

# Incentivizing Sharing in Realtime D2D Streaming Networks: A Mean Field Game Perspective

Jian Li\*, Rajarshi Bhattacharyya\*, Suman Paul\*, Srinivas Shakkottai\*, and Vijay Subramanian†

\*Dept. of ECE, Texas A&M University, †Dept. of EECS, University of Michigan

Email: \*{lj0818, rajarshibh, sumanpaul, sshakkot}@tamu.edu, †vgsubram@umich.edu

**Abstract**—We consider the problem of streaming live content to a cluster of co-located wireless devices that have both an expensive unicast base-station-to-device (B2D) interface, as well as an inexpensive broadcast device-to-device (D2D) interface, which can be used simultaneously. Our setting is a streaming system that uses a block-by-block random linear coding approach to achieve a target percentage of on-time deliveries with minimal B2D usage. Our goal is to design an incentive framework that would promote such cooperation across devices, while ensuring good quality of service. Based on ideas drawn from truth-telling auctions, we design a mechanism that achieves this goal via appropriate transfers (monetary payments or rebates) in a setting with a large number of devices, and with peer arrivals and departures. Here, we show that a Mean Field Game can be used to accurately approximate our system. Furthermore, the complexity of calculating the best responses under this regime is low. We implement the proposed system on an Android testbed, and illustrate its efficient performance using real world experiments.

## I. INTRODUCTION

There has recently been much interest in networked systems for collaborative resource utilization. These are systems in which agents contribute to the overall welfare through their individual actions. Usually, each agent has a certain amount of resources, and can choose how much to contribute based on the perceived return via repeated interactions with the system. An example is a peer-to-peer file sharing network, wherein each peer can contribute upload bandwidth by transmitting chunks to a peer, and receive downloads of chunks from that peer as a reward. Interactions are bilateral, and hence tit-for-tat type strategies are successful in preventing free-riding behavior [1]. More generally, collaborative systems entail multilateral interactions in which the actions of each agent affect and are affected by the collective behavior of a subset of agents. Here, more complex mechanisms are needed to accurately determine the value of the contribution of each individual to the group.

An example of a collaborative system with repeated multilateral interactions is a device-to-device (D2D) wireless network. Suppose that multiple devices require the same content chunk. The broadcast nature of the wireless medium implies that several agents can be simultaneously satisfied by a single transmission. However, they might each have different values

for that particular chunk, and may have contributed different amounts in the past to the transmitting agent. Furthermore, D2D systems undergo “churn” in which devices join and leave different clusters as they move around. How then is an agent to determine whether to collaborate with others, and whether it has received a fair compensation for its contribution?

Our objective in this paper is to design mechanisms for cooperation in systems with repeated multilateral interactions. As in earlier literature, we assume that there exists a currency to transfer utility between agents [2], [3], and our goal is to determine how much should be transferred for optimal collaboration. We focus on wireless content streaming as our motivating example. In particular, as shown in Figure 1, we assume that all devices are interested in the same content stream, and receive a portion of chunks corresponding to this stream via a unicast base-station-to-device (B2D) interface. The B2D interface has a large energy and dollar cost for usage, and the devices seek to mitigate this cost via simultaneously sharing chunks through broadcast D2D communication.

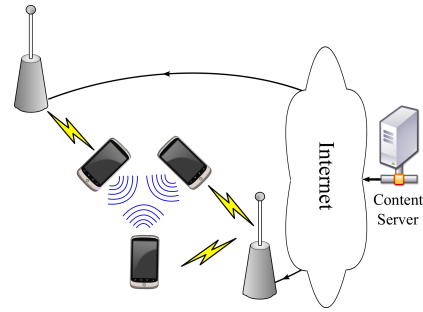


Fig. 1. Wireless content distribution via multiple interfaces [4].

A content sharing system is described in [4], in which the objective is to achieve *live streaming* of content synchronously to multiple co-located devices. The system architecture of that work forms an ideal setting for studying mechanism design in which multilateral interactions occur. The setup is illustrated in Figure 2. Here, time is divided into *frames*, which are subdivided into *T slots*. A *block* of data is generated by the content server in each frame, and the objective is to ensure that this block can be played out by all devices two frames after its generation, *i.e.*, data block  $k$  is generated in frame  $k - 2$ , and is to be played out in frame  $k$ . Such a strict delay constraint between the time of generation and playout of each data block ensures that the *live* aspect of streaming is maintained.

Research was funded in part by NSF grants CNS 1149458, AST 1443891, III 1219071, AST 1343381, AST 1443972, DTRA grant HDTRA 1-13-1-0030 and AFOSR FA9550-13-1-0008. 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 funding agencies.

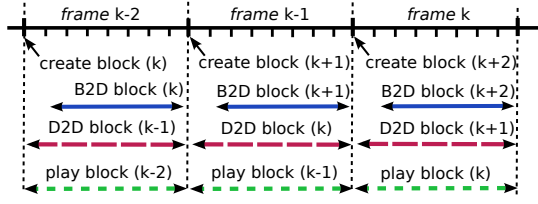


Fig. 2. Streaming architecture [4] in which each block must be delivered within two frames after its creation.

Upon generation of block  $k$  (in frame  $k - 2$ ), the content server divides it into  $N$  chunks and performs random linear coding (RLC) over these chunks [5]. The server unicasts some of these coded chunks to each device using its B2D interface. This number is to be kept small to reduce B2D usage. Next, in frame  $k - 1$ , the devices use the broadcast D2D network to disseminate these chunks among themselves. At the end of frame  $k - 1$  the devices attempt to decode block  $k$ . If a device  $i$  has received enough coded chunks to decode the block, it plays out that block during frame  $k$ . Otherwise,  $i$  will be idle during this frame. The use of RLC results in two desirable system features. *First*, the server can unicast a fixed number of chunks to the devices in each frame over a lossy channel (Internet plus B2D link) without any feedback. *Second*, the devices do not need to keep track of what chunks each one possesses while performing D2D broadcasts.

The notion of quality of experience (QoE) here is *delivery-ratio*, which is the average ratio of blocks desired to the blocks generated [6]. For instance, a delivery ratio of 95% would mean that it is acceptable if 5% of the blocks can be skipped. In [4], it was shown that, assuming complete cooperation by the participating devices, it is possible to design a chunk sharing scheme whereby all devices would meet their QoE targets with minimal usage of the B2D interface. *But how do we design a mechanism to ensure that the devices cooperate?*

The setting of interest in this paper is that of a large number of D2D clusters, each with a fixed number of agents, and with all clusters interested in the same content stream. Examples of such settings are sports stadia, concerts or protest meetings, where a large number of agents gather together, and desire to receive the same live-stream (replays, commentary, live video *etc.*) Devices move between clusters as agents move around, causing churn. The objective of our work is to develop an incentive framework wherein each device truthfully reports the number of chunks that it receives via B2D and its deficit in each frame, so that a system-wide optimal allocation policy can be employed. Such an incentive framework should be lightweight and compatible with minimal amounts of history retention. Finally, we also desire to implement the system on Android devices and measure its real world performance.

