

Modeling and Analysis of Generalized Slotted-Aloha MAC Protocols in Cooperative, Competitive and Adversarial Environments*

Richard T.B. Ma, Vishal Misra, Dan Rubenstein
tbma@ee.columbia.edu, misra@cs.columbia.edu, danr@ee.columbia.edu
Columbia University

Abstract

Aloha [1] and its slotted variant [2] are commonly deployed Medium Access Control (MAC) protocols in environments where multiple transmitting devices compete for a medium, yet may have difficulty sensing each other's presence. This paper models and evaluates the throughput that can be achieved in a system where nodes compete for bandwidth using a generalized version of slotted-Aloha protocols. We evaluate the channel utilization and fairness of these types of protocols for a variety of node objectives, including maximizing aggregate throughput of the channel, each node greedily maximizing its own throughput, and attacker nodes that attempt to jam the channel. If all nodes are selfish and greedily attempt to maximize their own throughputs, a situation similar to the traditional Prisoner's Dilemma[3] arises. Our results reveal that under heavy loads, greedy strategies reduce the utilization, and that attackers cannot do much better than attacking during randomly selected slots.

1 Introduction

In many communication networks, the communication medium is often shared by multiple users who must compete for access. In Ethernet[4], nodes use CSMA/CD [5, 6] as a MAC protocol. However, for wireless ad-hoc networks or sensor networks, carrier sensing may not be effective. This is because nodes may not be able to sense one another's presence, yet their transmissions may still interfere. Ad hoc networks, sensor networks, and competing "hotspot" 802.11 gateways are examples where the "hidden terminal problem" occurs. The Aloha protocol [1] is a fully decentralized medium access control protocol that

*This material was supported in part by the National Science Foundation under CAREER Awards CNS-0133829 and CNS-0238299, and by gifts from the Intel IT Research Council and Microsoft Research. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

does not perform carrier sensing. The subsequent slotted-Aloha[2] protocol was introduced to improve the utilization of the shared medium by synchronizing the transmission of devices within time-slots.

In this work, we consider a generalization of the slotted-Aloha protocol. Like slotted-Aloha, the decision to transmit within a slot has a random component. However, in traditional slotted-Aloha, the user continues transmission in subsequent slots until a subsequent collision. In our generalized version, the user may cease transmitting with some fixed (non-zero) probability. We model a system of N users implementing this generalized protocol with tunable parameters via Markov Models that allow us to measure the rate at which nodes attempt to transmit packets (cost), and their rates of success (throughput). In parts, we impose budget constraints that restrict the nodes' costs, such that the fraction of slots within which a node attempts transmissions is bounded. In practice, these additional constraints may be due to energy constraints, or a bandwidth constraint placed on the network application. This generalized version of slotted-Aloha is worth studying for two reasons. First, it is derived from a protocol that is commonly used today. Second, we will show that the generalized versions can outperform the original version, both in terms of aggregate throughput, as well as the ability to cope with malicious users.

The main contributions of this paper can be summarized as follows: 1) We formulate different user behaviors under a generalized slotted-Aloha protocol where users make decisions using a two-state system. 2) We identify throughput bounds for a system of cooperative users and explore the trade-off between user throughput and short-term fairness. 3) Under non-cooperative/selfish behaviors of the users, we identify a Prisoner's Dilemma phenomenon. 4) Under adversarial behavior of one user, we measure the maximum possible deterioration of the non-adversarial users' throughputs.

We organize our paper as follows. In Section 2, we review related work. In Section 3, we motivate the protocol and construct a Markov Model for the generalized slotted-Aloha protocol. In Section 4, we measure the sys-

tem throughput in a cooperative environment where users want to maximize the total throughput of the system. In Section 5 we evaluate both the aggregate and individual user throughputs where selfish users exist in the system and show the game as a Prisoner's Dilemma situation. In section 6, we explore a system in which attackers try to minimize the throughputs of the remaining nodes. Section 7 concludes.

2 Related Work

The Aloha protocol and its slotted version have been studied since the early seventies. Because of Aloha's inherent instability [7, 8, 9], Early research focused on stabilizing the Aloha protocols [9, 10]. In this work, we focus on the performance of stable slotted-Aloha type systems, where only a finite number of users will access the shared medium simultaneously. The justification of this assumption relies on the implementation of admission control procedures in today's networks. Early work on slotted Aloha with finite number of users can be found in [7].

Recent work using Game Theory to analyze users behaviors in MAC protocols and wireless ad-hoc networks can be found in [11, 12] and [13, 14] respectively. More specifically, game-theoretical analysis of the Aloha protocols can be found in [15, 16, 17, 18].

MacKenzie and Wicker's work [15, 16] discusses the stability of slotted-Aloha with selfish user behaviors and perfect information. Our work is different in way that we focus on performance (attainable throughput) instead of stability. In terms of data backlog at the users, we consider scenarios of *elastic transfers*, where users always have data to send and utilize whatever bandwidth is available, and hence classical stability results do not apply to our analysis.

