Learning Relaying Strategies in Cellular D2D Networks with Token-Based Incentives

Nicholas Mastronarde, Viral Patel
Department of Electrical Engineering
State University of New York at Buffalo
Buffalo, NY USA

Jie Xu, Mihaela van der Schaar
Department of Electrical Engineering
University of California at Los Angeles
Los Angeles, CA USA

***Abstract*—We consider a cellular network where intelligent cellular devices owned by selfish users are incentivized to cooperate with each other by using tokens, which they exchange electronically to "buy" and "sell" downlink relay services, thereby increasing the network's capacity. We endow each device with the ability to learn its optimal cooperation strategy *online* in order to maximize its long-term utility in the dynamic network environment. We investigate the impact of the token exchange system on the overall downlink network performance and the performance of individual devices in various deployment scenarios involving mixtures of high and low mobility users. Our results suggest that devices have the greatest incentive to cooperate when the network contains many highly mobile users (e.g., users in motor vehicles). Moreover, within the token system, devices can effectively learn to cooperate online, and achieve over 20% higher throughput on average than with direct transmission alone, all while *selfishly* maximizing their own utility.**

Keywords—D2D cooperative relaying; token exchange system; online learning; LTE-Advanced

I. INTRODUCTION

Cooperative device-to-device (D2D) relaying can increase the capacity of cellular networks [1] and has the potential to significantly improve the performance of important applications such as multiuser wireless video streaming [2][3]. For this reason, cooperative communication and D2D technologies are central to the LTE-Advanced standard, which aims to address an anticipated exponential increase in mobile data over the next few years. However, most existing work on cooperative transmission assumes that all network devices will act as relays whenever requested. Unfortunately, this assumption does not hold when devices are cell-phones/laptops/tablets that are owned by self-interested users, which aim to maximize their own utility. In this situation, devices may not be willing to use their limited battery energy to relay other devices' data without any direct benefit to themselves. Therefore, to stimulate cooperation among self-interested devices, and realize the full potential of cooperative relaying, proper incentives must be designed into the system.

One way to incentivize cooperation is to introduce a "payment" to reward the self-interested devices when they act as relays and introduce a "charge" to the devices that receive help from relays. To this end, electronic tokens have been proposed [4][5]. The key idea is that, each time a device receives help from a relay, it pays the relay with a token, which the relay can use to get relay service in the future. As long as the expected future benefit of having an additional token outweighs the immediate cost of relaying, then a self-interested device will be willing to act as a relay [5].

In this paper, we adopt a token-based approach similar to [5]. However, unlike in [5], where the focus is on designing token-based incentive schemes from a system perspective, in this paper, we investigate how the devices can *learn* to adapt their wireless cooperation and token gathering strategies online, at run-time, based on their experience. Specifically, we study how an *individual user* in the system can *learn* its optimal (best response) cooperation strategy online, and how this strategy impacts and is impacted by the other heterogeneous devices in the network, which are simultaneously learning.

There is a lot of great literature on cooperative relaying in wireless and cellular networks (e.g., [1]-[3][6][7][10][12]). However, these works do not consider selfish users that require incentives to cooperate, which is the focus of this paper. Importantly, many of the techniques and solutions in the existing literature on cooperative relaying can be implemented in conjunction with our proposed framework.

Our contributions are as follows:

- We model each device's decision problem, i.e., whether or not to relay, as a Markov decision process (MDP). The objective of a device is to maximize its long-term utility, which is defined as the difference between the benefit it gains when it receives data through a relay and the energy cost it incurs when it relays data for another device.
- We propose a low complexity learning algorithm that enables each individual device to optimize its long-term utility in the dynamic network environment.
- We systematically evaluate the system performance in various deployment scenarios involving both high mobility and low mobility users. Our simulation results suggest that devices have the greatest incentive to cooperate when the network contains many highly mobile users (e.g., users in motor vehicles).