### Related Work

The question of how to assign value to wireless broadcast transmissions is intriguing. For instance, [7] considers a problem of repeated interaction with time deadlines by which each node needs to receive a packet. Each node declares its

readiness to help others after waiting for a while; the game lies in choosing this time optimally, and the main result is to characterize the price of anarchy that results. However, decision making is myopic, *i.e.*, devices do not estimate future states while taking actions. In a similar fashion, [3] propose a scheme for sharing 3G services via WiFi hotspots using a heuristic scheme that possesses some attractive properties. Here too, decision making is myopic.

*Perfect Bayesian and Mean Field Equilibria:* The typical solution concept in dynamic games is that of the Perfect Bayesian Equilibrium (PBE). The PBE requires each agent to keep track of their beliefs on the future plays of all other agents in the system, and play the best response to that belief. The dynamic pivot mechanism [8] extends the truth-telling VCG idea [9] to dynamic games. It provides a basis for designing allocation schemes that are underpinned by truthful reporting.

Computation of PBE becomes intractable when the number of agents is large. An accurate approximation in this regime is that of a Mean Field Game (MFG) [10]. In MFG, the agents assume that each opponent would play an action drawn *independently* from a fixed distribution over its action space. The agent chooses an action that is the best response against actions drawn in this fashion. The system is said to be at Mean Field Equilibrium (MFE) if this best response action is itself a sample drawn from the assumed distribution, *i.e.*, the assumed distribution and the action are consistent with each other [11], [12]. We will use this concept in our setting where there are a large number of peer devices with peer churn.

To the best of our knowledge, there is no prior work that considers truth-telling mechanism design for multilateral repeated games in the mean field setting. In particular, we will see that aligning two concepts of value—from the system perspective and from that of the agents—is crucial to our goal.

### Organization and Main Results

We describe our system model in Section II. Our system consists of a large number of clusters, with agents moving between clusters. The lifetime of an agent is geometric; an agent is replaced with a new one when it exits. Each agent receives a random number of B2D chunks by the beginning of each frame, which it then shares using D2D transmissions.

In Section III, we present a MFG approximation of the system, which is accurate when the number of clusters is large. Here, the agents assume that the B2D chunks received and deficits of the other agents would be drawn independently from some distribution in the future, and optimize against that assumption when declaring their states. The objective is to incentivize agents to truthfully report their states (B2D chunks and deficit) such that a schedule of transmissions (called an “allocation”) that minimizes the discounted sum of costs can be used in each frame. The mechanism takes the form of a scheme in which tokens are used to transfer utility between agents. A nuance of this regime is that while the system designer sees each cluster as having a new set of users (with IID states) in each time frame, each user sees states of all its competitors *but not itself* as satisfying the

mean field distribution. Reconciling the two view points is needed to construct a cost minimizing pivot mechanism, whose truth-telling nature is shown in Section IV. This is our main contribution in this paper. The allocation itself turns out to be simple, and follows a min-deficit first policy [4].

Next, in Sections V–VI, we present details on how to prove the existence of the MFE in our setting. Due to a paucity of space, we only present an intuitive view of many steps. Although this proof is not trivial, it follows in a high-level sense in the manner of [11], [12]. We then turn to computing the MFE and the value functions needed to determine the transfers in Section VII. The value iteration needed to choose allocation is straightforward. Throughout the paper, we do not present proofs of the main results, which are available in a technical report [13].

We present details of our Android implementation of a music streaming app used to collect real world traces in Section VIII. We discuss the viability of our system in Section IX, and illustrate that under the current price of cellular data access, our system provides sufficient incentives to participate. Finally, we conclude in Section X.

## II. CONTENT STREAMING MODEL

We consider a large number of D2D clusters, with all clusters interested in the same content stream. We assume that each cluster consists of  $M$  co-located peer devices<sup>1</sup> denoted by  $i \in \{1, \dots, M\}$ . The data source generates the stream in the form of a sequence of blocks. Each block is further divided into  $N$  chunks for transmission. We use random linear network coding over the chunks of each block (with coefficients in finite field  $F_q$  of size  $q$ ). We assume that the field size is very large; this assumption can be relaxed without changing our cooperation results. Time is divided into frames, which are further divided into slots. At each time slot  $\tau$ , each device can simultaneously receive up to one chunk on each interface. We assume that chunks are encrypted so that non-participants cannot use overheard chunks. However, we still need to ensure cooperation by system participants.

**B2D Interface:** Each device has a (lossy) B2D unicast channel to a base-station. For each device  $i$ , we model the number of chunks received using the B2D interface in the previous frame by a random variable with (cumulative) distribution  $\zeta$ , independent of the other devices. The support of  $\zeta$  is the set  $\{0, 1, \dots, T\}$ , denoted by  $\mathbb{T}$ . The statistics of this distribution depend on the number of chunks transmitted by the server and the loss probability of the channel. In [4], a method for calculating statistics based on the desired quality of service is presented. We take the distribution  $\zeta$  as given.

**D2D Interface:** Each device has a zero-cost D2D broadcast interface, and only one device can broadcast over the D2D network at each time  $\tau$ . For simplicity of exposition, we will assume that the D2D broadcasts are always successful; the more complex algorithm proposed in [4] to account for unreliable D2D is fully consistent with our incentive scheme. Since

each D2D broadcast is received by all devices, there is no need to rebroadcast any information. It is then straightforward to verify that the order of D2D transmissions does not impact performance. Thus, we only need to keep track of the number of chunks transmitted over the D2D interfaces during a frame in order to determine the final state of the system.

**Allocation:** We denote the total number of coded chunks of block  $k$  delivered to device  $i$  via the B2D network during frame  $k - 2$  using  $e_i[k] \sim \zeta$ . We call the vector consisting of the number of transmissions by each device via the D2D interfaces over frame  $k - 1$  as the “allocation” pertaining to block  $k$ , denoted by  $\mathbf{a}[k]$ . Also, we denote the number received chunks of block  $k$  by device  $i$  via D2D during frame  $k - 1$  using  $g_i[k]$ . Due to the large field size assumption, if  $e_i[k] + g_i[k] = N$ , it means that block  $k$  can be decoded, and hence can be played out. For simplicity of exposition, we develop our results assuming that the allocation is computed in a centralized fashion in each cluster. However, we actually implement a distributed version on the testbed and assume that the announcement of the deficit and chunks received over the D2D interface is heard by all participating devices.

**Quality of Experience:** Each device  $i$  has a delivery ratio  $\eta_i \in (0, 1]$ , which is the minimum acceptable long-run average number of frames device  $i$  must playout. In the mobile agents model, we assume that all devices have the same delivery ratio  $\eta$  for simplicity. It is straightforward to extend our results to the case where delivery ratios are drawn from some finite set of values. The device keeps track of the current deficit using a deficit queue with length  $d_i[k] \in \mathbb{K}$ . The set of possible deficit values is given by  $\mathbb{K} = \{k\eta - m : k, m \geq 0, m \leq \lfloor k\eta \rfloor\}$ , where for  $x \in \mathbb{R}$ ,  $\lfloor x \rfloor = \max\{k \in \mathbb{Z} : k \leq x\}$  is the largest whole number that  $x$  is greater than. Note that  $\mathbb{K}$  is a countable set and the possible deficit values are all non-negative. In fact, by the well-ordering principle  $\mathbb{K}$  can be rewritten as  $\{d_n\}_{n \in \mathbb{N}}$  with  $d_n$  an increasing sequence (without bound) such that  $d_1 = 0$ . We will use this representation to enumerate the elements of  $\mathbb{K}$ . If a device fails to decode a particular block, its deficit increases by  $\eta$ , else it decreases by  $1 - \eta$ . The impact of deficit on the user’s quality of experience is modeled by a function  $c(d_i[k])$ , which is convex, differentiable and monotone increasing. The idea is that user unhappiness increases more with each additional skipped block.