Jin and Kesidis's work [17] discusses the equilibrium of a non-cooperative game for Aloha protocols. In their non-cooperative game formulation, each user only uses one transmitting probability (i.e., always in a backlogged state). Moreover, utility functions and payments are specified for each user. In our work, on the other hand, the formulation is for a generalized slotted-Aloha protocol which considers the Markovian decisions depending on whether the most recent transmission is a success (in a Free State) or a failure (in a Backlogged State). And we do not impose any payment on the users. Our settings capture more realistic features in real Aloha systems.

Altman et al. [18] consider slotted-Aloha systems as both cooperative and non-cooperative games with partial information. Their work assumes that there are a finite number of sources without buffer. The arrival packets to each source follows a Bernoulli process. As in typical slotted-Aloha, users only control the backlog probability in both games. In our work, we consider the saturated arrival when each user always has packets to transmit. But users' strate-

gies are more broad. Because users are also allowed to choose a non-zero probability to backoff even its previous transmission is a success. In addition, we analyze an adversarial game where an attacker who wants to minimize other users' throughput appears in the game.

3 Protocol Description and Model

In this section, we describe a generalized slotted-Aloha MAC protocol and construct a Markov Model from which its throughput can be measured. The slotted-Aloha protocol can be implemented as a 2-state system, where the state maintains the outcome of the previously attempted transmission. A node is in its *Free State* if the most recent transmission from that node is a success. Otherwise, the node is in its *Backlogged State*. In the Free State, a node transmits during the next slot with probability 1, and in the Backlogged State, it transmits during the next slot with probability p . Our generalization of the above protocol is to allow a node to vary the probability with which a node transmits a packet when it resides within the Free State.

Our evaluation will consider a network of N nodes, where N is often 2. We assume that nodes can coordinate slot transmission times and can estimate the number of nodes N with which they compete for bandwidth. However, because nodes' transmissions may interfere but cannot be deciphered, methods to prevent slot contention that require explicit communication and coordination among the competing members (e.g., TDMA, RTS-CTS) cannot be used.

Each node x can tune its protocol using two parameters p_1^x and p_2^x , which respectively are the transmitting probability in the Free State and the Backlogged State for node x . Given N and the transmitting probabilities for each of the nodes in each of the states, it is possible to compute the *throughput* T_x , which is the fraction of slots within which x successfully completes a transmission, and the *cost* C_x , which is the fraction of slots within which x attempts transmission.

Nodes may have physical limitations (e.g. power consumption constraints or application throughput constraints) that may bound its cost function. We bound allowed cost by a *budget*, B_x , such that a node's parameters must produce a cost $C_x \leq B_x$.

When we consider cooperating nodes that seek to maximize throughput, we are also interested in system *fairness*: all nodes should get an equal share of the throughput. In addition, we assume that it is undesirable for any one node to "capture" the medium for an extended number of slots - a long-term capture can be thought of as unfair over a short duration. Koksals work [19] gives an analysis of the short-term fairness of MAC protocols. It provides some insight into why MAC protocols exhibit bad short-term fairness using two different fairness indexes. In this paper, we measure short-term fairness via a more fundamental quantity defined

as follows:

Definition 1 Let D_x be the number of consecutive slots following an initially successful transmission over which node x successfully transmits packets (i.e., if there are k successful consecutive transmissions, then $D_x = k-1$). The system is said to be **M -short-term fair** to all nodes if $E[D_x] \leq M$ for all nodes x .

Each node's decision to transmit within a particular slot depends only on the outcome of its previous attempt (success or failure), and does not depend on the state of other nodes. Hence, this protocol is easily implemented in a distributed manner. Moreover, each node's decision is in fact Markovian, as it depends only on its previous attempt's outcome. For simplicity, we will assume that a node always has a packet to send in a slot whenever our slotted-Aloha variant decides to transmit a packet in a slot (i.e., nodes have a sufficient backlog of packets). However, our model also easily captures the case where a packet enters the queue to be transmitted with a fixed probability.

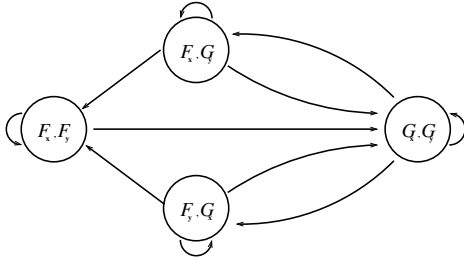


Figure 1. Two-node Markov Chain.

