Contention Resolution under Selfishness

G. Christodoulou* K. Ligett† E. Pyrga‡

December 9, 2010

Abstract

In many communications settings, such as wired and wireless local-area networks, when multiple users attempt to access a communication channel at the same time, a conflict results and none of the communications are successful. Contention resolution is the study of distributed transmission and retransmission protocols designed to maximize notions of utility such as channel utilization in the face of blocking communications.

An additional issue to be considered in the design of such protocols is that selfish users may have incentive to deviate from the prescribed behavior, if another transmission strategy increases their utility. The work of Fiat et al. [8] addresses this issue by constructing an asymptotically optimal incentive-compatible protocol. However, their protocol assumes the cost of any single transmission is zero, and the protocol completely collapses under non-zero transmission costs.

In this paper, we treat the case of non-zero transmission cost c. We present asymptotically optimal contention resolution protocols that are robust to selfish users, in two different channel feedback models. Our main result is in the Collision Multiplicity Feedback model, where after each time slot, the number of attempted transmissions is returned as feedback to the users. In this setting, we give a protocol that has expected cost $2n + c\log n$ and is in $o(1)$-equilibrium, where n is the number of users.

1 Introduction

Consider a set of sources, each with a data packet to be transmitted on a shared channel. The channel is time-slotted; that is, the transmissions are synchronized

*Saarland University, Germany. gchristo@mpi-inf.mpg.de Partially supported by DFG grant Kr 2332/1-3 within Emmy Noether Program and by EPSRC grant EP/F069502/1.

†Cornell University, Ithaca, NY katrina@cs.cornell.edu Supported in part by an NSF Graduate Research Fellowship, NSF grant 0937060 to the Computing Research Association for the CIFellows Project, and NSF grant 1004416.

‡Technische Universität München, Germany. pyrga@in.tum.de

and can only take place at discrete steps. The packets are of fixed length and each fits within one time slot. If only one user transmits in a given slot, the transmission is successful. If more than one user attempts to transmit messages in the same slot, a collision occurs, and all transmissions for that time step fail; the failed packets will need to be retransmitted later. Typically, the goal of the system designer is to optimize global notions of performance such as channel utilization or average throughput. If all of the sources were under centralized control, avoiding collisions would be simple: Simply allow one source to transmit at each time step, alternating in a round-robin or other "fair" fashion.

What happens, though, if the transmission protocol must be executed in a distributed fashion, with minimal additional communication? Further, what happens if each source selfishly wishes to minimize the expected time before she transmits successfully, and will only obey a protocol if it is in her best interest? Can we still design good protocols in this distributed, adversarial setting? What information does each user need to receive on each time step in order for such protocols to function efficiently?

In this game theoretic contention resolution framework, Fiat et al. [8] design an incentive-compatible transmission protocol which guarantees that (w.h.p.) all players will transmit successfully in time linear in the total number of users. One of the nice features of their protocol is that it only needs a very simple feedback structure to work: They assume only that each player receives feedback of the form $0/1/2^+$ after each time step (*ternary* feedback), indicating whether zero, one, or more than one transmission was attempted. This positive result is actually based on a very negative observation, that the price of anarchy [15] in this model is unbounded: If transmission costs are assumed to be zero, it is an equilibrium strategy for all users to transmit on all time steps! Clearly, this is an undesirable equilibrium. Fiat et al. [8] construct their efficient equilibrium using this bad equilibrium quite cleverly as a "threat"— the players agree that if not all players have exited the game by a certain time, they will default to the always-transmit threat strategy for a very large time interval. This harsh penalty then incentivizes good pre-deadline behavior.

It is natural, however, to assume that players incur some transmission cost $c > 0$ (attributable to energy consumption; see, for example, [17, 24]) every time they attempt a transmission, in addition to the costs they incur due to the total length of the protocol. If the cost of transmitting is even infinitesimally non-zero, though, the threat strategy used in [8] is no longer at equilibrium, and the protocol breaks down. We address this here, by developing efficient contention resolution protocols that are ε-incentive-compatible even under *strictly positive transmission costs*.

In this work, we consider the model of Collision Multiplicity Feedback (CMF). After each collision, while the identity of the collided packets is lost, the *number* of the packets that participated in the collision is broadcast to all users. This information can be estimated (with small error) by a number of energy detectors. The CMF model has gained substantial attention in the literature because it admits significantly better throughput results than can be achieved with ternary feedback. In this setting, we present an efficient contention resolution protocol that is robust to selfish players. We also propose a contention resolution protocol for a *perfect information* feedback model, where all players can identify *which* players transmitted.

1.1 Our Results

There are three main technical challenges we address:

1. Separating players by developing unique ids. Our protocols differ from previous incentive-compatible contention resolution protocols in that they are *history dependent*; that is, the protocol uses the transmission history in order to construct a random ordering of the users. We will use algorithmic techniques similar to those for "tree"-style protocols, originally developed by Capetanakis [7, 6] and later further developed in the information theory and networks literatures.

2. Developing a "threat". Like Fiat et al. [8], we use a threat strategy to incentivize good behavior. However, the threat they use collapses under nonzero transmission costs[1]. On the other hand, we use a weaker threat strategy and need to develop efficient protocols that are incentive compatible despite our weaker punishment.

3. Detecting deviations. In order for a threat to be effective, the players need to be able to detect and punish when one of them cheats. Our protocol employs a system of checksums and transmissions whose purpose is to communicate information on cheating—despite the fact that a "transmission" gives no other information to the players than its presence or absence.

We address these challenges in two different feedback models, described below. In each feedback model we consider, we present asymptotically optimal protocols that guarantee to each user expected average cost that is linear in the total number of users. Our protocols are in ε-equilibrium, where ε goes to 0 as the number of players grows. This form of approximate equilibrium is a very reasonable

[1]Remember from the earlier description that the threat of Fiat et al. requires that all remaining players will be transmitting in every step with probability 1 for a large number of steps. This is an equilibrium only when the cost of a transmission is zero.

equilibrium notion in our setting, as games with a large number of players are technically the only interesting case—if there were only a constant number of players, one could simply play the exact time-independent equilibrium that appears in [8], although this has exponential cost.

Perfect Information First, in Section 3, we consider the feedback model of *perfect information*, where each user finds out after each time step what subset of sources attempted a transmission. Perfect information is a very common assumption in non-cooperative game theory. Making this strong assumption allows us to highlight the game theoretic aspects of the problem, and the insight we develop in the context of this strong feedback model is useful in developing algorithms under more restrictive feedback.

Unlike the trivial protocol under global ids, the randomized protocol that we present is *fair* in the sense that all users have the same expected cost, regardless of how each individual chooses to label them with local ids. Protocol PERFECT has expected cost $n/2 + \log n$ and is a $o(1)$-equilibrium w.r.t. the number of players n.

Collision Multiplicity Next, in Section 4, we present the main result of this work. Here, we study a model with Collision Multiplicity Feedback (CMF), (otherwise known as M-ary feedback; see e.g. [25, 20, 23]), in which after each time slot, users find out the exact number of packets which were transmitted during the slot, but not the identities of the transmitting players. In some practical situations, this feedback can be measured as the total energy on the channel during one slot, by means of a number of energy detectors. Our protocol MULTIPLICITY has expected cost $2n + c\log n$, where c is the transmission cost, and is a $o(1)$-equilibrium.