**Transfers:** We assume the existence of a currency (either internal or a monetary value) that can be used to transfer utility between agents [2], [3]. In our system, a negative transfer is a price paid by the agent, while a positive value indicates that the agent is paid by the system. Such transfer systems are well established; see for instance a review in [2]. Transfers are used by agents either to pay for value received through others’ transmissions, or to be compensated for value added to others by transmitting a chunk. We assume that the transmissions in the system are monitored by a reliable device, which can then report these values to decide on the transfers. In practice we use the device that creates each ad-hoc network as the monitor.

An *allocation policy* maps the values of the B2D chunks received and deficits as revealed by agents, denoted by  $\theta[k] :=$

<sup>1</sup>Our analysis is essentially unchanged when there are a random but finite number of devices in each cluster.

$(\hat{e}[k], \hat{d}[k-1])$ , to an allocation for that frame  $\mathbf{a}[k]$ . Given an allocation, agents have no incentive to deviate from it, since an agent that does not transmit the allocated number of chunks would see no benefit; those time slots would have no transmissions by other agents either. The fundamental question is that of how to incentivize the agents to reveal their states truthfully so that the constructed allocation can maximize system-wide welfare.

### III. MEAN FIELD MODEL AND MECHANISM DESIGN

Our system consists of  $JM$  agents (or users) organized into  $J$  clusters with  $M$  agents per cluster. As mentioned earlier, time is slotted into frames. At the end of a frame, any agent  $i$  can leave the system only to be replaced by a new agent (also denoted by  $i$ ) whose initial deficit is drawn from a (cumulative) distribution  $\Psi$  with support  $\mathbb{K}$ . This event occurs with probability  $\bar{\delta} = (1 - \delta)$  independently for each agent, so that the lifetimes of the agents are geometrically distributed. As described in the previous section, we assume that the number of chunks received via B2D for agent  $i$  in frame  $k$ , denoted by  $e_i[k]$ , is chosen in an *i.i.d.* fashion according to the (cumulative) distribution  $\zeta$ , with support  $\mathbb{T}$ ; one such distribution is the binomial distribution. In addition to the agents having geometrically distributed lifetimes, we also allow mobility in our set-up. In particular, in every frame we assume that all the agents are randomly permuted and then assigned to clusters such that there are exactly  $M$  agents in each cluster. Using this system as a starting point we will develop our mean-field model that will be applicable when the number of clusters  $J$  is extremely large.

The mean field framework in Figure 3 illustrates system relationships that will be discussed below. The blue/dark tiles apply to the value determination process for mechanism design, which will be discussed in this section. The beige/light tiles are relevant to showing the existence of an MFE on which the mechanism depends, which will be discussed in Sections V–VI. Due to space constraints, proofs of all results are deferred to a technical report [13].

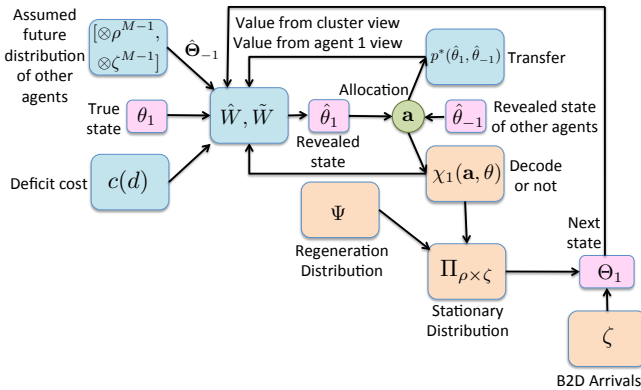


Fig. 3. The Mean Field system from perspective of agent 1.

The mean field model yields informational and computational savings, since otherwise each agent will need to not only be cognizant of the values and actions of all agents, but

also track their mobility patterns. Additionally, the mean field distribution accounts for regenerations, which do not have to be explicitly accounted for when determining best responses.

There is, however, an important nuance that the mean-field analysis introduces: when there are a large number of clusters, each cluster sees a different group of agents in every frame with their states drawn from the mean-field distribution, but even though each agent interacts with a new set of agents in every frame, it's own state is updated based on the allocations made to it, so that the differing viewpoints of the two entities need to be reconciled while providing any incentives.

The number of chunks received over the B2D interface and the deficit value constitute the state of an agent at the beginning of a frame. At frame  $k$  we collect together the state variables of all the agents in system as  $\theta[k] = (e[k], d[k-1])$ . Our mechanism then aims to achieve

$$W(\theta[k]) = \min_{\{\mathbf{a}[l]\}_{l=k}^{\infty}} \mathbb{E} \left\{ \sum_{j=1}^J \sum_{l=k}^{\infty} \delta^{l-k} \sum_{i \in s_j[l]} v_i(\mathbf{a}_{s_j}[l], \theta_i[l]) \right\}, \quad (1)$$

where  $j = 1, 2, \dots, J$  is the number of clusters in the system,  $s_j[k]$  is the set of agents in cluster  $j$  at frame  $k$ ,  $\mathbf{a}_{s_j}$  is the allocation in cluster  $j$  and  $v_i(\mathbf{a}_{s_j}[l], \theta_i[l])$  is the value that agent  $i$  makes from the allocation in frame  $k$ . For agent  $i$  set  $j_i[k]$  to be the cluster he belongs in during frame  $k$ , i.e.,  $i \in s_{j_i[k]}[k]$ .

Given the allocation in each cluster, if agent  $i$  does not regenerate, then his deficit gets updated as

$$d_i[k] = (d_i[k-1] + \eta - \chi_i(\mathbf{a}_{j_i[k]}[k], \theta_i[k]))^+, \quad (2)$$

where  $(\cdot)^+ = \max(\cdot, 0)$ , whereas if the agent regenerates, then  $d_i[k] = \tilde{d}_i[k]$  where  $\tilde{d}_i[k]$  is drawn *i.i.d.* with distribution  $\Psi$ . Here,

$$\chi_i(\mathbf{a}, \theta_i) = 1_{\{e_i + g_i(\mathbf{a}) = N\}} = \begin{cases} 1 & \text{if } e_i + g_i(\mathbf{a}) = N \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where  $\chi_i(\cdot)$  is 1 if and only if agent  $i$  obtains all  $N$  coded chunks to be able to decode a block,  $g_i(\mathbf{a})$  is the number of packets agent  $i$  can get during a frame under the allocation  $\mathbf{a}$  (where we suppress the dependence of  $\mathbf{a}$  on  $\theta$ ). We specialize to the case where the value per frame for agent  $i$  with system state  $\theta$  and vector of allocations  $\mathbf{a}$  is given by  $v_i(\mathbf{a}, \theta_i) = c((d_i + \eta - \chi_i(\mathbf{a}, \theta_i))^+)$  if there is no regeneration and  $v_i(\mathbf{a}, \theta_i) = c(\tilde{d}_i)$  otherwise, where  $\tilde{d}_i$  is *i.i.d.* with distribution  $\Psi$  and  $c(\cdot)$  is the holding cost function that is assumed to be convex and monotone increasing.