Figure 1 shows the state transition diagram for a two-node system with node x and y . F_x and G_x represent that node x is in a free state and a backlogged state respectively. A system for N nodes is easily modeled by as a Markov Model where the chain would consist of 2^N states. The transition matrix for the above two-node Markov Model is:

$$P = \begin{pmatrix} 1 - p_1^x p_1^y & 0 & 0 & p_1^x p_1^y \\ (1 - p_1^x) p_2^y & 1 - p_2^y & 0 & p_1^x p_2^y \\ (1 - p_1^x) p_2^x & 0 & 1 - p_2^x & p_1^x p_2^x \\ 0 & p_2^y (1 - p_2^y) & p_2^x (1 - p_2^x) & p_{44} \end{pmatrix}$$

where $p_{44} = p_2^x p_2^y + (1 - p_2^x)(1 - p_2^y)$.

If $p_1^x, p_1^y, p_2^x, p_2^y > 0$, the Markov Model is positive-recurrent. Let $\vec{\pi} = \{\pi_1, \pi_2, \pi_3, \pi_4\}$ to be the steady state distribution. The throughput and cost of node x are:

$$T_x = \pi_1(p_1^x)(1 - p_1^y) + \pi_2(p_1^x)(1 - p_2^y) + \pi_3(p_2^x)(1 - p_1^y) + \pi_4(p_2^x)(1 - p_2^y). \quad (1)$$

$$C_x = \pi_1(p_1^x) + \pi_2(p_1^x) + \pi_3(p_2^x) + \pi_4(p_2^x). \quad (2)$$

4 Cooperative Performance Analysis

In this section, we assume that nodes cooperate to fairly (i.e., equally) share the available bandwidth to maximize the aggregate system throughput. Clearly, if it were permissible to bias the allocation toward one of the nodes, the system could achieve full utilization by allowing only one node to transmit at all time. If a centralized scheduler or carrier sensing mechanism were permitted, we could also make fair share of the medium with 100% utilization. Here, we seek an unbiased and distributed solution for all nodes such that nodes will achieve the same performance on average.

Theorem 1 For two homogeneous nodes with $p_1^x = p_1^y = p_1$ and $p_2^x = p_2^y = p_2$, $\sup\{T_x + T_y\} = 2/3$.

All proofs of the theorems can be found in technical report [21]. Theorem 1 upper-bounds the maximum fair throughput at $2/3$, which is achieved in the limit as both nodes choose $\{p_1 = 1, p_2 \rightarrow 0\}$. This solution is intuitive: collisions are less likely to occur in a carrier-sense free environment when nodes are very unlikely to start trying to transmit, but hold the medium until a subsequent collision.

Theorem 2 For N homogeneous nodes with $p_1 = 1, p_2 \rightarrow 0$, the total throughput tends to $\frac{N}{2N-1}$.

Intuitively, when the number of nodes increases in the system, the system throughput decreases. However, Theorem 2 shows that the throughput does not drop to zero: even when the number of nodes tends to infinity, we can still achieve a total throughput of one half. Note that this result differs from the traditional performance bound $(1/e)$ of slotted-Aloha because our generalized model permits the capture of the resource. This allows a node to continually use the channel while all other nodes back off. An alternative analysis of this capture phenomenon can be found in [7]. Although the solution $\{p_1 = 1, p_2 \rightarrow 0\}$ maximizes throughput, it is not short-term fair. As $p_2 \rightarrow 0$, we have $E[D_x] \rightarrow \infty$. We next consider how to enforce short-term fairness:

Theorem 3 For N homogeneous nodes with $p_1 = 1$ and $p_2 \geq 1 - \sqrt[N-1]{1 - 1/M}$, the system is M -short-term fair.

Theorem 3 quantifies how to select p_2 to achieve a certain short-term fairness. In particular, in order to achieve M -short-term fairness, we can choose the following value of $p_2 = 1 - \sqrt[N-1]{1 - 1/M}$.

Figure 2 plots the total throughput under different short-term fairness constraints (M) as the number of nodes, N is varied along the x -axis. Without sacrificing much throughput, we can achieve very good short-term fairness. For example, if we want the system to be 8-short-term fair, we can achieve a total throughput close to $1/2$ even for large N . Actually, when $N \rightarrow \infty$, the total throughput does not collapse to zero.

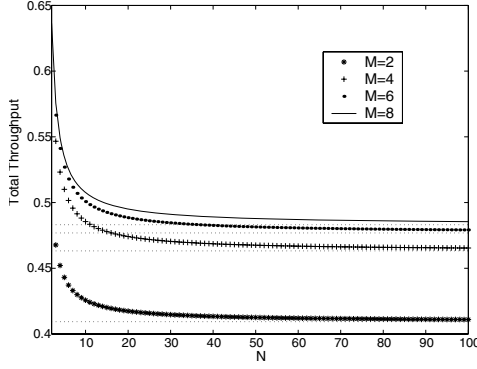


Figure 2. Throughput under different fairness conditions.