1.2 Related Work

Contention resolution for communication networks is a well-studied problem. The ALOHA protocol [1], given by Abramson in 1970 (and modified by Roberts [22] to its slotted version), is one of the most famous multiaccess communication protocols. However, Aloha leads to poor channel utilization due to an unnecessarily large number of collisions. Many subsequent papers study the efficiency of multiaccess protocols when packets are generated by some stochastic process (see for example [12, 11, 21]). Such statistical arrival models are very useful, but cannot capture worst-case scenarios of bursty inputs, as in [5], where batched arrivals are modeled by all n packets arriving at the same time. To model this worst-case scenario, one needs n nodes, each of which must simultaneously transmit a packet; this is also the model we use in this work.

One class of contention resolution protocols explicitly deals with conflict resolution; that is, if $k \geq 2$ users collide (out of a total of n users), then a resolution algorithm is called on to resolve this conflict (it makes sure that all the packets that collided are successfully transmitted), before any other source is allowed to use the channel. For example, [7, 6, 14, 26] describe *tree algorithms* whose main idea is to iteratively give priority to smaller groups, until all conflicts are resolved, with $\Theta(k + k\log(n/k))$ makespan. We use a similar splitting technique in the algorithms we present here.

A variety of upper and lower bounds for the efficiency of various protocols have been shown. For the binary model (transmitting players learn whether they succeed or fail; non-transmitting players receive no feedback) when k is known, [10] provides an $O(k + \log k \log n)$ algorithm, while [18] provides a matching lower bound. For the ternary model, [13] provides a bound of $\Omega(k(\log n / \log k))$ for all deterministic algorithms. In all of the results mentioned so far in this subsection, it is assumed that players will always follow the protocol given to them, even if it is not in their own best interest.

The CMF model we consider in this paper was first considered by Tsybakov [25], where he proposed a protocol with throughput 0.533. Later Pippenger [20] showed, using a random-coding existential argument, that the capacity of the channel is 1. Ruszinkó and Vanroose later in [23] gave a constructive proof of the same result, by designing a particular protocol reaching the throughput 1. Georgiadis and Papantoni-Kazakos [9] considered the case when the collision multiplicity can be estimated by energy detectors, up to an upper bound.

More recently, a variety of *game theoretic* models of slotted Aloha have also been proposed and studied in an attempt to understand selfish users; see for example [2, 16, 3]; also [17, 2] for models that include non-zero transmission costs. Much of the prior game theoretic work only considers transmission protocols that always transmit with the same fixed probability (a function of the number of players in the game). By contrast, we consider more complex protocols, where a player's transmission probability is allowed to be an arbitrary function of her play history and the sequence of feedback she has received. Other game theoretic approaches have considered pricing schemes [27] and cases in which the channel quality changes with time and players must choose their transmission levels accordingly [19, 28], and [4] for a related model.

As discussed above, Fiat et al. [8] study a model very similar to the one we present here; while the feedback mechanism they assume is not as rich as ours, crucially, their model does not incorporate transmission costs. The idea of a threat is used both in [8] and in our work, as a way to incentivize the players to be obedient. However, the technical issues involved are completely different in the two papers. [8] uses a threat in a form of a *deadline*, while ours use a *cheat detection* mechanism to identify the fact that someone deviated from the protocol. The protocol in Fiat et al. [8] relies for its threat on the existence of an extremely inefficient equilibrium, where all players constantly transmit their packets for an exponentially long period. This equilibrium is used as an artificial deadline, that is, the protocol switches to that inefficient equilibrium after linear time. This threat is history independent, in the sense that it will take place after a linear number of time steps, regardless of any transmission history of the players. This history-independent threat relies critically on the absence of transmission costs. On the other hand, our threat is history dependent and makes use of a cheat-detection mechanism: any deviation from the protocol is identified with at least constant probability, and the existence of a deviation is communicated to all players. In response, all players switch their transmission strategy according to the exponential time-independent equilibrium (all players transmit at every slot with probability $\Theta(1/\sqrt{n})$).

The communication allowed by the feedback models we employ is critical in allowing us to perform cheat detection and in allowing us to communicate the presence of a cheater, so that she can be punished. Although we are not aware of lower bound results that explicitly rule this out, we suspect that the ternary feedback mechanism used by Fiat et al. [8] is not rich enough to allow such communication.

2 Definitions

Game structure Let $N = \{1, 2, \ldots, n\}$ be the set of players in the game. Every player carries a single packet of information that she wants to send through a common channel, and all players (and all packets) are present at the start of the protocol. We assume that time t is discretized, divided into slots $t = 1, 2, \ldots$. At any given slot t, a pending player i has two available pure strategies: She can either try to transmit her packet in that slot or stay quiet. We represent the action of a player i by an indicator variable X_{it} that takes the value 1 if player i transmitted at time t, and 0 otherwise. The transmission vector X_t at time t is represented by $X_t = (X_{1t}, X_{2t}, \ldots, X_{nt})$, while the number of attempted transmissions at time t is denoted by $Y_t = \sum_{i=1}^{n} X_{it}$. The transmission sequence X^t is the sequence of transmission vectors of time up to t: $X^t = (X_1, X_2, \ldots, X_t)$. In a (mixed) strategy p_i, a player transmits at time t with probability $p_{it} = Pr[X_{it} = 1]$. If exactly one player transmits in a given slot, we say that the transmission is *successful*, and the player who transmitted leaves the game;[2] the game then continues with the rest of the players. If two or more players attempt to transmit at the same slot, then they all fail and remain in the game. The game ends when all players have successfully transmitted.

[2] Alternatively, we can assume the player transmits with $p_{it} = 0$ on all subsequent rounds.

Feedback At every time step t, the actual transmission vector is X_t. However, individual players do not have access to this complete information. Instead, after playing at step t, each player i receives some feedback I_{it} that is a function of X_t. The feedback is symmetric, i.e., every player who receives feedback gets the same information, i.e., $I_{jt} = I_{kt} = I_t$. At any time step t, a player's selected action is a (randomized) function of the entire sequence of actions she has taken so far and the *history* $H_{it} = (I_{i1}, \dots, I_{i(t-1)})$ of the feedback that she has received from the channel. For convenience, we define a player i's *personal history* $h_{it} = (X_{i1}, X_{i2}, \dots, X_{it})$ as the sequence of actions she took.

We distinguish between two different feedback structures: (1) *Perfect information:* After each time step t, all players receive as feedback the identities of the players who attempted a transmission, i.e., $I_{it} = X_t, \forall i \in N$. However, there are not shared global ids for the players; each player potentially has a different local ordering on the player set, so, for example, the concept of the "lexicographically first player" has no common meaning. (2) *Collision Multiplicity Feedback (CMF):* After each time step t, all players receive as feedback the cardinality of the set of players who attempted a transmission, but do not find out their identities. Here, $I_{it} = Y_t = \sum_{j=1}^{n} X_{jt}, \forall i \in N$.

