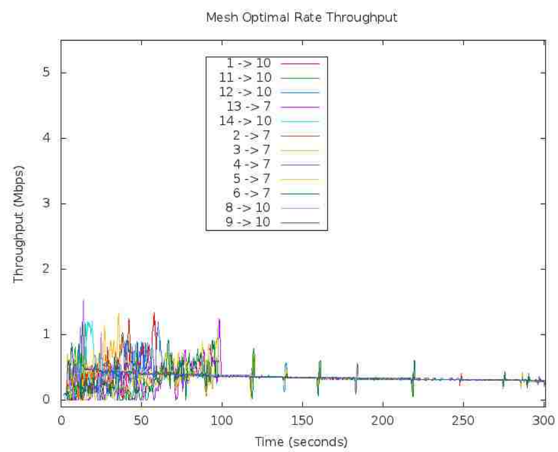


(b) Simulated Mesh

Figure 4.22: Simulated mesh: 12 flows at full capacity



(a) Mesh Allocations



(b) Simulated Mesh

Figure 4.23: Simulated mesh: 12 flows at 55% capacity

Chapter 5

Conclusions

In this thesis we have used the ns-3 simulator to generalize experimental work done by Wang et al. and tested the II, IC, AC, and PI models in various mesh topologies to compare functionality. We show that the PI model is based on two assumptions that do not hold in simulator, namely that the relationship between interference and the interferer rate is always linear and that interference from multiple interferers is additive. This causes the PI model to not always correctly predict receive rates. Despite these flawed assumptions, we've shown that the PI model still provides performance improvements over the three BC variants. The PI model provides fairer bandwidth throughput for flows in a dense network topology. The PI model also causes control traffic to be more consistent than other models. We show that control traffic is starved when contention is high. It is possible that when contention is high, network capacity may decrease, which is not accounted for by these models. These questions regarding contention and network capacity, the interference to interferer rate relationship, and how interference is affected by multiple interferers, must be further validated in a real network.

References

[1] Nicola Baldo, Manuel Requena-Esteso, Josê Nûòez-Martînez, Marc Portolés-Comeras, Jaume Nin-Guerrero, Paolo Dini, and Josep Mangues-Bafalluy. Validation of the IEEE 802.11 MAC model in the ns3 simulator using the EXTREME testbed. In *Proceedings of 3rd International ICST Conference on Simulation Tools and Techniques*. ACM, March 2010. doi: 10.4108/ICST.SIMUTOOLS2010.8705. URL `http://dl.acm.org/citation.cfm?id=1808224`.

[2] Brahim Bensaou and Zuyuan Fang. A fair MAC protocol for IEEE 802.11-based ad hoc networks: Design and implementation. *IEEE Transactions on Wireless Communications*, 6(8):2934–2941, August 2007. doi: 10.1109/TWC.2007.051014. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4290035`.

[3] Randy Buck, Rich Lee, Phil Lundrigan, Charles Knutson, and Daniel Zappala. WiFu: A software toolkit for experimental wireless transport protocols. in preparation, 2012.

[4] Lijun Chen, Steven H. Low, and John C. Doyle. Joint congestion control and media access control design for ad hoc wireless networks. In *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 2212–2222, March 2005. doi: 10.1109/INFCOM.2005.1498496. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1498496&tag=1`.

[5] Jing Deng, Ben Liang, and Pramod K. Varshney. Tuning the carrier sensing range of IEEE 802.11 MAC. In *Proceedings of IEEE Global Telecommunications Conference*, volume 5, pages 2987–2991, December 2004. doi: 10.1109/GLOCOM.2004.1378900. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1378900`.

[6] Steven H. Low and David E. Lapsley. Optimization flow control–I: Basic algorithm and covergence. *IEEE/ACM Transactions on Networking*, 7(6):861–874, December 1999. doi: 10.1109/90.811451. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=811451`.

[7] Thyagarajan Nandagopal, Tae-Eun Kim, Xia Gao, and Vaduvur Bharghavan. Achieving MAC layer fairness in wireless packet networks. In *Proceedings of the 6th Annual

International Conference on Mobile Computing and Networking, pages 87–98, August 2000. doi: 10.1145/345910.345925. URL http://dl.acm.org/citation.cfm?id=345925&preflayout=tabs.

[8] Dragoş Niculescu. Interference map for 802.11 networks. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, pages 339–350, August 2007. doi: 10.1145/1298306.1298355. URL http://dl.acm.org/citation.cfm?id=1298355.

[9] Guangyu Pei and Thomas Henderson. Validation of OFDM error rate model in ns-3, May 2009. URL http://www.nsnam.org/~pei/80211b.pdf.

[10] Guangyu Pei and Thomas Henderson. Validation of ns-3 802.11b PHY model, May 2009. URL http://www.nsnam.org/~pei/80211b.pdf.

[11] David Ripplinger, Sean Warnick, and Daniel Zappala. *First-Principles Modeling of Wireless Networks for Rate Control*. Master's thesis, Brigham Young University, July 2011.

[12] Jingpu Shi, Omer Gurewitz, Vincenzo Mancuso, Joseph Camp, and Edward W. Knightly. Measurement and modeling of the origins of starvation in congestion controlled mesh networks. In *Proceedings of the 27th Conference on Computer Communications*, pages 1633–1641, April 2008. doi: 10.1109/TNET.2009.2019643. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5164899.

[13] Lei Wang. *Modeling and Designing Fair Rate Control for Wireless Mesh Networks With Partial Interference*. PhD thesis, Brigham Young University, December 2011.

[14] Lei Wang, David Ripplinger, A. Rai, S. Warnick, and D. Zappala. A convex optimization approach to decentralized rate control in wireless networks with partial interference. In *Proceedings of the 49th IEEE Conference on Decision and Control*, pages 639–646, December 2010. doi: 10.1109/CDC.2010.5717212. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5717212.