5 Competitive Performance Analysis

In the last section, we identified the lower bounds of the obtainable throughput among cooperating nodes, even taking into account short-term fairness requirements. In this section, we assume that each node is autonomous and sets its protocol parameters to greedily maximize its own throughput. First, let us see how a single node can increase its own throughput by deviating from the cooperative solution. After that, we formulate a constrained optimization problem for each node to maximize its throughput. We construct a *Stackelberg game*[3] which a pair of nodes can play. This game reveals that a *Prisoner's Dilemma*[3] phenomenon can occur.

5.1 Consequence of Selfish Behavior

Suppose N nodes are originally cooperative and use $p_1 = 1$ and $p_2 = 1 - \sqrt[N-1]{1 - 1/M}$ to achieve the maximum M -short-term fair aggregate throughput. In this system, each node x obtains throughput:

$$T_x = \rho/N = (M - 1)/[N(M - 1) + 1/p_2].$$

If one node deviates from this setting and sets $p_2 = 1$ instead, its throughput increases to

$$T'_x = p_b = (1 - p_2)^{N-1} = 1 - 1/M.$$

Its throughput now equals the probability that no other node is transmitting in each time-slot. Comparing the above two equalities, we have:

$$\frac{T'_x}{T_x} = \frac{N(M - 1) + 1/p_2}{M} = N + \frac{1 - Np_2}{Mp_2}$$

Hence, by unilaterally changing p_2 to be 1, a selfish node can usually increase its throughput at least N times (if $Np_2 < 1$). This change sacrifices the throughput of all of the other nodes, which no longer obtain any throughput.

5.2 Stackelberg Game

We have shown that a single selfish node can increase its own throughput in a setting where all other nodes are cooperative. We now explore what happens when multiple nodes set their parameters in a greedy fashion. Here, let us consider a network with two selfish nodes, x and y , each of which wants to maximize its own throughput. In addition, we assume that each node has respective budget constraints $C_x \leq B_x$ and $C_y \leq B_y$. C_x and C_y are the costs of both nodes as defined in Equation (2). $B_x, B_y \in (0, 1]$ are two budget constants which physically restrict the average number of packets the node can transmit in each time-slot. As mentioned earlier, this constraint is a simple way to model bounds on channel usage, which may exist, for instance, in order to preserve battery power. The competition between these two nodes is modeled as a *Stackelberg game*[3], in which a “leader” chooses a strategy (i.e. the transmitting probabilities in both the Free State and the Backlogged State) and then a “follower”, informed of the leader’s choice, chooses a strategy. We formulate a non-cooperative Stackelberg game as follows:

Players: The leader node x and the follower node y .

Strategy: $S^x = \{p_1^x, p_2^x\}$ for x ; $S^y = \{p_1^y, p_2^y\}$ for y .

Payoff: T_x and T_y for x and y respectively.

Game rule: x decides $\{p_1^x, p_2^x\}$ first. y decides $\{p_1^y, p_2^y\}$ after knowing $\{p_1^x, p_2^x\}$.

Follower’s Problem: The follower y is given the leader’s chosen parameters. It then simply sets its own parameters to maximize its own throughput. More formally, for any given \tilde{S}^x , the follower node y solves:

$$\begin{aligned} \hat{S}^y(\tilde{S}^x) &= \arg \max T_y(\tilde{S}^x, \hat{S}^y) \\ \text{Subject to: } &C_y(\tilde{S}^x, \hat{S}^y) \leq B_y. \end{aligned}$$

Leader’s Problem: The leader knows that the follower will choose its parameters to greedily maximize its own throughput. Therefore, the leader must choose its protocol parameters that will maximize its throughput, given the follower will subsequently choose its own parameters to maximize its throughput. More formally, the leader node x solves:

$$\begin{aligned} \hat{S}^x &= \arg \max T_x(\hat{S}^x, \hat{S}^y(\hat{S}^x)) \\ \text{Subject to: } &C_x(\hat{S}^x, \hat{S}^y(\hat{S}^x)) \leq B_x. \end{aligned}$$

In order to solve this Stackelberg game, we first solve the follower’s problem for every possible strategy taken by node x . Thus, we obtain the best response strategy of y as a function of node x ’s strategy. After that, the leader decides its optimal strategy according to node y ’s best response strategy. This procedure is often referred to as *backward induction* [20]. The resulting game solution is often referred to as a Stackelberg equilibrium.

5.3 Three Equilibrium Regions

We solve the above Stackelberg game for nodes who have the same budget constraints, which means $B_x = B_y$.