Transmission protocols We define f_{it}, a *decision rule* for player i at time t, as a function that maps a pair $(h_{i(t-1)}, H_{i(t-1)})$ to a probability p_{it}. A *protocol* f_i for player i is simply a sequence of decision rules $f_i = f_{i1}, f_{i2}, \cdots$. A protocol is *symmetric* or *anonymous* and is denoted by $f = f_1 = \ldots = f_n$ iff the decision rule assigns the same probabilities to all players with the same personal history. In other words, if $h_{it} = h_{jt}$ for two players $i \neq j$, it holds that $f_{i,t+1}(h_{it}, H_{it}) = p_{i(t+1)} = p_{j(t+1)} = f_{j,t+1}(h_{jt}, H_{jt})$.[3]

Individual utility Given a transmission sequence X^T wherein all players eventually transmit successfully, define the *latency* or *success time* S_i of agent i as $\arg\min_t (X_{it} = 1, X_{jt} = 0,\ \forall j \neq i)$. The cost to player i is made up of costs for the time-to-success and transmission costs: $C_i(X^T) = S_i + c \sum_{t \le S_i} X_{it}$. Given a transmission sequence X^T of actions so far, a decision rule f induces a probability distribution over sequences of further transmissions. In that case, we write $E_i^f(X^T)$ for the expected total cost incurred by a sequence of transmissions that starts with X^T and then continues based on f. In particular, for $X^0 := \emptyset$, $E_i^f(X^0)$ is the expected cost of the sequence induced by f.

[3] Notice that since the feedback is symmetric, $h_{it} = h_{jt}$ implies $H_{it} = H_{jt}$.

Equilibria The objective of every player is to minimize her expected cost. We say that a protocol f is in *equilibrium* if for any transmission sequence X^t the players cannot decrease their cost by unilaterally deviating; that is, for all players i, $E_i^f(X^t) \leq E_i^{(f'_i, f_{-i})}(X^t)$, for all f'_i, t. Similarly, we say that a protocol f is in an ε-*equilibrium* if for any transmission history X^t $E_i^f(X^t) \leq (1+\varepsilon)E_i^{(f'_i, f_{-i})}(X^t)$, for all f'_i, t.

Social utility We are interested in providing the players with a symmetric protocol that is in equilibrium (or ε-equilibrium, for some small $\varepsilon > 0$) such that the average expected cost[4] $E_i^f(X^0)$ of any player i is low.

3 Perfect Information

In this section we consider the perfect information setting, where in every time slot t, every user i receives feedback $I_{it} = X_t$, the exact vector of transmissions in t. It is important to note that although in this setting each player can distinguish between her opponents, we do not assume that the players have globally visible id numbers.[5] Instead, we assume that each player has a set of local id numbers to distinguish among her opponents, but one player's local labeling of all other players may be independent of another's. Unlike global ids, local ids do not pose very hard implementation constraints; for instance, in sensor networks or ad-hoc mobile wireless networks, users may be able to derive local ids for the other players based on relative topological position.

The main idea of the protocol we present here is to generate unique, random ids for each player based on her transmission history. Next, with the use of those random ids, the players are synchronized in a fair manner, and can exit the game within short time.

As mentioned earlier, this section is presented with transmission costs c as some negligible, nonzero ε, rather than treating general nonzero costs. This greatly simplifies the presentation of this section and allows us to focus on the main ideas that the more complicated protocol of Section 4 is based on. No such assumption on c will be made when describing the protocol for the more restricted (and thus more challenging) feedback model. Further, in Section 4.2 we discuss how non negligible transmission costs can be incorporated into this protocol as well.

[4]Since we are interested in symmetric protocols, then the average expected cost is equal to the expected cost of any player, and hence the social utility coincides with the individual utility.

[5]In fact, if the identities of the players were common knowledge, then simply transmitting in lexicographic order would be incentive-compatible and achievable.

3.1 Protocol PERFECT

Protocol PERFECTworks in rounds. In each round k there is exactly one *split* slot and some $\ell_k \geq 0$ *leave* slots. In a split slot, all pending players (i.e., the players that are still in the game) transmit with probability $1/2$, independently of each other and their personal history. We define the id $id_i(k)$ of player i, after the kth split slot has taken place, as a binary string of length k that represents the transmission history of player i only on the split slots. When, after some round k, a player i manages to obtain a unique id, i.e., $id_i(k) \neq id_j(k)$, for all $i \neq j$, she will be assigned a leave slot. During a leave slot a single prescribed player transmits with probability 1, while all other players stay quiet. Such a player has a successful transmission and exits the game.

The protocol is given as Protocol 3.1. We will now describe what happens in a round in more detail. Consider the beginning of round $k+1$, for $k \geq 0$. The first slot s is a split slot. Let n_T be the total number of players that transmitted in s. Every player observes the actions of all other players in slot s, and updates their ids: The id of player j is updated by appending "1" or "0" at the end of $id_j(k)$, depending on whether j transmitted in s or not. If there are players that obtained unique ids within slot s then they must be assigned leave slots. Let U_{k+1} be the set containing those players. The order in which those players will be assigned leave slots depends on the number n_T of players that transmitted in s: The players in U_{k+1} are assigned leave slots in order of increasing id if n_T is even, and in order of decreasing id otherwise. All players in U_{k+1} will transmit successfully and exit the game.[6]

[6]A player i who is currently assigned a leave slot, keeps transmitting until she succeeds. This technical detail ensures that the protocol will not collapse if some other player j tries to transmit at that slot. This might happen only in the case that j deviates from the protocol and transmits in the leave slot assigned to i.

Protocol 3.1 PERFECT

Require: Code for player i, at the beginning of round $k+1$:

N' : set of all players still in the game (pending players)

$id_j(k)$: id of player $j \in N'$ w.r.t. the first k split slots

Transmit in current *split* slot s with probability $1/2$

$n_T :=$ #players that transmitted in slot s

for $j \in N'$ **do**

  $X_{j(k+1)} := 1$ if j transmitted in s, 0 otherwise

  $id_j := id_j(k) \cdot X_{j(k+1)}$ {Update the id's of all players, appending $X_{j(k+1)}$ (denoted by "$\cdot$") }

$U_{k+1} := \{j \in N' : id_j \neq id_k,$ for all $k \in N', k \neq j\}$

if $|U_{k+1}| > 0$ **then** {There are players that obtained a unique id with the last split}

  if n_T is even **then** {Sort U_{k+1} in increasing or decreasing order according to the value of n_T}

    sort U_{k+1}, in *increasing* order of id_j

  else

    sort U_{k+1}, in *decreasing* order of id_j

  for $j \in U_{k+1}$ **do** {Players in U_{k+1} are assigned *leave* slots}

    if $i = j$ **then** {i is the player that is assigned the current leave slot}

      Transmit in current slot s_i with probability 1

    else {another player is assigned the current leave slot}

      Stay quiet in current slot s_j with probability 1

[End of round $k+1$]

The order in which the players in U_{k+1} are assigned leave slots is not fixed, so as to avoid giving incentive to a player to strategically create an id, instead of obtaining a random one. If the players in U_{k+1} were always assigned leave slots in order of increasing id, then players would have incentive to remain quiet in split slots, as this would result in an id of smaller value (they append a "0" to their id) than if they transmitted (they would append a "1" to their id). To avoid this, the protocol prescribes the order to be increasing or decreasing with equal probability (since n_T is equally likely to be even or odd); this way the players have nothing to gain by choosing any specific id.

Fixing a player i, any player j (including i) is equally likely to obtain a unique id in any round k, regardless of whether i transmits or stays quiet during the split slot of round k. Therefore, i cannot reduce the expected number of rounds she needs until she obtains a unique id, nor the expected number of players that will be assigned leave slots before she does. Also, due to the assumption of arbitrarily small transmission cost c, player i has no incentive to deterministically stay quiet during a split so as to save on the transmission costs she incurs. Therefore, if no player could succeed during a split slot (i.e, have a successful transmission due to being the single player transmitting in a split), then the expected cost of i would have been exactly the same, whether i transmits or stays quiet in a split. (Player i has no reason to deviate from the protocol in a leave round, as this can only increase her total cost.) The possibility of players succeeding during a split, creates an imbalance between the expected costs a player has per step when she transmits and when she stays quiet during the split. However, the following Theorem suggests that the maximum gain over all possible deviations is very small.