As there are a large number of clusters, in every frame there is a completely different set of agents that appear at any given cluster. The revealed states of these agents will be drawn from the mean field distribution. Hence, from the perspective of some cluster  $l$ , the revealed state of the agents in that cluster  $\hat{\theta}_l$  will be drawn according to the (cumulative) distributions  $[\otimes \rho^M, \otimes \zeta^M]$ , with  $\rho$  pertaining to the deficit, and  $\zeta$  pertaining to the B2D transmissions received by that agent. Note that the support of  $\rho$  is  $\mathbb{K}$  while the support of  $\zeta$  is  $\mathbb{T}$ ,

and  $\otimes$  indicates the *i.i.d* nature of the agent states. Whereas from the perspective a particular agent  $i$ , the revealed states of all the other agents in that cluster will be drawn according to  $\hat{\Theta}_{-i} \sim [\otimes \rho^{M-1}, \otimes \zeta^{M-1}]$ . These facts will simplify the allocation problem in each cluster and also allow us to analyze the MFE by tracking a particular agent.

First, we consider the allocation problem as seen by the clusters. Pick any finite number of clusters. In the mean-field limit, the agents from frame to frame will be different in the entire set, therefore the allocation decision in each cluster can be made in an distributed manner, independent of the other clusters; this is one of the chaos hypotheses of the mean-field model. This then implies that the objective in (1) is achieved by individual optimization in each cluster, *i.e.*,

$$W(\hat{\theta}[k]) = \sum_{j=1}^J W_j(\hat{\theta}_{s_j}[k]), \quad (4)$$

where we recall that  $\hat{\theta}_{s_j}[k]$  is the revealed state of agents in cluster  $j$  at time  $k$  and

$$W_j(\hat{\theta}_{s_j}[k]) = \min_{\{\mathbf{a}_{s_j}[l]\}_{l=k}^{\infty}} \sum_{l=k}^{\infty} \delta^{l-k} \sum_{i \in s_j[l]} v_i(\mathbf{a}_{s_j}[l], \hat{\theta}_i[l]). \quad (5)$$

The value function in the mean-field is determined by the first solving the following Bellman equation

$$\hat{W}(\hat{\theta}) = \min_{\mathbf{a}} \sum_{i=1}^M v_i(\mathbf{a}, \hat{\theta}_i) + \delta \mathbb{E} \left\{ \hat{W}(\hat{\Theta}) \right\} \quad (6)$$

to obtain function  $\hat{W}(\cdot)$ , where  $\hat{\theta}$  is the  $M$ -dimensional revealed state vector (with elements  $\hat{\theta}_i$ ) and the future revealed state vector  $\hat{\Theta}$  is chosen according to  $[\otimes \rho^M, \otimes \zeta^M]$ , and thereafter setting  $W_j(\hat{\theta}_{s_j}[k]) = \hat{W}(\hat{\theta}_{s_j}[k])$  for every  $j = 1, 2, \dots, J$ . This observation then considerably simplifies the allocation in each cluster to be the greedy optimal, *i.e.*, determine (multi)function

$$\mathbf{a}^*(\hat{\theta}) = \arg \min_{\mathbf{a}} \sum_{i=1}^M v_i(\mathbf{a}, \hat{\theta}_i), \quad (7)$$

and for  $j = 1, 2, \dots, J$  we set  $\mathbf{a}_{s_j}^* = \mathbf{a}^*(\hat{\theta}_j)$ .

Next, we consider the system from the viewpoint of a typical agent  $i$ ; w.l.o.g let  $i = 1$ . Any allocation results in the deficit changing according to (2) and the future B2D packets drawn according to  $\zeta$ , whereas the state of every other agent that agent 1 interacts with in the future gets chosen according to the mean field distribution. Then the value function (of the cluster) from the perspective of agent 1 is determined using

$$\begin{aligned} \tilde{W}(1, (\hat{\theta}_1, \hat{\theta}_{-1})) = \\ \min_{\mathbf{a}} \sum_{i'=1}^M v_{i'}(\mathbf{a}, \hat{\theta}_{i'}) + \delta \mathbb{E} \left\{ \tilde{W}(1, (\hat{\Theta}_1, \hat{\Theta}_{-1})) | \mathbf{a}, \hat{\theta}_1 \right\}. \end{aligned} \quad (8)$$

Here,  $\hat{\theta}_{-1}$  represents the revealed states of all the agents in cluster except 1,  $\hat{\Theta}_{-1} \sim [\otimes \rho^{M-1}, \otimes \zeta^{M-1}]$ , and for  $\hat{\Theta}_1$ , the deficit term is determined via (2) (setting  $\theta_i = \hat{\theta}_i$ ) while the

B2D term follows  $\zeta$ . This recursion yields a function  $\tilde{W}(1, \cdot)$  which applies to all agents. Using this function, one can also determine the allocation that agent 1 expects his cluster to perform, namely,

$$\begin{aligned} \tilde{\mathbf{a}}(\hat{\theta}_1, \hat{\theta}_{-1}) = \\ \arg \min_{\mathbf{a}} \sum_{i'=1}^M v_{i'}(\mathbf{a}, \hat{\theta}_{i'}) + \delta \mathbb{E} \left\{ \tilde{W}(1, (\hat{\Theta}_1, \hat{\Theta}_{-1})) | \mathbf{a}, \hat{\theta}_1 \right\}. \end{aligned} \quad (9)$$

Using the two allocations  $\mathbf{a}^*$  and  $\tilde{\mathbf{a}}$  we can write down the value of agent 1 from the system optimal allocation and the value of agent 1 in the allocation that the agent thinks that the system will be performing. For a given allocation function  $\mathbf{a}(\cdot)$  (for the state of agents in the cluster where agent 1 resides at present), we determine the solution to the following recursion

$$V(\mathbf{a}(\hat{\theta}), \tilde{\theta}_1) = v_1(\mathbf{a}, \tilde{\theta}_1) + \delta \mathbb{E} \left\{ V(\mathbf{a}(\hat{\Theta}_1, \hat{\Theta}_{-1}), \tilde{\Theta}_1) \right\} \quad (10)$$

to get function  $V(\cdot, \cdot)$ , where  $\tilde{\theta}_1$  is an arbitrary state variable,  $\tilde{\Theta}_1$  follows (2) (setting  $\theta_i = \tilde{\theta}_i$ ),  $\mathbf{a}$  is an arbitrary allocation, the B2D term is generated independently, and  $\hat{\Theta}_{-1}$  is chosen using the mean-field distribution. Notice that  $\tilde{\theta}_1 = \theta_1$  would yield the true value of allocation  $\mathbf{a}$  to agent 1. By the cluster optimal allocation (what the cluster actually does), agent 1 gets  $V(\mathbf{a}^*(\hat{\theta}_1, \hat{\theta}_{-1}), \theta_1)$  whereas from the perception of agent 1 he thinks he should be getting  $V(\tilde{\mathbf{a}}(\hat{\theta}_1, \hat{\theta}_{-1}), \theta_1)$  (based on what he thinks the cluster should be doing).