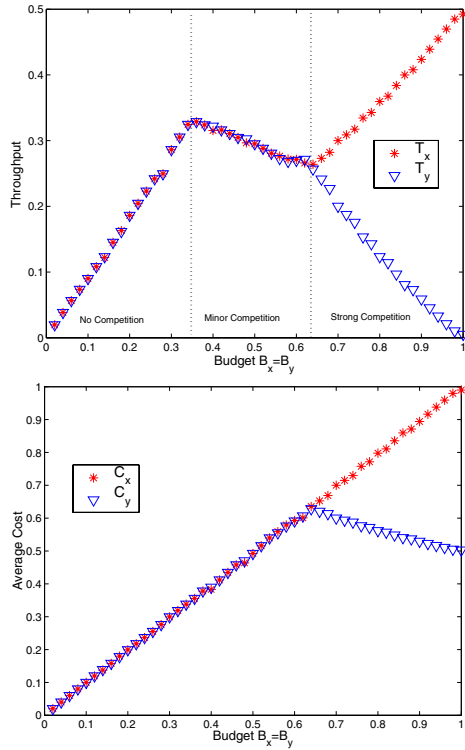


Figure 3. Throughput and cost in equilibrium.

Figure 3 shows the throughputs and costs of both nodes at the Stackelberg equilibrium. The x -axis indicates the budget constraint for both nodes. The change in the throughputs resulting from a slight increase in budget behaves differently in three different regions: 1) When the budget is less than $1/3$, both nodes achieve the same throughput. Here, they utilize their entire budgets, and their throughput is simply limited by the budget constraints. Hence, increasing the budget increases their achieved throughputs. 2) When the budget is between $1/3$ and $2/3$, both nodes again choose similar strategies and achieve similar throughputs. However, increasing both nodes' budgets decreases each node's throughput. In this region, the throughput is limited by not only the budget constraints, but also by the competition between these two nodes. Note that because similar strategies are chosen, it does not matter (to a node) whether it is chosen to be the leader or the follower. 3) When the budget is more than $2/3$, the leader can select parameters that give it a larger fraction of the throughput. The follower, still wishing to maximize its own throughput, actually becomes less aggressive. In this region, it is clearly preferable to be the leader.

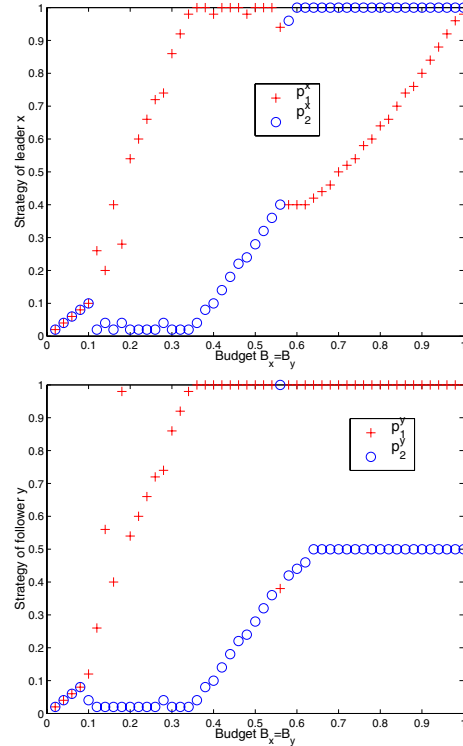


Figure 4. Strategies in equilibrium.

Figure 4 shows the strategies of both nodes at the Stackelberg equilibrium. In the first two solution regions, both nodes use similar strategies. When the budgets are close to $1/3$, the greedy strategies are similar to what would be selected by cooperative nodes, and the aggregate throughput approaches $2/3$. As the budgets are further increased, greedy strategies cause increased contention of the medium and the rate of interference becomes significant. When the budgets exceed $2/3$, the leader sets $p_2 = 1$, which means that if a transmission fails in a slot, it attempts retransmission during the next slot. This makes sense intuitively because the follower, attempting to maximize its own throughput with its confined budget, must back off with high probability after a collision, and the "safest" time for the follower to transmit will be following a previous successful transmission. Hence, the follower sets $p_1 = 1$, since it can only be in the Free State when the leader is also in the Free State, and it can only successfully transmit when the leader is in the Free State.

5.4 Prisoner's Dilemma

From the above Stackelberg game, the leader node x can achieve a throughput of at most $T'_x = 1/2$. Note that this throughput is better than it can gain in a cooperative environment ($T_x < 1/3$). We now assume that both nodes

decide their strategies simultaneously.

We consider three budget scenarios and the possible strategies that would be played in the Stackelberg game:

1) Low budget region: $B_x = B_y = 0.34$. Strategy $S_C = S^x = S^y = \{p_1 = 0.98, p_2 = 0.02\}$.

2) Medium budget region: $B_x = B_y = 0.5$. Strategy $S_M = S^x = S^y = \{p_1 = 1, p_2 = 0.28\}$.

3) High budget region: $B_x = B_y = 0.8$. Strategy $S_L = S^x = \{p_1 = 0.64, p_2 = 1\}$, and $S_F = S^y = \{p_1 = 1, p_2 = 0.5\}$.

Strategy S_C in the lower budget region is similar to the strategy played by nodes in a cooperative environment. Strategy S_M , which is more aggressive than S_C , is played by both the leader and the follower in the middle budget region. Finally, S_F and S_L are the respective strategies of the leader and the follower in the high budget region.