Theorem 3.1. *Protocol* PERFECT *is a $o(1)$-equilibrium. Moreover, the expected total cost of a player i is $E_i(X^0) = n/2 + \log n$.*

Regarding the expected cost, note that player i is expected to obtain a unique id in $\log n$ rounds and the players are equally likely to be assigned the k-th leave slot, for all $1 \leq k \leq n$. The next section is devoted to the proof of Theorem 3.1.

3.2 Analysis of Protocol PERFECT

In order to prove Theorem 3.1 we will need to bound the ratio of the minimum expected cost a player can achieve by any sequence of deviations over the cost she incurs if she follows the protocol. First, Lemma 3.2 will bound the maximum gain a player can have by deviating from the protocol in a single step. More precisely, it is shown that staying quiet during a split slot is a better strategy than transmitting, for any pending number of players $n' \geq 2$. The idea behind this lies in the way that the protocol is constructed: if it were not for the possibility of players exiting during a split slot, the expected cost of any player i would be exactly the same, regardless of whether i transmits in split slots, or stays quiet. In other words, we would have had an exact equilibrium. However, for each split slot t, there is some positive (but small) probability that among the n' pending players, only a single player transmits in t. That player not only obtains a unique id in t, but also has a successful transmission during the split. When this is the case, the rest of the pending players need not wait for a leave slot to be assigned to her, as she already exits the game during the split. We can show that it is actually beneficial for a player to stay quiet, hoping that someone else will succeed during the split, instead of transmitting in the hope that she will be the unique player doing so (see Lemma 3.2 for more details). As a result, a player minimizes her expected cost if she stays quiet with probability 1 in *all* split slots and follows the protocol during leave slots (Corollary 3.3). This is what the proof of Theorem 3.1 is based on.

The next lemma shows that the optimal strategy for any player is to always stay quiet in split slots.

Lemma 3.2. *Let t be some split slot, in the beginning of which there are $n' > 2$ pending players and let i be one of them. Consider the expected future costs (starting from the beginning of t) E^T, E^Q of i if she transmits, or if she stays quiet in t with probability 1, respectively, and follows the protocol in all subsequent slots. Then,*

$$E^T - \frac{n'-1}{2^{n'-1}} < E^Q < E^T.$$

Proof. Let N' be the set of pending players at the beginning of t, with $|N'| = n'$. Consider some player $i \in N'$, and let G_i be the group of i at the beginning of slot t (i.e., the set of players – including i – that share the same id as i at the beginning of slot t). In order to simplify the presentation, we will assume that $|G_i| > 2$. The case $|G_i| = 2$ can be treated separately in a similar manner. Note that it cannot be the case that $|G_i| = 1$, as that would imply that i has obtained a unique id *before* the split slot t, contradicting the fact that i was pending at t.

Let X be the space of all possible transmission vectors X_t for time step t. W.l.o.g., we will restrict X_t, X only to the n' pending players. With slight abuse of notation we refer to X_t both as the transmission vector, and the event that the transmission vector X_t is observed. We define:

$$\begin{aligned} & A^T := \{X_t \in X : X_{it} = 1 \text{ and } X_{jt} = 0, \forall j \in G_i, j \neq i\} \\ \text{and} \quad & A^Q := \{X_t \in X : X_{it} = 0 \text{ and } X_{jt} = 1, \forall j \in G_i, j \neq i\}. \end{aligned}$$

In other words, A^T is the set of all events such that i transmits in t and obtains a unique id, while A^Q is the set of all events such that i stays quiet in t, and obtains a unique id. We also define,

$$\begin{aligned} & B^T := \{X_t \in X : X_{it} = 1 \text{ and } \exists j \in G_i, j \neq i, \text{s.t. } X_{jt} = 1,\} \\ \text{and} \quad & B^Q := \{X_t \in X : X_{it} = 0 \text{ and } \exists j \in G_i, j \neq i, \text{s.t. } X_{jt} = 0,\}. \end{aligned}$$

B^T and B^Q thus contain all possible transmission vectors such that i does not obtain a unique id in t: B^T contains those in which i transmits in t; B^Q contains those in which i stays quiet in t. The sets A^T, A^Q, B^T, B^Q constitute a partition of X.

For any event $x \in X$, let $\mathbf{E}[x]$ be the expected *future cost* of i at the end of slot t given that x was observed at t. Also, for ease of notation, let $\mathbf{Pr}[x]$ correspond to the probability that event x occurs, *given that* $X_{it} = 1$ if $x \in A^T \bigcup B^T$, and *given that* $X_{it} = 0$ if $x \in A^Q \bigcup B^Q$. We now define

$$E_1^T = \sum_{a \in A^T} \mathbf{Pr}[a] \cdot \mathbf{E}[a], \qquad E_2^T = \sum_{b \in B^T} \mathbf{Pr}[b] \cdot \mathbf{E}[b],$$
$$E_1^Q = \sum_{a \in A^Q} \mathbf{Pr}[a] \cdot \mathbf{E}[a], \qquad E_2^Q = \sum_{b \in B^Q} \mathbf{Pr}[b] \cdot \mathbf{E}[b].$$

Then $E^T = E_1^T + E_2^T$ and $E^Q = E_1^Q + E_2^Q$.

At a high level, the approach we take below is to pair up outcomes with equal probability in order to bound the expected gain from deviating from the protocol.

Case 1: Player i obtains a unique id in t, i.e., $A^T \bigcup A^Q$ happens.

We will first consider the events in which i obtains a unique id at t and at the same slot either *all* other players are quiet, or they *all* transmit. In order for i to obtain a unique id at t, she must single herself out, in the first case by transmitting, in the second by being quiet. Let $a^T \in A^T$, $a^Q \in A^Q$ be precisely the events that at t all players in $N' \setminus \{i\}$ stay quiet and i transmits, and that all players in $N' \setminus \{i\}$ transmit and i stays quiet, respectively.

The two events a^T, a^Q have the same probability of occurring, in particular,

$$\mathbf{Pr}[a^T] = \mathbf{Pr}[a^T | X_{it=1}] = \mathbf{Pr}[a^Q | X_{it=0}] = \mathbf{Pr}[a^Q] = \frac{1}{2^{n'-1}}.$$

However, if a^T is observed, then player i has a successful transmission during the split slot t, i.e., there is no need for i to wait for a leave slot to be assigned to her. In this case, the future expected cost of i at the end of t is $\mathbf{E}[a^T] = 0$. On the other hand, if a^Q is observed, and given our assumption that that $n' > 2$, all players but i had a collision during t, and no player exits during the split slot t. Player i obtains a unique id and she will be assigned a leave slot. Since $|G_i| > 2$, i is the only player from her group (and the only player in general in N') that obtained a unique id. Therefore, she is the only player in N' to be assigned a leave slot. The expected future cost of i after t in this case is $\mathbf{E}[a^Q] = 1$.