We will use the different value functions to define the transfer for agent 1 depending on the reported state variable  $\hat{\theta}_1$  such that the transfer depends on the difference between what he gets from the system optimal allocation and what he expects the system to do from his own perspective. Using this logic we set the transfer for agent 1 as

$$\begin{aligned} p^*(\hat{\theta}_1, \hat{\theta}_{-1}) = V(\mathbf{a}^*(\hat{\theta}), \hat{\theta}_1) - V(\tilde{\mathbf{a}}(\hat{\theta}), \hat{\theta}_1) \\ + C(\hat{\theta}_{-1}) - (\tilde{W}(1, (\hat{\theta}_1, \hat{\theta}_{-1})) - V(\tilde{\mathbf{a}}(\hat{\theta}), \hat{\theta}_1)). \end{aligned} \quad (11)$$

where  $C(\hat{\theta}_{-1})$ , following the Groves pivot mechanism, can be chosen using the recursion

$$C(\hat{\theta}_{-1}) = \min_{\mathbf{a}_{-1}} \sum_{i \neq 1} v_i(\mathbf{a}_{-1}, \hat{\theta}_i) + \delta \mathbb{E} \left\{ C(\hat{\Theta}_{-1}) \right\}, \quad (12)$$

where  $\hat{\Theta}_{-1} \sim [\otimes \rho^{M-1}, \otimes \zeta^{M-1}]$ , and  $\mathbf{a}_{-1}$  is used to denote an allocation in a system in which agent 1 is not present.

The Groves pivot mechanism idea ensures that the payoff of agent 1,  $V(\mathbf{a}^*(\hat{\theta}), \hat{\theta}_1) - p^*(\hat{\theta}_1, \hat{\theta}_{-1})$ , is equal  $\tilde{W}(1, (\hat{\theta}_1, \hat{\theta}_{-1})) - C(\hat{\theta}_{-1})$ . This is simply the value of the system as a whole from the viewpoint of agent 1, minus a function only of  $\hat{\theta}_{-1}$ . As in the Groves mechanism, such a payoff naturally promotes truth-telling as a dominant strategy at each step.

The family of transfers defined achieve two important functions: first, they translate the system optimal viewpoint to what the typical agent perceives the system optimal to be which is a novel feature of the mean-field model (the first two terms achieve this); and second, as in the incentive compatible dynamic mechanism of [8], they account for the



penalty the remaining agents pay for presence of the agent in question, again using the typical agent's perception of the system optimal policy (the last three terms achieve this).

#### Allocation Scheme

The basic building block of our mechanism is the per-frame optimal allocations that solve (1). We will now spell out the allocation in greater detail. First, we observe that the allocation problem separates into independent allocation problems in each cluster that have the same basic structure. Therefore, it suffices to discuss the allocation problem for one cluster.

From (7), the objective in this cluster is

$$\min_{\mathbf{a}} \sum_{i=1}^M c((d_i[k-1] + \eta - \chi_i(\mathbf{a}[k], \theta_i[k]))^+) \quad (13)$$

An optimal allocation is determined using the following observations. First, we partition the agents into two sets, ones who cannot decode the frame even if they never transmit during the  $T$  slots of the D2D phase and the rest; the former agents are made to transmit first. After this we determine agents who have extra chunks (number of slots that they can transmit on such that there is still time to decode whole frame) and make these agents transmit their extra chunks. After all the extra chunks have been transmitted, it is easy to see using the properties of the holding cost function that agents are made to transmit in a minimum-deficit-first fashion in order to prioritize agents with large deficits. This is summarized in the following lemma.

*Lemma 1:* The algorithm delineated in Algorithm 1 provides an optimal greedy allocation.

---

#### Algorithm 1 Optimal Mean Field D2D Allocation Rule

---

At the beginning of each frame  $k-1$ , given the arrivals  $(e_1[k], \dots, e_M[k])$ :

Partition the devices into sets  $\mathcal{S} = \{i \in \{1, \dots, M\} : N - e_i[k] \leq T, e_i[k] + \sum_{j \neq i} e_j[k] \geq N\}$  and  $\mathcal{S}^c$ .

If  $\mathcal{S} = \emptyset$ , none of the agents can decode the block. Else,

**Phase 1)** Let all the agents in  $\mathcal{S}^c$  transmit all that they initially received for the next  $T_1 = \min\{\sum_{i \in \mathcal{S}^c} e_i[k], T\}$  slots.

If there exists time and a need for more transmissions,

**Phase 2)** Let each agent  $i \in \mathcal{S}$  transmit up to  $e_i[k] + T - N$  of its initial chunks.

**Phase 3)** While there exists time and a need for more transmissions, let devices in  $\mathcal{S}$  transmit their remaining chunks in an increasing order of their deficit values.

---

## IV. PROPERTIES OF MECHANISM

### A. Truth-telling as dominant strategy

Since we consider a mean-field setting, we will assume that deficit of agent  $i$  changes via the allocation while the deficits of all the other agents are drawn using the given distribution  $\rho$ . The  $\mathbf{e}$  values are generated *i.i.d.* with distribution  $\zeta$ . Based on the system state report  $\theta[k]$  at time  $k$ , we assume that the mechanism makes the optimal greedy allocation  $\mathbf{a}^*[k]$  from (7)

and levies transfers  $\mathbf{p}^*[k]$  from (11) that uses the allocations from the agent's perspective from (9). We can then show that truthfully revealing the state, *i.e.*,  $(d, e)$  values at the beginning of every frame is incentive compatible.

*Theorem 2:* Our mechanism  $\{\mathbf{a}^*[k], \mathbf{p}^*[k]\}_{k=0}^{\infty}$  is incentive compatible.

### B. Value functions and optimal strategies

Given that truth-telling is a dominant strategy, we will henceforth assume that it is in force and now show that the value function given by the solution to (6) is well-defined and can be obtained using value iteration. Similarly, we will show that both the value function and the optimal allocation policy from a agent's perspective, given by (8) and (9) respectively, exist and can also be determined via value iteration.

Define operators  $T_1$  and  $T_2$  by

$$T_1 w(\theta) = \sum_{i=1}^M v_i(\mathbf{a}^*(\theta), \theta_i) + \delta \mathbb{E}\{w(\Theta)\} \quad (14)$$

$$T_2 \tilde{w}(1, (\theta_1, \theta_{-1})) = \min_{\mathbf{a}} \sum_{i'=1}^M v_{i'}(\mathbf{a}, \theta_{i'}) + \delta \mathbb{E}\{\tilde{w}(1, (\Theta_1, \Theta_{-1})) | \mathbf{a}, \theta_1\} \quad (15)$$

using (6) and (8), respectively.

*Theorem 3:* The following hold:

- 1) There exists a unique  $W(\theta)$  such that  $T_1 W(\theta) = W(\theta)$ , and given  $\theta$  for every  $\mathbf{w} \in \mathbb{R}_+^M$ , we have  $\lim_{n \rightarrow \infty} T_1^n \mathbf{w} = W(\theta)$ ;
- 2) There exists a unique  $\tilde{W}(1, (\theta_1, \theta_{-1}))$  such that  $T_2 \tilde{W}(1, (\theta_1, \theta_{-1})) = \tilde{W}(1, (\theta_1, \theta_{-1}))$ , and given  $(\theta_1, \theta_{-1})$  for every  $\mathbf{w} \in \mathbb{R}_+^M$ , we have  $\lim_{n \rightarrow \infty} T_2^n \mathbf{w} = \tilde{W}(1, (\theta_1, \theta_{-1}))$ ; and
- 3) The Markov policy  $\tilde{\mathbf{a}}((\theta_1, \theta_{-1}))$  obtained from (9) is an optimal policy to be used in cluster  $j_1[\cdot]$  from the viewpoint of agent 1.