The remainder of this paper is organized as follows. In Section II, we present the system model and formulate an individual device's optimization problem. In Section III, we propose a simple and effective algorithm that an individual device can deploy to learn the optimal cooperation strategy online. In Section IV, we present our simulation results. We conclude in Section V.

978-1-4799-2851-4/13/$31.00 ©2013IEEE

II. SYSTEM MODEL

A. Cooperative Downlink Network Model

We consider a cellular network comprising N mobile transceiver devices.[1] We assume that time is slotted into discrete time intervals indexed by $t \in \mathbb{N}$. At any given time t, a fraction of the devices need to receive data from the base station; however, some of these devices may experience bad channel conditions (e.g., due to fading or shadowing), which will limit their received data rate. In this situation, intermediate devices can act as device-to-device relays to help *deliver the data from the base station to the destination device*.

We use an Amplify-and-Forward (AF) cooperation scheme [12] and a simple algorithm for deciding when a device will request a relay and which relay it will select. Importantly, the token system and learning solution proposed in this paper are *not* dependent on these specific details: in fact, they can be applied to Decode-and-Forward (DF) cooperation schemes and any other relay selection strategies that use one relay at a time for each cooperative link, e.g., [6][7].[2]

Outbound relay demand rate: Suppose that in slot t, device $j \in \{1,\ldots,N\}$ wants to receive data from the base station (device 0). We assume that device j has a target receive rate $r^{j,\text{target}}$ that is achievable at a target Signal-to-Interference-and-Noise Ratio (SINR) $\Gamma^{j,\text{target}}$. Assuming that the source power is fixed at P^0, there is a non-zero probability λ_t^j that the device j will not achieve its target rate in slot t: i.e., the outage probability

$$\lambda_t^j = \Pr(r_t^{0j} < r^{j,\text{target}}) = \Pr(\Gamma_t^{0j} < \Gamma^{j,\text{target}}), \quad (1)$$

where Γ_t^{0j} is the SINR of the channel between the base station and device j, and r_t^{0j} is the corresponding receive rate for device j. If device j is unable to receive its target rate during slot t, then it will attempt to find a relay through which it can achieve its target rate. For this reason, we refer to λ_t^j as device j's instantaneous *outbound relay demand rate* (ORDR). Note that a device's instantaneous ORDR is unknown a priori because it depends on the device's geographic location, its distance from the nearest base station, and the underlying network conditions. Moreover, since devices are mobile, their instantaneous ORDRs are time-varying. For convenience, in the remainder of this section, we model device j's ORDR as a fixed value λ^j; however, in Section III, we assume that it is time-varying and propose a learning algorithm that enables the device to dynamically adapt as its ORDR changes over time.

Illustrative relay selection strategy: If device j is unable to receive its target rate during slot t, then it attempts to find a relay through which it can achieve its target rate. Following standard relay channel analysis for AF cooperation [12], we obtain the following received SINR over the cooperative link:

$$\Gamma_t^{0ij} = \frac{\Gamma_t^{0i}\Gamma_t^{ij}}{\Gamma_t^{0i} + \Gamma_t^{ij} + 1}, \quad (2)$$

where $i \in \{1,\ldots,N\}$ denotes the relay device and Γ_t^{0i} and Γ_t^{ij} are the SINRs of the source-relay and relay-destination channels, respectively. Assuming that the relay transmission power is P^i, device j selects the relay i^* that can meet the target SINR while using the least power: i.e.,

$$i^* = \arg\min_{i \in \text{cell}(j)} \{P^i : \Gamma_t^{0ij} \geq \Gamma^{j,\text{targ}}\}, \quad (3)$$

where $\text{cell}(j)$ is the set of devices that are within range of the same base station as device j. If there is no relay for which $\Gamma_t^{0ij} \geq \Gamma^{j,\text{targ}}$ that is also willing to provide relay services, then device j will receive its data directly from the base station at the rate $r_t^{0j} < r^{j,\text{targ}}$.