For all $a_1 \in A^T \setminus \{a^T\}$, i still obtains a unique id at t, but she does not have a successful transmission during the split; this time, there were more players that transmitted in t (but of course no player from G_i other than i). In this case, player i will wait for a leave slot to be assigned to her. Also, for all $a_2 \in A^Q \setminus \{a^Q\}$, no player exits during the split slot t: i is the only player of her group to stay quiet in t; therefore, there are at least two players from G_i that transmitted in t, meaning that no player could succeed during t; there could be other players that obtained a unique id, but they will also have to wait for a leave–slot. All players in $N \setminus \{i\}$ transmit in t with probability $1/2$. The order in which leave slots are assigned, is equally likely to be increasing or decreasing. Therefore, for each $a_1 \in A^T \setminus \{a^T\}$, there is $a_2 \in A^Q \setminus \{a^Q\}$ (in a one-to-one correspondence), such that $\mathbf{Pr}[a_1] = \mathbf{Pr}[a_2]$ and $\mathbf{E}[a_1] = \mathbf{E}[a_2]$. (Namely, a_2 is the event in which each player in N' plays the opposite in t than what she does in a_1.) Thus,

$$
\begin{aligned}
E_1^T &= \sum_{a \in A^T} \mathbf{Pr}[a] \cdot \mathbf{E}[a] = \frac{1}{2^{n'-1}} \cdot 0 + \sum_{a \in A^T \setminus \{a^T\}} \mathbf{Pr}[a] \cdot \mathbf{E}[a], \text{and} \\
E_1^Q &= \sum_{a \in A^Q} \mathbf{Pr}[a] \cdot \mathbf{E}[a] = \frac{1}{2^{n'-1}} \cdot 1 + \sum_{a \in A^Q \setminus \{a^Q\}} \mathbf{Pr}[a] \cdot \mathbf{E}[a].
\end{aligned}
$$

Therefore,

$$
E_1^T - E_1^Q = -\frac{1}{2^{n'-1}} \tag{1}
$$

Case 2: Player i does not obtain a unique id in t, i.e., B^T or B^Q happens.

For each $j \in N' \setminus \{i\}$ let $b_j^Q \in B^Q$ be the event that all players but j were quiet in t (j is the only player in N' that transmitted in t). This player also exits during t, and thus i does not have to wait for a leave slot for j. Similarly let $b_j^T \in B^T$, for all $j \in N' \setminus \{i\}$, be the event that all players but j transmitted in t. (j is the only player in N' that stayed quiet in t). In this case no player exits during the split slot t; Since $|N'| > 2$ (by assumption), and all players but j transmitted, there were at least 2 players (including i) that transmitted – and had a collision– in t. Therefore, all players that obtain a unique id during t must be assigned a leave slot.

Again,

$$
\mathbf{Pr}[b_j^T | X_{it} = 1] = \mathbf{Pr}[b_j^Q | X_{it} = 0] = \frac{1}{2^{n'-1}}.
$$

Also $\mathbf{E}[b_j^Q] = \mathbf{E}[b_j^T] - 1$, since when b_j^Q occurs, i has to wait for one leave slot less than when b_j^T occurs; apart from that the events b_j^T, b_j^Q are again symmetric.

Let $B^{T'} = \{b_1^T, \cdots, b_{i-1}^T, b_{i+1}^T, \cdots, b_{n'-1}^T\}$, $B^{Q'} = \{b_1^Q, \cdots, b_{i-1}^Q, b_{i+1}^Q, \cdots, b_{n'-1}^Q\}$.

Again, for each $b_1 \in B^T \setminus B^{T'}$, there is $b_2 \in B^Q \setminus B^{Q'}$, such that $\mathbf{Pr}[b_1] = \mathbf{Pr}[b_2]$ and $\mathbf{E}[b_1] = \mathbf{E}[b_2]$. Thus,

$$
\begin{aligned}
E_2^Q &= \sum_{b \in B^Q} \mathbf{Pr}[b] \cdot \mathbf{E}[b] \\
&= \sum_{j \in N' \setminus \{i\}} \mathbf{Pr}[b_j^Q] \cdot \mathbf{E}[b_j^Q] + \sum_{B \in B^Q \setminus B^{Q'}} \mathbf{Pr}[b] \cdot \mathbf{E}[b],
\end{aligned}
$$

and

$$
\begin{aligned}
E_2^T &= \sum_{b \in B^T} \mathbf{Pr}[b] \cdot \mathbf{E}[b] \\
&= \sum_{j \in N' \setminus \{i\}} \mathbf{Pr}[b_j^T] \cdot \mathbf{E}[b_j^T] \ + \sum_{b \in B^T \setminus B^{T'}} \mathbf{Pr}[b] \cdot \mathbf{E}[b] \\
&= \sum_{j \in N' \setminus \{i\}} \frac{1}{2^{n'-1}} \left(\mathbf{E}[b_j^Q] + 1 \right) + \sum_{b \in B^Q \setminus B^{Q'}} \mathbf{Pr}[b] \cdot \mathbf{E}[b] \\
&= \sum_{b \in B^Q} \mathbf{Pr}[b] \cdot \mathbf{E}[b] \ + \ \frac{n'-1}{2^{n'-1}}
\end{aligned}
$$

Therefore,

$$
E_2^T - E_2^Q = \frac{n'-1}{2^{n'-1}} \tag{2}
$$

From Eq. 1, 2 we obtain than

$$
E^T - E^Q = E_1^T - E_1^Q + E_2^T - E_2^Q = \frac{n'-2}{2^{n'-1}}.
$$

This shows that staying quiet is a better strategy for any split slot t. Also, the decrease in the future expected cost player i has by being quiet, is $\frac{n'-2}{2^{n'-1}} < \frac{n'-1}{2^{n'-1}}$. □

Corollary 3.3. *Let X^t be any transmission sequence up to time t, and let i be a player that is still pending at the end of time slot t. It is an optimal strategy for i to stay quiet in all split slots $s > t$ (and to follow the protocol during leave slots). For each split slot in which i stays quiet the expected cost of i decreases by less than $\frac{n'-1}{2^{n'-1}}$, where n' is the number of pending players in the beginning of s.*

Proof. Let G_{is} denote the group that i belongs to at the beginning of some split slot $s > t$. The split slot s will result to splitting the players in $G_{is} \setminus \{i\}$ into two new groups, and each of which will have, in expectation, the same size. Therefore, i will join a group whose expected size is the same, whether she transmits in the split or not. Moreover, i has no effect on how groups other than her own split. In particular, i's action cannot (in expectation over the actions of the other players) affect the number of players that obtain a unique id in any split slot (and thus exit the game).

As a result, if it were not for the possibility that players exit during split slots, (and given our assumption that the transmission cost is arbitrarily small), i would have no reason to explicitly choose to deterministically stay quiet during a split over transmitting, or the other way around. Note that we can repeat the steps of the proof of Lemma 3.2, bounding the total gain of a player who defects and stays silent in multiple split slots by computing her future cost at each deviation with respect to *any* sequence incorporating future deviations rather than following the protocol. Therefore, it is beneficial for i to be quiet in *every* future split slot. The bound on the gain then also follows from Lemma 3.2. □

We are now ready to prove Theorem 3.1.

of Theorem 3.1. Consider any given transmission sequence X^t, for some time slot t, and any player i who is still pending at t. Let $A_i(X^t) = t + c\sum_{\tau \leq t} X_{i\tau}$ be the actual cost that i has already incurred up to time t (remember that here we have made the assumption that the transmission cost c is negligible). Let $R_i(X^t)$ be the total expected cost of i, including both her cost so far and her future cost if she follows the protocol from time $t+1$ until she exits the game. Similarly, let $Q_i(X^t)$ be the total expected cost of i, including both her cost so far and her future cost if she always stays quiet during future split–slots (and behaves according to the protocol in leave–slots).