Now, let us consider two situations where two nodes share a common medium and must choose their parameters to maximize their individual throughputs without knowing what their opponent will do:

To Cooperate or to Compete? Consider a game in which two nodes, each with a budget of 0.5. Each node can either choose a strategy that is cooperative or greedy. The throughput of both nodes can be depicted by the following table.

	S_C	S_M
S_C	(0.3246,0.3246)	(0.0034,0.9288)
S_M	(0.9288,0.0034)	(0.2951,0.2951)

The most efficient solution is at (S_C, S_C) . However, a selfish node will note that whichever strategy its opponent chooses, its throughput will be increased by choosing S_M . Here, we see a typical Prisoner's Dilemma [3]. Although from a global perspective, both nodes know the best solution is (S_C, S_C) , from any hypothetical local point, strategy S_M should always be played. This is because, for any fixed strategy by the opponent, choosing S_M is always better than choosing S_C . Strategy S_M is called the *dominating strategy* [3] for both nodes and the solution (S_M, S_M) is the unique *Nash equilibrium* [3] of this game.

To Lead or to Follow? In the second game, we assume that nodes have budgets in the third region. As before, the nodes are better off playing a greedy strategy. However, now the nodes must also decide whether to choose the leader's strategy or the follower's strategy.

	S_F	S_L
S_F	(0.25,0.25)	(0.1233,0.3595)
S_L	(0.3595,0.1233)	(0,0)

Here, a node's strategy is not clear. A node is always better off choosing the opposing strategy of its competitor. Choosing the follower strategy is more conservative. A throughput of at least 0.1233 is ensured, but the throughput can be at most 0.25. If the leader strategy is chosen, a throughput of 0.3595 is possible, but a throughput of 0 is also a possible outcome. Interestingly, this game has two symmetric Nash

equilibrium solutions, which are (S_F, S_L) and (S_L, S_F) .

6 Adversarial Model Analysis

All previous scenarios assume that each node, whether cooperative or selfish, is interested in maximizing its own throughput. In this section, we consider an attacking node whose goal is to use its restricted budget to minimize the throughput of the other nodes in the system, i.e., to cause as many of its packets to collide with what would otherwise be successful transmissions. We first discuss how much damage is caused by a random (stateless) attack. Next, we formulate this attack model as another Stackelberg game.

6.1 Pure Random (Stateless) Attack

If an attacker is able to transmit a packet in every slot, it can clearly prevent any transmission from being successful. We assume that the adversary node has a budget $B \in (0, 1]$, allowing it to transmit in at most a fraction B of the slots.

Definition 2 *An adversary node uses a p -pure random attack if it transmits a packet in each time-slot independently with probability p .*

By Definition 2, an adversary node with a budget B can use a p -pure random attack for any $p \leq B$. We can imagine that p -pure random attack for a communication channel is identical to a lossy channel where a packet is lost with probability p .

Theorem 4 *Suppose there are two nodes x and y , where node x is an adversary node which uses p -pure random attack. Then, regardless of its strategy node y 's throughput, T_y , is equal to $(1 - p)C_y$.*

Theorem 4 formalizes the intuitive result that a p -pure random attack reduces the capacity by a multiplicative factor of $1 - p$ of the original capacity. Interestingly and countering our preliminary intuition, if we have more than one cooperative node, the damage caused by a p -pure random attack is often larger than a factor of $1 - p$:

Theorem 5 *Suppose originally N homogeneous nodes, which use $p_1 = 1$ and $p_2 < 1/N$ in the system, achieve an aggregate throughput ρ . If an adversary node joins the system and uses a p -pure random attack, then the aggregate throughput of the N cooperative node is less than $(1 - p)\rho$.*

An explanation of this result is as follows: as more nodes participate in the cooperative process, the expected number of slots between transmissions in the Backlogged State grows at a faster rate than the expected number of slots between transmissions in the Free State. A random seeding of losses forces more nodes to spend more time in the

Backlogged State, and as a result, each node attempts fewer transmissions over time, yet still loses a fraction p of the attempts to the random loss process.

6.2 Adversarial Stackelberg Game

Now, let us compute the reduction in throughput that an attacking node can cause if it maximizes its attack power under a 2-state system. As in section 5.2, we introduce a Stackelberg game in this section. The difference between the previous model and this model is that we assume the leader node x is the attacker and its sole objective is to minimize the throughput of node y . Because the leader always has the advantage over the follower, making the attacking node the leader maximizes its potential for damage. We still assume that node x and y have budget constraints: $C_x \leq B_x$ and $C_y \leq B_y$ respectively. The adversarial Stackelberg game is formally described as follows:

- Players:** The leader node x and the follower node y .
Strategy: $S^x = \{p_1^x, p_2^x\}$ for x ; $S^y = \{p_1^y, p_2^y\}$ for y .
Payoff: $-T_y$ and T_y for x and y respectively.
Game rule: x decides $\{p_1^x, p_2^x\}$ first. y decides $\{p_1^y, p_2^y\}$ after knowing $\{p_1^x, p_2^x\}$.