## V. MEAN FIELD EQUILIBRIUM

In the mean-field setting, assuming the state of every other agent is drawn *i.i.d.* with distribution  $\rho \times \zeta$ , the deficit of any given agent evolves as a Markov chain. We start by showing that this Markov chain has a stationary distribution. If this stationary distribution is the same as  $\rho$ , then the distribution  $\rho$  is defined as a mean-field equilibrium (MFE); we use the Schauder fixed point theorem to show the existence of a fixed point  $\rho$ . Using the regenerative representation of the stationary distribution of deficits given  $\rho$  and a strong coupling result, we prove that the mapping that takes  $\rho$  to the stationary distribution of deficits is continuous using a strong coupling result. Finally, we show that the set of probability measures to be considered is convex and compact so that existence follows.

### A. Stationary distribution of deficits

Fix a typical agent  $i$  and consider the state process  $\{d_i[k]\}_{k=-1}^{\infty}$ . This is a Markov process in the mean-field setting: if there is no regeneration, then the deficit changes as per the allocation and the number of B2D packets received,

and is chosen via the regeneration distribution otherwise. The allocation is a function of the past  $d_i$ , the number B2D packets received and the state of the other agents. The number of B2D packets received and the state of the other agents are chosen *i.i.d* in every frame. This Markov process has an invariant transition kernel. We construct it by first presenting the form given the past state and the allocations, namely,

$$\begin{aligned} \mathbb{P}(d_i[k] \in B | d_i[k-1] = d, e_i[k] = e, \mathbf{a}) \\ = \delta 1 \left\{ (d + \eta_i - \chi_i(\mathbf{a}, (d, e)))^+ \in B \right\} + (1 - \delta) \Psi(B), \end{aligned} \quad (16)$$

where  $B \subseteq \mathbb{R}^+$  is a Borel set and  $\Psi$  is the density function of the regeneration process for deficit. In the above expression, the first term corresponds to the event that agent  $i$  can either decode the packet using D2D transmissions or not, and the second term captures the event that the agent regenerates after frame  $k$ .

*Lemma 4:* The Markov chain where the allocation is determined using (7) based on choosing the states of all users other than  $i$  *i.i.d* with distribution  $\rho \times \zeta$  and the number of B2D packets of user  $i$  independently with distribution  $\zeta$ , and the transition probabilities in (16) is positive Harris recurrent and has a unique stationary distribution.

We denote the unique stationary distribution for the deficit of a typical agent by  $\Pi_{\rho \times \zeta}$ ; the dependence on  $\Psi$  is suppressed.

#### B. Agent and cluster decision problems

Suppose that each agent has common information about the distribution for the deficit  $\rho \in \mathcal{M}_1(\mathbb{K})$  (where  $\mathcal{M}_1(\mathbb{K})$  is the set of probability measures on  $\mathbb{K}$ ); this is one of the mean-field assumptions. We further assume that  $\rho \in \mathcal{P}$  where

$$\mathcal{P} = \{ \rho | \rho \in \mathcal{M}_1(\mathbb{K}) \text{ with finite mean} \}. \quad (17)$$

We will also assume that the regeneration distribution  $\Psi \in \mathcal{P}$ . From Section IV, the best strategy for each agent is to truthfully reveal its state based on the transfers suggested in each frame as per (11). Then each cluster simply maximizes the system value function by choosing the greedy optimal allocation based on (7).

#### C. Mean field equilibrium

Given the distribution for deficit  $\rho$  and the station distribution  $\Pi_{\rho \times \zeta}$ , we have the following definition.

*Definition 1:* (Mean field equilibrium). Let  $\rho$  be the common cumulative distribution for deficit with truth-telling-truth being optimal for each agent in every frame. Then, we say that the given  $\rho$  and truth-telling constitutes a mean field equilibrium if

$$\rho(d) = \Pi_{\rho \times \zeta}(d), \forall d \in \mathbb{K} \quad (18)$$

#### VI. EXISTENCE OF MFE

The main result showing the existence of MFE is as follows.

*Theorem 5:* There exists an MFE of  $\rho$  and truth-telling policy such that  $\rho(d) = \Pi_{\rho \times \zeta}(d)$ ,  $\forall d \in \mathbb{K}$ .

As mentioned earlier, we will be specializing to the space  $\mathcal{M}_1(\mathbb{K})$ , its subset  $\mathcal{P}$  and further subsets of  $\mathcal{P}$ . The primary topology on  $\mathcal{M}_1(\mathbb{K})$  that we will consider is the uniform

norm topology, i.e., using the  $l_\infty$  norm given by  $\|\rho\| = \max_{d \in \mathbb{K}} \rho(d)$ . Another topology on  $\mathcal{M}_1(\mathbb{K})$  that we will use is the point-wise convergence topology, i.e.,  $\{\rho_n\}_{n=1}^\infty \subset \mathcal{M}_1(\mathbb{K})$  converges to  $\rho \in \mathcal{M}_1(\mathbb{K})$  point-wise if  $\lim_{n \rightarrow \infty} \rho_n(d) = \rho(d)$  for all  $d \in \mathbb{K}$ ; it is easily verified that the convergence is the same as weak convergence of measures. Also, define the mapping  $\Pi^*$  that takes  $\rho$  to the invariant stationary distribution  $\Pi_{\rho \times \zeta}(\cdot)$ . Let  $\mathcal{P}' \subset \mathcal{P}$ . We will use the Schauder fixed point theorem to prove existence which is given as follows.

*Theorem 6:* (Schauder Fixed Point Theorem). Suppose  $\mathcal{F}(\mathcal{P}') \subset \mathcal{P}'$ ,  $\mathcal{F}$  is continuous and  $\mathcal{F}(\mathcal{P}')$  is contained in a convex and compact subset of  $\mathcal{P}'$ , then  $\mathcal{F}$  has a fixed point.

Note that from the definition of  $\mathcal{P}$ , it is already convex. Then in the following section, we will prove that under the topology generated by the uniform norm,  $\Pi^*$  is continuous and the image of  $\Pi^*$  for a specific subset  $\mathcal{P}'$  is pre-compact.

#### Steps to Prove MFE Existence

We first need to prove the continuity of  $\Pi^*$  with the uniform norm topology. For this we will start by showing that for any sequence  $\rho_n \rightarrow \rho$  with  $\rho_n, \rho \in \mathcal{P}$  in uniform norm,  $\Pi^*(\rho_n) \Rightarrow \Pi^*(\rho)$  (where  $\Rightarrow$  denotes weak convergence). Finally, using some properties of  $\mathcal{M}_1(\mathbb{K})$  we will strengthen the convergence result to prove that  $\Pi^*(\rho_n) \rightarrow \Pi^*(\rho)$  in uniform norm too.

1) *Continuity of the mapping  $\Pi^*$ :* We will restrict our attention to subset of probability measures  $\mathcal{P}(E) \subset \mathcal{M}_1(\mathbb{K})$  such that

$$\mathcal{P}(E) = \left\{ \rho \in \mathcal{M}_1(\mathbb{K}) : \sum_{d \in \mathbb{K}} d \rho(d) \leq E \right\} \quad (19)$$

where  $E$  is a given non-negative constant; in other words, probability measures with a specified bound on the mean and not just a finite mean. We will assume that the regeneration distribution  $\Psi \in \mathcal{P}(E')$  for some  $E'$ . Later on we will specify the values of  $E$  and  $E'$  to be used.