B. Token System

Providing a relay transmission costs the relay energy without providing it any immediate benefit. Consequently, without proper incentives, no transceiver will be willing to cooperate. To overcome this, we incentivize devices to cooperate through the use of tokens, which they exchange electronically to "buy" and "sell" relay services. Therefore, in order for a device to request relay service, it must have a token available to "buy" the service from another device. After a device places a relay request, a relay replies with either a positive or negative acknowledgement (ACK/NACK) [5] indicating if it is willing to relay or not (we describe how this decision is made in Section II.D). If the device's request is ACK'd, then a token transfer takes place (from the requesting device to the relaying device) and relay transmission occurs.

Several techniques that enable secure electronic token transactions have been proposed [4][11], which require no central entity. Our work assumes that such technology is in place to implement the proposed token exchange system.

C. Model of a Network Device

We now describe our model of device $j \in \{1,\ldots,N\}$.

Token holding state: At a given time, device j holds $k^j \in \mathcal{K} = \{0,1,\ldots,T\}$ tokens, where T is the total number of tokens in the network.

Battery state/energy budget: Mobile devices have limited battery energy to support their operations. We let $p_{\max}^j$ represent the energy that the user allocates to relaying data for other users. $p_{\max}^j$ can be set based on user preferences. With each relay transmission that device j provides, it expends some energy and therefore reduces its energy budget $p^j \in [0, p_{\max}^j]$. When its energy budget reaches 0, the device can neither relay nor ask for relay service. We refer to $p^j = 0$ as the *dead state*. In our problem formulation, we assume that p^j is continuous; however, we quantize p^j in our simulation results to reduce the implementation complexity. Note that the budget is for *energy spent relaying*; hence, it is not decreased when transmitting or receiving the device's own data.

[1] The proposed solution can also be applied to an ad-hoc network where there is no base station.

[2] The system can be extended to work with cooperation schemes that use multiple relays, e.g., those based on randomized space-time block coding [10].

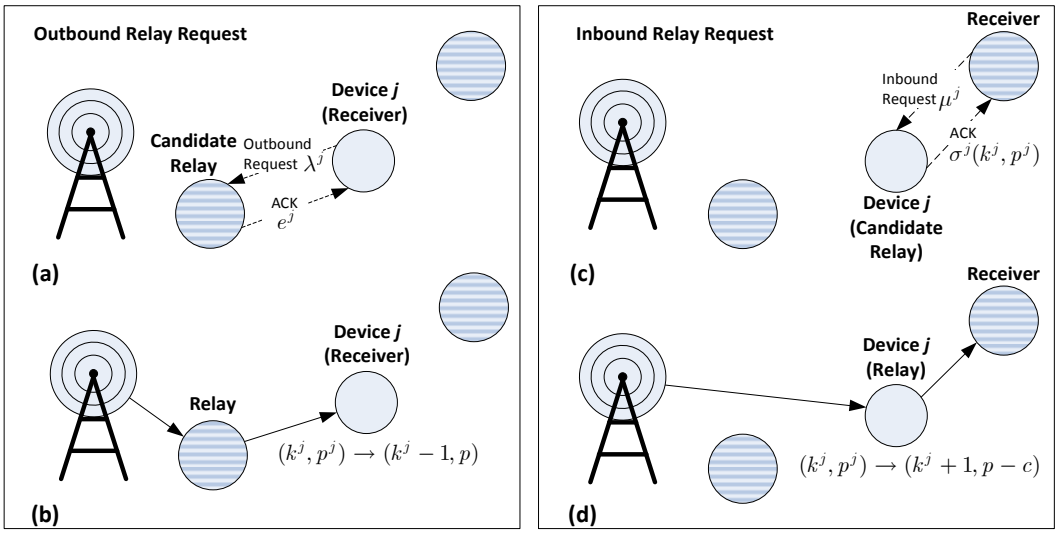

Fig. 1. Illustration of outbound and inbound relay requests in cellular network with cooperative downlink transmission. (a) Device j requests a relay with probability λ^j and its request is ACK'd with probability e^j. (b) After its request is ACK'd, device j gets help from the relay in exchange for one token. (c) Device j receives a relay request with probability μ^j and ACK's the request based on its cooperation strategy $\sigma^j(k^j, p^j)$. (d) After it ACK's the request, device j acts as a relay and expends c units of energy in exchange for one token.