We define F_i^R as the expected *future* cost for following the protocol and F_i^Q the future cost for "staying quiet in all splits" (i.e. F_i^R, F_i^Q measure the expected cost from time $t+1$ until i exits the game). Therefore, $R_i(X^t) = A_i(X^t) + F_i^R$, and $Q_i(X^t) = A_i(X^t) + F_i^Q$. Corollary 3.3 implies that $Q_i(X^t)$ is the minimum possible expected total cost player i can incur, allowing deviations from the protocol. Therefore, in order to show that protocol PERFECT is a $o(1)$–equilibrium, it suffices to show that

$$\frac{R_i(X^t)}{Q_i(X^t)} = 1 + o(1).$$

If the number of players that have already transmitted successfully up to time t (and thus exited the game) is at least $n - \sqrt{n}$, then $A_i(X^t) \geq n - \sqrt{n}$. On the other hand, $F_i^Q \geq 0$, while $F_i^R \leq \log\sqrt{n} + \sqrt{n} \leq 2\sqrt{n}$. (In the worst case all pending players at t share the same id as i, and i will need in expectation $\log\sqrt{n}$ steps to obtain a unique id; similarly, in the worst case, i will be assigned a leave slot last, after all $\sqrt{n}$ remaining players.) This means that

$$\frac{R_i(X^t)}{Q_i(X^t)} = \frac{A_i(X^t) + F_i^R}{A_i(X^t) + F_i^Q} \leq \frac{A_i(X^t) + 2\sqrt{n}}{A_i(X^t)} = 1 + \frac{2\sqrt{n}}{n - \sqrt{n}} = 1 + \frac{2}{\sqrt{n} - 1} = 1 + o(1).$$

Assume now that the pending number of players at t, is more than $\sqrt{n}$. Since $F_i^Q < F_i^R$, and $A_i(X^t) \geq 0$,

$$\frac{R_i(X^t)}{Q_i(X^t)} = \frac{A_i(X^t) + F_i^R}{A_i(X^t) + F_i^Q} \leq \frac{F_i^R}{F_i^Q} = \frac{F_i^R}{F_i^R - D_Q} = 1 + \frac{D_Q}{F_i^R - D_Q}, \tag{3}$$

where D_Q is the expected decrease in the total expected cost of player i if she always stays quiet in split slots after time t. (Remember that Corollary 3.3 implies that the best a player can do to reduce her total expected cost is to stay quiet in each split slot.) Thus, all we need to show is that $\frac{D_Q}{F_i^R - D_Q} = o(1)$.

Let m be the number of pending players in total at time t, let m_i be the number of pending players that share the same id as i at time t, and let S denote the expected number of split slots that i must participate in until he obtains a unique id and exits the game. Note that $S = \Theta(\log m_i)$. Also, let x_k be the expected number of players that exit the game within k rounds (after slot t), with $x_0 = 0$. We remark on the fact that the values of S, x_k remain the same whether i follows the protocol or stays quiet during splits.

The expected gain that i has by staying quiet in the split slot of round $k+1$ is bounded by (Lemma 3.2) $\frac{m-x_k-1}{2^{m-x_k-1}}$. If k_i is the last round that i participates in, then let x be the expected number of players that exit the game in all rounds after slot t and before round k_i begins. Then the expected cost of i if he follows the protocol is $S+x$. Also, the total expected gain that i may have by staying quiet in every split round he participates in, is bounded by $\frac{m-x-1}{2^{m-x-1}}S$. Now, the right hand side of Eq. 3 is bounded by

$$1 + \frac{\frac{m-x-1}{2^{m-x-1}}S}{S + x - \frac{m-x-1}{2^{m-x-1}}S}.$$

If $x \geq \frac{m}{2}$, then

$$\frac{R_i(X^t)}{Q_i(X^t)} < 1 + \frac{\frac{m-x-1}{2^{m-x-1}}S}{S+x-S} = 1 + \frac{(m-x-1)S}{2^{m-x-1}x} < 1 + \frac{S}{x} = 1 + o(1),$$

since $x \geq \frac{m}{2} = \Omega(\sqrt{n})$. If on the other hand $x < \frac{m}{2}$, then

$$\frac{R_i(X^t)}{Q_i(X^t)} < 1 + \frac{\frac{m-x-1}{2^{m-x-1}}S}{S - \frac{m-x-1}{2^{m-x-1}}S} < 1 + \frac{\frac{m/2-1}{2^{m/2-1}}}{1 - \frac{m/2-1}{2^{m/2-1}}} = 1 + o(1).$$

□

4 Collision Multiplicity Feedback

In this section, we present the main result of the paper, Protocol MULTIPLICITY. This protocol works under the CMF channel model, in which, after each slot t, all players are informed about the number of attempted transmissions, i.e., $Y_t = \sum_{i \in N} X_{it}$. This information can be provided by a number of energy detectors, and it is broadcast to all users. CMF is a well-studied and important feedback model in information theory, with many nice theoretical results (see e.g. [25, 20, 9, 23]).

Here we give an overview of the main ideas of Protocol MULTIPLICITY and we will describe it in detail in Section 4.1. Protocol PERFECT presented in Section 3.1 crucially relies on users knowing how each player acts at every time step, and thus does not work under this feedback model. Instead, we design a different $o(1)$-equilibrium protocol that overcomes this difficulty and has expected average cost $2n+c\log n$. Recall that in Protocol PERFECT the *id* of a player is a string denoting all the random choices the player had to make during split slots. For every time t, the players are partitioned into groups according to their ids, i.e., all the players in the same group share the same id. Moreover, all players perform a split at the same time, regardless of the group they belong to. The protocol MULTIPLICITY that we present here, again uses the idea of randomly generated ids, but here, each group splits separately, in a sequential manner. Each split is followed by a *validation* slot, whose aim is to verify that the split has been performed "correctly".

In the CMF model, the players have an incomplete view of history. This might give incentive to some players to "cheat" and pretend to belong to a different group (of smaller size), in order to exit the game sooner. We discourage such a behavior using a "threat": if any deviation from the protocol is observed, then all players switch their transmission strategy according to a costly equilibrium strategy. Finally, a *cheat-detecting* mechanism is needed. The validation slots accomplish exactly this task; the cheating is noted *by all players* if some validation step fails. In that case, all the players get punished by switching to an exponential-cost time-independent equilibrium protocol (punishment protocol):

The punishment protocol Fiat, Mansour and Nadav [8] show that, for transmission cost $c=0$ and k pending players, the time-independent symmetric protocol where every player transmits with probability $p=\Theta(1/\sqrt{k})$ is in equilibrium. It gives average packet latency $e^{\Theta(\sqrt{k})}$, and thus switching to this protocol can be used as a high-cost punishment for detected defections. For our punishment protocol, we adapt this protocol to the general case of transmission cost $c\geq 0$: the transmission probability becomes $p_c=\Theta(1/\sqrt{k(1+c)})$ and the expected cost is $(1+c)e^{\Theta(\sqrt{k/(1+c)})}$.

4.1 Protocol MULTIPLICITY

The protocol works in rounds. Every round consists of a sequence of *split* and *validation* slots such that each player *participates* in exactly one split slot in each round. We say that a player participates in a split slot s, if the protocol requires the player to decide randomly whether she will transmit in s or not.[7]

Assume that at the end of round $k \geq 0$, there are M_k different groups (each group consists of players that have the same transmission history with respect to the split slots they have participated in).[8] Let $G_{j,k}$ be the jth group. The players do not know which players belong to group $G_{j,k}$, but they will be aware of the size $|G_{j,k}|$ of the group.