We start with the following preliminary result that establishes compactness of sets like  $\mathcal{P}(E)$  in the uniform norm topology; note that convexity is immediate.

*Lemma 7:* The set of probability measures  $\mathcal{P}(E)$  on  $\mathbb{K}$  is a compact set of  $l_\infty$  for every  $E \in \mathbb{R}_+$ .

Next we present a coupling result based on results presented by Thorisson [14] (see chapter 1). This result will be used in proving continuity of the stationary distribution of the deficit process under the topology of point-wise convergence and in strengthening the convergence result.

*Theorem 8:* Let  $\{\rho_n\}_{n=1}^\infty \in \mathcal{M}_1(\mathbb{K})$  converge weakly to  $\rho \in \mathcal{M}_1(\mathbb{K})$ , then there exists a coupling, i.e., random variables  $\{X_n\}_{n=1}^\infty$ ,  $X$  on a common probability space and a random integer  $N$  such that  $X_n \sim \rho_n$  for all  $n \in \mathbb{N}$ ,  $X \sim \rho$  and  $X_n = X$  for  $n \geq N$ .

This result shows that weak convergence of probability measures on  $\mathbb{K}$  is equivalent to convergence of probability measures in total variation norm, and hence, also in uniform norm.

Next we show that  $\Pi_{\rho \times \zeta} \in \mathcal{P}(E)$  whenever  $\rho \in \mathcal{P}(E)$ .

*Lemma 9:* If  $\rho \in \mathcal{P}(E)$  for  $E \geq \frac{\delta \eta}{1 - \delta}$  and the regeneration distribution  $\Psi \in \mathcal{P}(E')$  for  $E' \leq E - \frac{\delta \eta}{1 - \delta}$ , then the stationary

distribution of the deficit process of any specific user  $\Pi_{\rho \times \zeta} \in \mathcal{P}(E)$ .

Next we show continuity properties of the mapping  $\Pi^*$ .

*Theorem 10:* The mapping  $\Pi^* : \mathcal{P}(E) \mapsto \mathcal{P}(E)$  is continuous in the uniform topology. In addition,  $\Pi^*$  has a fixed point in  $\mathcal{P}(E)$ .

## VII. VALUE DETERMINATION

We now turn to computing the system value from the viewpoint of agent 1. Here, we suppose there are  $M = 4$  agents in each cluster, and all have  $\eta = 0.95$ ,  $\delta = 0.9995$ . Hence, each agent spends an average of 2000 frames in the system before leaving. A new agent has a deficit drawn uniformly at random from the interval  $[0, 13]$ . Each agent needs to receive  $N = 10$  packets to decode the block, and there are  $T = 8$  time slots in each frame. We wish to determine the value function from the perspective of the cluster and from the perspective of agent 1, using (6) and (8).

The following observation is useful to determine the allocations  $\mathbf{a}^*$  and  $\tilde{\mathbf{a}}$ . It is straightforward to find  $\mathbf{a}^*$ , since it simply follows Algorithm 1. Now, consider  $\tilde{\mathbf{a}}$ . It is simple to see that it too would follow Phases 1 and 2 of Algorithm 1. Then, from the perspective of agent 1, after the completion of these two phases, there are only two classes of allocations—those in which he transmits and those in which he does not. Now, since all the other agents that agent 1 comes in contact with in the future are drawn from  $[\otimes \rho^{M-1}, \otimes \zeta^{M-1}]$ , the allocation should follow a greedy minimization with respect to the other agents. Thus, we only need consider two allocations while conducting value iterations: min-deficit-first with agent 1 (identical to Phase 3 of Algorithm 1) and min-deficit-first without agent 1 (just set aside agent 1 in Phase 3 of Algorithm 1).

We first run the system according to Algorithm 1, and use the results to find the empirical deficit distribution, denoted by  $R$ . This is identical to the Mean Field deficit distribution. The empirical distribution of deficit  $R$ , is shown in Figure 4. We find that deficit lies in the range  $0 - 13$ .

With  $\eta = 0.95$ , the (countable) deficit set is  $\{0, 0.05, 0.1, 0.15, \dots\}$ . With a deficit range of  $0 - 13$ , there are totally 260 potential values for deficit. For the number of B2D chunks received  $e$ , we take values 3, 4 and 5 (uniformly). Therefore, there are totally  $260^4 \times 3^4$  states in the system. Using  $R$  to represent the MF deficit distribution, and a linear holding cost function, we run value iteration; we present an example for a few states in Figure 5. We thus obtain the mean field value functions.

The empirical distribution of the average discounted transfers over the lifetime of each device is shown in Figure 6. The average transfer is 18039. We will discuss the economic implications of this observation after describing the Android experiments in the next section.

## VIII. ANDROID IMPLEMENTATION

We now describe experiments on an Android testbed using a cluster size of four Google Nexus 7 tablets. We modified the

kernel of Android v 4.3 to simultaneously allow both WiFi and 3G interfaces to transmit and receive data.

Our system consists of a server application on a desktop that codes data and sends it to the tablets over the Internet, an Android app that receives data over Internet on a 3G interface and shares it over the WiFi interface, and a monitor that keeps track of the state of the system and generates a trace of events. The server initializes each tablet in the system with a randomly selected number of chunks. Additionally, churn is emulated in the system by making the application on the tablet reset randomly with a probability  $\bar{\delta} = 5 \times 10^{-4}$  (*i.e.*,  $\delta = 0.9995$ ).

We set the frame duration as 500 ms. Since we have  $\delta = 0.9995$ , this means that the average duration that a device spends in the system is 1000 seconds. We use an MP3 music file as the data, and divide it into blocks, with the blocks being further divided into chunks. Chunks are generated using an open source random linear coding library [15], using field size 256 and 10 degrees of freedom per block. Hence, a block is decodable with high probability if 10 chunks are received successfully. Each chunk has an average size of 1500 Bytes, and has a header that contains the frame number it corresponds to as well as its current deficit. The system maintains synchronization by observing these frame numbers.

The allocation algorithm proceeds as suggested by Algorithm 1. We approximate the three phases by setting back-off times for D2D access. Devices that cannot complete (*i.e.*, Phase 1 devices) should be the most aggressive in D2D channel access. We set them to randomly back-off between 1 and 5 ms before transmission. Devices that can afford to transmit some number of chunks (Phase 2) should be less aggressive, and transmit chunks by backing off between 1 and 15 ms. Finally, each device enters Phase 3, and modulates its aggressiveness based on deficits. Each device normalizes its deficit based on the values of deficits that it sees from all other transmissions, and backoff proportional to this deficit within the interval of 5 to 15 ms. The average error in value due to a back-off based implementation is about 10 – 15%.

We conducted experiments to determine the stable delivery ratio achieved using D2D for different B2D initializations per frame. We present some sample deficit trajectories in Figure 7. The random resets emulating peer churn are visible as sharp changes in the deficit. We found that on average, B2D transfer of 4 chunks to each device is sufficient to ensure a delivery ratio of over 0.95. Hence, it is easy to achieve a 60% reduction in B2D usage, while maintaining a high QoE.