Outbound relay demand rate (ORDR): λ^j denotes the j th device's ORDR as defined in (1). It should be interpreted as the probability that device j wants to use a relay to receive its data from the base station (it is not the probability that the device actually gets help from a relay). A device's ORDR is unknown a priori for the reasons described in Section II.A. The definition of the ORDR is illustrated in Fig. 1(a).

Inbound relay demand rate (IRDR): Let μ^j denote the probability that device j is asked to relay data for another device. A device's IRDR is unknown a priori because it depends on the device's geographic location, its distance from the nearest base station, the locations of other devices in the network, and the underlying network conditions. The definition of the IRDR is illustrated in Fig. 1(c).

Relay recruitment efficiency: Let e^j denote the j th device's relay recruitment efficiency, which is defined as the following conditional probability:

$$e^j = \Pr(\text{ACK recv'd} | \Gamma_t^{0j} < \Gamma^{j,\text{targ}}, k^j > 0, p^j > 0). \quad (4)$$

In words, e^j is the conditional probability that device j gets help from a relay (i.e., receives an ACK), given that it requires a relay transmission, has enough tokens to "pay" a relay, and has non-zero battery energy. It follows that, if $k^j > 0$ and $p^j > 0$, then the probability that device j actually receives an ACK from a relay in a time slot can be written as $\lambda^j e^j$. A device's relay recruitment efficiency is unknown a priori because it depends on the device's geographic location, its distance from the nearest base station, the locations of other devices in the network, the underlying network conditions, and the cooperation strategies, token holdings, and battery states of the other devices. The definition of the relay recruitment efficiency is illustrated in Fig. 1(a).

Cost and benefit: Let $b^j(r)$ be the benefit gained by device j when it receives data through a relay at data rate r and let c^j be the cost incurred by device j (i.e., energy cost) when it provides a relay transmission to another device. Let $b^j = \mathbf{E}b(r)$ be the expected benefit over all possible transmission rates. We normalize the costs to this expected benefit (i.e., by dividing the cost by b).

Cooperation actions: When the device receives a relay request, it must decide if it will ACK the request. The action set $\mathcal{A}(p^j)$ depends on the device's battery state p^j : i.e., $\mathcal{A}(p^j) = \{0,1\}$ if $p^j > 0$ and $\mathcal{A}(p^j) = \{0\}$ if $p^j = 0$, where $a^j = 1 \in \mathcal{A}(p^j)$ (ACK) means that the device is willing to act as a relay and $a^j = 0 \in \mathcal{A}(p^j)$ (NACK) means that it is not.[3] If the device is in the dead state (i.e., $p^j = 0$), then it cannot act as a relay (i.e., $\mathcal{A}(0) = \{0\}$).

Cooperation strategy/policy: Let $\sigma^j(k^j, p^j) \in \mathcal{A}(p^j)$ denote the device's cooperation strategy when it has k^j tokens and is in power state p^j. Given its IRDR and cooperation strategy, the probability that device j actually relays data for another device in a time slot is $\mu^j \sigma^j(k^j, p^j)$. The role of the cooperation strategy is illustrated in Fig. 1(c).