Follower's Problem: Given \tilde{S}^x , the follower node y solves:

$$\hat{S}^y(\tilde{S}^x) = \arg \max T_y(\tilde{S}^x, \hat{S}^y)$$

Subject to : $C_y(\tilde{S}^x, \hat{S}^y) \leq B_y$.

Leader's Problem: The leader node x solves:

$$\hat{S}^x = \arg \min T_y(\hat{S}^x, \hat{S}^y(\hat{S}^x))$$

Subject to : $C_x(\hat{S}^x, \hat{S}^y(\hat{S}^x)) \leq B_x$.

6.3 Two Equilibrium Regions

By backward induction, we solve the above adversarial Stackelberg game for nodes who have the same budget constraints, i.e., $B_x = B_y$. In the upper part of Figure 5, we plot the throughput of the follower (non-attacking) node y when x chooses the optimal 2-state attacking strategy. We also plot the curve $B_y(1 - B_y)$, which gives the throughput of node y when the attacker uses a p -pure random attack with $p = B_y$. In the lower part of Figure 5, we show the costs incurred by both nodes. We identify two regions in the Stackelberg equilibrium solutions: 1) When the budget is less than $2/3$, both nodes use up their budgets. The throughput of node y when attacked by the optimal 2-state attacker is identical to its throughput when attacked by a p -pure random attacker. 2) When the budget is larger than $2/3$, node y 's throughput is slightly but observably lower when attacked by the optimal 2-state attacker than when attacked by the p -pure attacker.

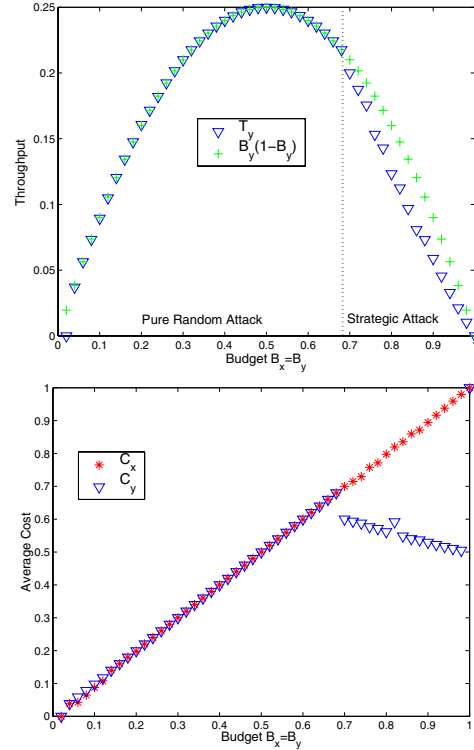


Figure 5. Throughput and cost in (adversary) Stackelberg equilibrium.

Intuitively, the attacking node will always use its entire budget to attack. But surprisingly, a strategic, 2-state attack cannot do better than pure random attack if the adversary node does not have a budget larger than $2/3$. When the budget is larger than $2/3$, the 2-state attack is only slightly more effective.

6.4 Random Attack Vs. Strategic Attack

We plot the strategies (p_1 and p_2 values) of both nodes in Figure 6. We find that the strategies played in the two budget regions are quite different. Not surprisingly, when the budget is less than $2/3$, the attacking node uses the pure random strategy $p_1^x = p_2^x = B_x$. Theorem 4 explains why the throughput T_y is so close to curve $B_y(1 - B_y)$ in the lower budget region. Our analysis reveals that y has multiple strategies to maximize its throughput. But all these strategies use up the budget B_y . While the set of strategies that can be played by node y seem to optimize its throughput against a 2-state adversary seem somewhat arbitrary, its resulting throughput is surprisingly always close to $B_y(1 - B_y)$.

After comparing the strategies played by both nodes in the larger budget region with those used by two non-cooperative, non-attacking nodes in Figure 4, we notice that

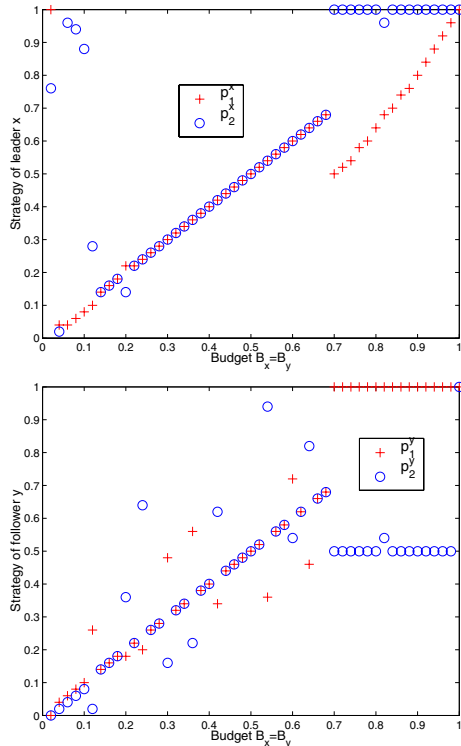


Figure 6. Strategies in (adversary) Stackelberg equilibrium.

they are strikingly similar. This means that an attacking node x chooses a strategy very similar to what is chosen by a node who wishes to greedily maximize its own throughput. Consequently, node y uses the same response strategy.