## IX. SYSTEM VIABILITY

We saw in Section VII that the average transfer to each agent is positive, meaning that the agents need to obtain some kind of subsidy in order to use the system. What kind of subsidy should they be given? The Android experiments indicate that each agent is able to save 60% of the B2D costs when participating in the system. Would this be sufficient?

The price of B2D service is currently \$10 per GB across many US cellular providers. Suppose that we consider music streaming at a rate of 250 kbps corresponding to our Android



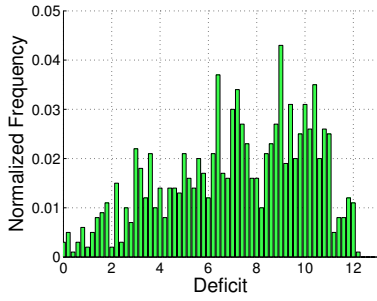


Fig. 4. Deficit distribution.

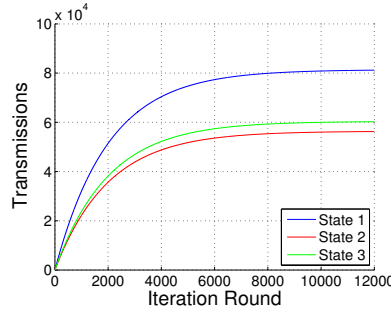


Fig. 5. Convergence of value iteration.

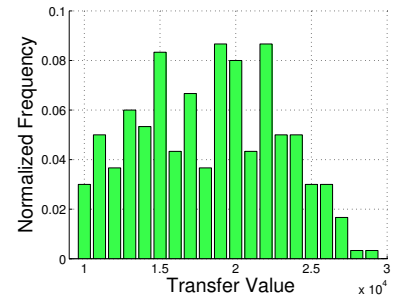


Fig. 6. Transfer distribution.

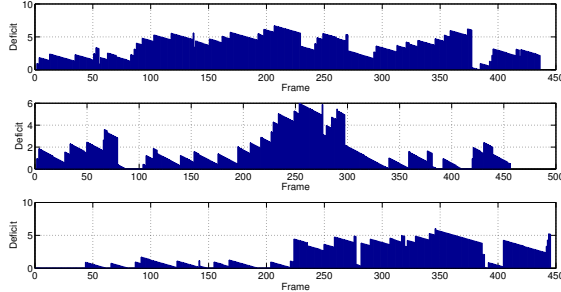


Fig. 7. Sample deficit trajectories. We have used  $\delta = 0.98$  in this run to illustrate frequent resets, which cause sharp decreases or increases.

system. If each device uses only B2D communication (no D2D at all), the cost of spending 1000 seconds in the system is 31.25 cents. The per frame communication cost is 0.0156 cents, and we can consider this to be the value of each frame to the agent.

The experiments in Section VIII indicate that the agents have to utilize their B2D connection for at least 40% of the chunks to maintain the desired QoE. Hence, the value that can potentially be received by participating in the D2D system is  $0.6 \times 0.0156 = 0.00936$  cents per frame. Let us assume a linear deficit cost function that takes a value of 0.00936 cents at deficit value of 15. In other words, if the agent were to experience a deficit of 15 or above in a frame, it gets no payoff from that frame. Using this linear transformation, we can translate the average transfer of 18039 (the value found in Section VII) over the entire 1000 seconds into a total of 11.26 cents. Thus, if each agent saves at least 11.26 cents, it has an incentive to participate in the D2D system. The actual saving is  $0.6 \times 31.25 = 18.75$  cents (60% of the B2D costs) per agent, which is well above the minimum required saving.

The situation is still better for video streaming at a rate of 800 kbps. A similar calculation indicates that a 16 minute video costs about \$1 using pure B2D, while the B2D cost in the hybrid system is only 40 cents, yielding a savings of 60 cents per agent. However, a savings of about 36 cents per agent is all that is needed to incentivize them to participate.

In a full implementation, each agent would place an amount (eg. 36 cents for a 16 minute average lifetime) in escrow with the monitor upon connecting. Each agent would receive transfers according to our mechanism, and, on average, would receive its amount back from the monitor for its contributions.

Hence, the system would also be budget balanced overall.

## X. CONCLUSION

We studied the problem of providing incentives for cooperation in large scale multi-agent systems, using wireless streaming networks as an example. The objective was to incentivize truth telling about individual user states so that a system wide cost minimizing allocation can be used. We showed how a mean field approximation for large systems yields a low-complexity framework under which to design the mechanism. We implemented the system on Android devices and presented results illustrating its viability using the current price of cellular data access as the basis for transfers.

## REFERENCES

- [1] B. Cohen, "Incentives build robustness in BitTorrent," in *Proc. WEIS*, Jun. 2003.
- [2] C. Akerly and R. Johari, "A peer-to-peer system as an exchange economy," in *Proc. GameNets*, Oct. 2006.
- [3] T. Yu, Z. Zhou, D. Zhang, X. Wang, Y. Liu, and S. Lu, "INDAPSON: An incentive data plan sharing system based on self-organizing network," in *IEEE Infocom*, 2014.
- [4] N. Abedini, S. Sampath, R. Bhattacharyya, S. Paul, and S. Shakkottai, "Realtime streaming with guaranteed QoS over wireless D2D networks," in *Proc. of ACM MOBIHOC 2013*, Bangalore, India, July 2013.
- [5] S. Deb, M. Médard, and C. Choute, "Algebraic gossip: A network coding approach to optimal multiple rumor mongering," *IEEE Trans. on Information Theory*, vol. 52, no. 6, pp. 2486–2507, 2006.
- [6] I. Hou, V. Borkar, and P. Kumar, "A theory of QoS for wireless," in *IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, April 2009.
- [7] I.-H. Hou, Y. Liu, and A. Sprintson, "A non-monetary protocol for peer-to-peer content distribution in wireless broadcast networks with network coding," in *Proceedings of WiOpt*, May 2013.
- [8] D. Bergemann and J. Välimäki, "The dynamic pivot mechanism," *Econometrica*, vol. 78, no. 2, pp. 771–789, 2010.
- [9] V. Krishna, *Auction Theory*. Academic Press, MA, U.S.A., 1997.
- [10] J. M. Lasry and P. Lions, "Mean field games," *Japanese Journal of Mathematics*, vol. 2, no. 1, pp. 229–260, 2007.
- [11] K. Iyer, R. Johari, and M. Sundararajan, "Mean field equilibria of dynamic auctions with learning," *ACM SIGecom Exchanges*, vol. 10, no. 3, pp. 10–14, 2011.
- [12] M. Manjrekar, V. Ramaswamy, and S. Shakkottai, "A mean field game approach to scheduling in cellular systems," in *Proceedings of IEEE Infocom*, Toronto, Canada, 2014.
- [13] J. Li, R. Bhattacharyya, S. Paul, S. Shakkottai and V. Subramanian, "Incentivizing Sharing in Realtime D2D Streaming Networks: A Mean Field Game Perspective," Tech. Report available at <http://www.ece.tamu.edu/~sshakkot/d2d-incentives.pdf>.
- [14] H. Thorisson, "Coupling methods in probability theory," *Scandinavian Journal of Statistics*, pp. 159–182, 1995.
- [15] "Network coding utilities," library available at <http://arni.epfl.ch/software>.