Consider round $k+1$. At the beginning of the round, the players decide, using (as in the perfect information setting) the parity of the total number x_k of players that transmitted during all the split slots of round k, whether groups will split in increasing or decreasing order of ids. According to this order, every group will perform a split, followed by a validation. Let $G_{j,k}$ be the current group performing a split and validation.

Split slot When it is group $G_{j,k}$'s turn, all players that belong to this group (and only these players) will transmit with probability $1/2$. All players (regardless of whether they belong in $G_{j,k}$ or not) note the number $n_{T,j}$ of the members of $G_{j,k}$ that transmitted in this slot. These $n_{T,j}$ players will form one of the new subgroups of $G_{j,k}$.

Validation slot The immediately next slot is a *validation* slot. All players of $G_{j,k}$ that transmitted in the previous split slot must now stay quiet, while those that did *not*, must now transmit. This second set of players will form the other subgroup of $G_{j,k}$. Again all players can see their number $n_{Q,j}$.

Right after the validation step has happened (and before the next group's turn comes), all players check if the members of $G_{j,k}$ were properly split into the two new subgroups, by checking that the sum of the sizes of the two subgroups equals $|G_{j,k}|$. If that is true, this group is properly divided, and the next group will split.

If the check failed, then there is some player that deviated from the protocol: Either one of the members of $G_{j,k}$ did not transmit in any of the split or validation slots of group $G_{j,k}$ (or even transmitted in both slots); or, some player that did not belong in $G_{j,k}$ transmitted in either of these two slots. All players note therefore the fact that someone has deviated, and they all switch to the punishment protocol.

We note that since now each group splits separately, there is no longer the need for explicit leave-slots. A user that obtains a unique id in round k will transmit successfully when her group $G_{j,k}$ performs a split if she is the only one to transmit (i.e., if she was the only member of $G_{j,k}$ to append a '1' to her id); or she will transmit successfully when her group performs the validation step if she was the

[7] As opposed to a player that the protocol instructs to stay quiet during s with probability 1.

[8] At round 0 all players belong to the same group, i.e., $M_0 = 1$.

only from $G_{j,k}$ that stayed quiet in the split (i.e., she was the only member of $G_{j,k}$ to append a '0' to her id).

If all players of a group $G_{j,k}$ transmitted during their corresponding split, the validation that normally follows will be skipped. The reason is that if all players of $G_{j,k}$ transmitted in the split, no one would transmit in the validation slot. All players would know that this slot should be empty, which would provide an incentive to "attack", i.e., transmit during that slot, disobeying the protocol.

4.2 Analysis

Theorem 4.1. *The expected cost of Protocol* MULTIPLICITY *for any player is* $2n + c\log n$.

Proof. The expected number of split-slots required for a player to obtain a unique id, is $\log n$. Consider a round k. Every player will transmit exactly once during this round (either in the split slot corresponding to her group, or in the validation slot immediately following that split). If M_{k-1} is the number of groups that have formed by the end of round $k-1$, then the duration of round k is at most $2M_{k-1}$ slots, where $M_{k-1} \leq 2^{k-1}$. Therefore the total expected cost for i is bounded by $2n + c\log n$. □

Theorem 4.2. *Protocol* MULTIPLICITY *is a* $o(1)$*-equilibrium.*

Proof. First, we will show that no player has an incentive to cheat. We say that a player "cheats" if she transmits during a slot she was not supposed to or did not transmit when she was expected to do so. If a player belonging to group $G_{j,k}$, for some j,k, did not transmit at the split slot of $G_{j,k}$, nor at the corresponding validation slot, then the sum $n_{T,j} + n_{Q,j}$ will be found less than $|G_{j,k}|$. This will make all players switch to the (high cost) punishment protocol. Similarly, if a player transmits to the validation slot of group $G_{j,k}$ when she was not supposed to, then $n_{T,j} + n_{Q,j} > |G_{j,k}|$.

The remaining case is when a player $i \notin G_{j,k}$ cheats by transmitting during the split slot[9] of group $G_{j,k}$. Assume that i cheats when there are $n' \leq n$ pending players in total. She will get caught with probability at least $1/4$ (if $G_{j,k}$ is of minimum size, i.e. 2).[10] In that case she will have expected future cost $(1+$

[9] Validation slots only happen if there is at least one player from $G_{j,k}$ that is supposed to transmit in them. Thus, a player cannot have a successful transmission by attacking a validation slot.

[10] If none of the members of $G_{j,k}$ transmitted during the split slot, then i has a successful transmission. If all members but one transmitted, then the cheating does not get caught (and i only had a failed transmission). In this case, it looks as if all members of $G_{j,k}$ transmitted, and the validation slot is skipped. In all other cases, i gets caught.

$c)e^{\Theta(\sqrt{n'/(1+c)})}$, worse than the corresponding cost of following the protocol, i.e., $O(n')$. Therefore, the expected cost of i becomes larger if she deviates instead of following the protocol.

We note that a player cannot gain by deterministically choosing to stay quiet during a split she participates in. She cannot save on the transmission cost c, as she would have to transmit during the validation slot anyway. Moreover, staying quiet or transmitting in the split does not affect the number of rounds a player must wait until she obtains a unique id: Consider a player i belonging to some group $G_{j,k}$. All other players from $G_{j,k}$ transmit during the corresponding split slot with probability $1/2$; the expected sizes of the two new groups to be formed are the same and i has exactly the same probability of obtaining a unique id, whether she transmits in the split, or stays quiet.

On the other hand, splits always happen before the corresponding validations. If i obtains a unique id during round k, then she exits the game earlier if she was the only player of her group transmitting at the split, than if she were the only quiet player. This implies that "always transmit" gives a smaller expected (future) cost. It is therefore the optimal strategy. Nevertheless, if p_τ is the probability for a player that transmits during split slots to succeed at the time step τ, then p_τ is also the probability for a player that stays quiet during split slots to succeed at the time step $\tau+1$ (since validation slots occur immediately after split slots). Let X^t be any transmission history, and suppose that $t+1$ is a split slot that i participates in (otherwise i's optimal strategy is to behave according to the protocol). Let $A(X^t) = t + \sum_{\tau \le t} X_{i\tau}$ be the actual cost incurred by i until time t, and let $T(X^t), Q(X^t)$ be the total expected cost of player i if she always transmits, stays quiet, respectively, in all future split slots she participates in. Then,

$$\begin{array}{rcl} T(X^t) - A(X^t) & = & \sum_{\tau=t+1}^{\infty} \tau p_\tau \\ Q(X^t) - A(X^t) & = & \sum_{\tau=t+1}^{\infty} (\tau+1) p_\tau = \sum_{\tau=t+1}^{\infty} (\tau p_\tau + p_\tau) = T(X^t) - A(X^t) + 1, \end{array}$$

and therefore, $\frac{Q(X^t)}{T(X^t)} = \frac{T(X^t)+1}{T(X^t)} = 1 + o(1)$.