In conclusion, if bandwidth requirements/capabilities are low, an attacker cannot do much better than attacking at random points in time. If the bandwidth requirements and capabilities are high, then an attacker behaves similarly to a node seeking to greedily maximize its own throughput.

7 Conclusion

In this paper, we generalize the slotted-Aloha protocol to a general two-state process. We constructed a Markov model for this generalized two-state protocol. Our results showed that if all nodes cooperating in an effort to maximize the aggregate throughput can achieve an aggregate of at least $1/2$. On the other hand, if all nodes are selfish and greedily attempt to maximize their own individual throughputs, a situation similar to the traditional Prisoner's Dilemma arises. Finally, we showed that attacking nodes with limited budgets can do little better than a random attack, and nodes with large budgets should behave like their greedy counterparts.

References

- [1] N. Abramson, "The Aloha system – another alternative for computer communications," *AFIPS Conference Proceedings*, vol. 36, 1970, pp.295-298.
- [2] L. Roberts, "Aloha packet system with and without slots and capture," *ACM SIGCOMM Computer Communication Review*, Vol. 5, Issue 2, 1975.
- [3] M. J. Osborne and A. Rubinstein, *A course in game theory*. The MIT Press Cambridge, Massachusetts, 1994.
- [4] "Ethernet, a local area network: Data Link Layer and Physical Layer Specifications (v1.0)," *Digital Equipment Corporation, Intel, Xerox*.
- [5] R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed packet switching for local computer networks," *Communication of the ACM* 19(7), p. 395-404, July, 1976.
- [6] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: Part I-carrier sense multiple-access modes and their throughput-delay characteristics," *TRANSCOM COM-23(12)*, p.1400-1416, 1975.
- [7] R. Rom and M. Sidi, *Multiple access protocols: performance and analysis*. Springer-Verlag New York, Inc., 1990.
- [8] D. Bertsekas and R. Gallager, *Data Networks*. Prentice-Hall, Englewood Cliffs, New Jersey, 1992.
- [9] S. Ghez, S. Verd, and S. Schwartz, "Stability properties of slotted-aloha with multipacket reception capability," *IEEE Trans. Automatic Control*, vol AC-33, no. 7, p. 640-649, July, 1988.
- [10] R. Rivest, "Network control by Bayesian broadcast," *IEEE Trans. on Information Theory*, vol 33, no. 3, p. 323-340, 1987.
- [11] G. Tan and J. Gutttag, "The 802.11 MAC Protocol Leads to Inefficient Equilibria," in *IEEE INFOCOM*, Miami, FL, March 2005.
- [12] M. Cagalj, S. Ganeriwal, I. Aad, and J.-P. Hubaux, "On selfish behavior in CSMA/CA networks," in *IEEE INFOCOM*, March 2005.
- [13] Z. Fang and B. Bensaou, "Fair bandwidth sharing algorithms based on game theory frameworks for wireless ad-hoc networks," in *Proceedings of the IEEE Infocom*, 2004.
- [14] V. Srinivasan, P. Nuggehalli, C. Chiasserini, and R. Rao, "Cooperation in wireless ad hoc networks," in *Proceedings of IEEE Infocom*, 2003.
- [15] A. MacKenzie and S. Wicker, "Selfish users in Aloha: a game theoretic approach," *Proceedings of the Fall 2001 IEEE Vehicular Technology Conference*, 2001.
- [16] —, "Stability of multipacket slotted aloha with selfish users and perfect information," *Proceedings of IEEE INFOCOM*, 2003.
- [17] Y. Jin and G. Kesidis, "Equilibria of a noncooperative game for heterogeneous users of an Aloha network," *IEEE Communication Letters* 6 (7) p.282-284, 2002.
- [18] E. Altman, R. E. Azouzi, and T. Jimnez, "Slotted Aloha as a game with partial information," *The International Journal of Computer and Telecommunications Networking Vol. 45 Issue 6*, 2004.
- [19] C. E. Koksall, H. Kassab, and H. Balakrishnan, "An analysis of short-term fairness in wireless media access protocols (poster session)," *Proceedings of the ACM SIGMETRICS*, 2000.
- [20] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic theory*. New York : Oxford University Press, 1995.
- [21] R. T. B. Ma, V. Misra, and D. Rubenstein, "Generalized slotted-Aloha in Cooperative, Competitive and Adversarial Environments," *Columbia University Electrical Engineering*, February, 2006.