$Q(X^t)$ is an upper bound on the total expected cost of the protocol: Staying quiet in split slots is always worse than transmitting; according to the protocol a player stays quiet only in half (in expectation) of the future split slots she participates in. Thus, Protocol MULTIPLICITY is a $o(1)$-equilibrium. □

Protocol PERFECT with non-negligible transmission cost Going back to the perfect information protocol, we can now easily adapt protocol PERFECT for the case that c is non-negligible. All we have to do, is add a validation slot after each split (all players that do not transmit during the split, must transmit during the validation). The punishment protocol is used again to force players to transmit in exactly one of the split and validation slots of each round. As in Protocol MULTIPLICITY, this way a player cannot decrease her transmission costs by staying quiet in a split. The analysis becomes similar to the analysis of Protocol MULTIPLICITY.

References

[1] N. Abramson. The ALOHA system: Another alternative for computer communications. In *Proceedings of the November 17-19, 1970, fall joint computer conference*, pages 281–285. ACM New York, NY, USA, 1970.

[2] E. Altman, R. El Azouzi, and T. Jiménez. Slotted aloha as a game with partial information. *Comput. Netw.*, 45(6):701–713, 2004.

[3] E. Altman, D. Barman, A. Benslimane, and R. El Azouzi. Slotted aloha with priorities and random power. In *Proc. IEEE Infocom*, 2005.

[4] V. Auletta, L. Moscardelli, P. Penna, and G. Persiano. Interference games in wireless networks. In *WINE*, pages 278–285, 2008.

[5] M. Bender, M. Farach-Colton, S He, B. Kuszmaul, and C. Leiserson. Adversarial contention resolution for simple channels. In *SPAA '05*, pages 325–332. ACM, 2005.

[6] J. Capetanakis. Generalized tdma: The multi-accessing tree protocol. *IEEE Transactions on Communications*, 27(10):1476–1484, 1979.

[7] J. Capetanakis. Tree algorithms for packet broadcast channels. *IEEE Transactions on Information Theory*, 25(5):505–515, 1979.

[8] A. Fiat, Y. Mansour, and U. Nadav. Efficient contention resolution protocols for selfish agents. In *SODA '07*, pages 179–188, Philadelphia, PA, USA, 2007. SIAM.

[9] L. Georgiadis and P. Papantoni-Kazakos. A collision resolution protocol for random access channels with energy detectors. *IEEE Transactions on Communications*, COM-30:2413–2420, November 1982.

[10] Mihály Geréb-Graus and Thanasis Tsantilas. Efficient optical communication in parallel computers. In *SPAA '92*, pages 41–48, New York, NY, USA, 1992. ACM.

[11] L. A. Goldberg and P. D. MacKenzie. Analysis of practical backoff protocols for contention resolution with multiple servers. *J. Comput. Syst. Sci.*, 58(1):232–258, 1999.

[12] L. A. Goldberg, P. D. Mackenzie, M. Paterson, and A. Srinivasan. Contention resolution with constant expected delay. *J. ACM*, 47(6):1048–1096, 2000.

[13] A. Greenberg and S. Winograd. A lower bound on the time needed in the worst case to resolve conflicts deterministically in multiple access channels. *J. ACM*, 32(3):589–596, 1985.

[14] Hayes J. An adaptive technique for local distribution. *IEEE Transactions on Communications*, 26(8):1178–1186, 1978.

[15] E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In *STACS'99*, pages 404–413, 1999.

[16] R.T. Ma, V. Misra, and D. Rubenstein. Modeling and analysis of generalized slotted-aloha mac protocols in cooperative, competitive and adversarial environments. In *ICDCS '06*, page 62, Washington, DC, USA, 2006. IEEE.

[17] A. MacKenzie and S. Wicker. Stability of multipacket slotted aloha with selfish users and perfect information, 2003.

[18] P. D. MacKenzie, C. G. Plaxton, and R. Rajaraman. On contention resolution protocols and associated probabilistic phenomena. *J. ACM*, 45(2):324–378, 1998.

[19] I. Menache and N. Shimkin. Efficient rate-constrained nash equilibrium in collision channels with state information. In *INFOCOM 2008.*, pages 403–411, 2008.

[20] N. Pippenger. Bounds on the performance of protocols for a multiple-access broadcast channel. *IEEE Transactions on Information Theory*, 27(2):145–151, 1981.

[21] P. Raghavan and E. Upfal. Stochastic contention resolution with short delays. Technical report, Weizmann Science Press of Israel, Jerusalem, Israel, Israel, 1995.

[22] L. Roberts. Aloha packet system with and without slots and capture. *SIGCOMM Comput. Commun. Rev.*, 5(2):28–42, April 1975.

[23] M. Ruszinko and P. Vanroose. How an erdöos-rényi-type search approach gives an explicit code construction of rate 1 for random access with multiplicity feedback. *IEEE Transactions on Information Theory*, 43(1):368–372, 1997.

[24] V. Srivastava, J. A. Neel, A. B. MacKenzie, J. E. Hicks, L. A. DaSilva, J. H. Reed, and R. P. Gilles. Using game theory to analyze wireless ad hoc networks. *IEEE Communications Surveys and Tutorials*, 7(5):46–56, 2005.

[25] B. Tsybakov. Resolution of a conflict of known multiplicity. *Problemy Peredachi Informatsii*, 1980.

[26] B. S. Tsybakov and V. A. Mikhailov. Free synchronous packet access in a broadcast channel with feedback. *Problems of Information Transmission*, 14(4):259–280, 1978.

[27] D. Wang, C. Comaniciu, and U. Tureli. Cooperation and fairness for slotted aloha. *Wirel. Pers. Commun.*, 43(1):13–27, 2007.

[28] D. Zheng, W. Ge, and J. Zhang. Distributed opportunistic scheduling for ad-hoc communications: an optimal stopping approach. In *MobiHoc '07*, pages 1–10. ACM, 2007.

Protocol 4.1 MULTIPLICITY

Require: Code for player i, at the beginning of round $k+1$:

$G_{1,k}, G_{2,k}, \ldots, G_{M_k,k}$	:	groups formed by the end of round k in order of increasing id
$index_i$	:	index of the group of player i in the sorted (according to the id corresponding to each group) sequence of groups at the end of round k
x_k	:	#players that transmitted in total in all the split slots of round k.

for j in *increasing* order if x_k is even, in *decreasing* order if x_k is odd **do**
  if $j \neq index_i$ **then**
    Stay quiet in current *split* slot s {Group $G_{j,k}$ performs a split}
    $n_{T,j}$:= #players that transmitted in slot s
    if $n_{T,j} \neq |G_{j,k}|$ **then** {The validation only happens if not all players of $G_{j,k}$ transmitted}
      Stay quiet in current *validation* slot s' {Group $G_{j,k}$ validates the number of its members that stayed quiet in the previous slot}
      $n_{Q,j}$:= #players that transmitted in slot s'
    else {$n_{T,j} = |G_{j,k}|$ holds}
      $n_{Q,j} := 0$
  else {Now is the turn of i's group}
    Transmit in current *split* slot s with probability $1/2$
    $n_{T,index_i}$:= #players that transmitted in slot s
    if $n_{T,index_i} \neq |G_{index_i,k}|$ **then** {The validation only happens if not all players of group $G_{index_i,k}$ transmitted}
      if i transmitted in s **then**
        Stay quiet in current *validation* slot s'
      else
        Transmit in current *validation* slot s'
      $n_{Q,index_i}$:= #players that transmitted in slot s'
    else {$n_{T,index_i} = |G_{index_i,k}|$ holds}
      $n_{Q,index_i} := 0$
  if $n_{T,j} + n_{Q,j} \neq |G_{j,k}|$ **then** {A player tried to "cheat"}
    Switch to the punishment protocol

[End of round $k+1$]