State evolution: Denote the device's state by $s^j = (k^j, p^j) \in \mathcal{S}$, where $k^j \in \mathcal{K} = \{0,1,...,T\}$ is its token holding state and $p^j \in [0, p_{\max}^j]$ is its battery state/energy

[3] More precisely, $\mathcal{A}(p^j) = \{0,1\}$ if $p^j > c^j$, where c^j is the amount of energy required to relay. However, to simplify the model description, we do not explicitly write this. This mismatch between model and reality is largely insignificant because it only changes the behavior of the system in the first time slot when $p^j < c^j$.

budget. When the device acts as a relay, it gains one token and loses c^j units of battery energy: i.e., $(k^j, p^j) \rightarrow (k^j + 1, p^j - c^j)$. When the device uses a relay, it loses one token and remains in the same battery state: $(k^j, p^j) \rightarrow (k^j - 1, p^j)$. The evolution of the state is illustrated in Fig. 1(b,d).

Let $P^j([k^{j'}, p^{j'}] \mid [k^j, p^j], a^j)$ denote the state transition probability function, which gives the probability of transitioning from state $s^j = (k^j, p^j)$ to state $s^{j'} = (k^{j'}, p^{j'})$ after taking cooperation decision a^j . The state transition probability function is defined as follows:

$$\begin{aligned}
&P^j([0, p^j] \mid [0, p^j], a^j) = 1 - \mu^j a^j, \\
&P^j([1, p^j - c^j] \mid [0, p^j], a^j) = \mu^j a^j, \\
&P^j([k^j - 1, p^j] \mid [k^j, p^j], a^j) = \lambda^j e^j, \\
&P^j([k^j, p^j] \mid [k^j, p^j], a^j) = 1 - \lambda^j e^j - \mu^j a^j, \\
&P^j([k^j + 1, p^j - c^j] \mid [k^j, p^j], a^j) = \mu^j a^j, \\
&P^j([k^j, 0] \mid [k^j, 0], a^j) = 1,
\end{aligned} \tag{5}$$

where $a^j \in \mathcal{A}(p^j)$ and $\mathrm{P}([k^{j'}, p^{j'}] \mid [k^j, p^j], a^j) = 0$ under all cases that are not included in (5). The first and second lines of (5) indicate that if device j has zero tokens and non-zero battery energy, then it remains in that state with probability $1 - \mu^j a^j$ (i.e., it does not provide help as a relay) or gains one token and loses c units of battery energy with probability $\mu^j a^j$ (i.e., it provides help as a relay), respectively. The third, fourth, and fifth lines of (5) indicate that if device j has non-zero tokens and non-zero battery energy, then it gets help in exchange for a token with probability $\lambda^j e^j$, remains in the same state with probability $1 - \lambda^j e^j - \mu^j a^j$ (i.e., it neither gets help nor provides help), and gains one token and loses c units of battery energy with probability $\mu^j a^j$ (i.e., it provides help), respectively. The final line of (5) indicates that device j cannot gain or lose tokens if its battery is dead (i.e., $p^j = 0$).

Expected utility: Let $u^j(k^j, p^j, a^j)$ denote the expected utility of being in state $s^j = (k^j, p^j)$ and taking cooperation action a^j : specifically,

$$u^j(k^j, p^j, a^j) = \begin{cases} -\mu^j a^j c^j, & \text{if } k^j = 0, p^j > 0 \\ \lambda^j e^j b^j - \mu^j a^j c^j, & \text{if } k^j > 0, p^j > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

In (6), when a device has non-zero tokens (i.e., $k^j > 0$), it either gets help from a relay with probability $\lambda^j e^j$ and receives benefit b^j or provides relay service with probability $\mu^j a^j$ and incurs cost c^j . If the device does not have any tokens (i.e., $k^j = 0$), it cannot request help but can provide help; therefore, it incurs cost c^j with probability $\mu^j a^j$. If the device is in a dead state (i.e., $p^j = 0$), it cannot seek or provide help.

D. Optimal Collaboration Strategy for a Single Device

We formulate the problem of determining a device's optimal cooperation strategy as an MDP [8]. The optimal *state value function* $V^j(k^j, p^j)$ represents how good it is to be in each state. It is given by the following Bellman equation:

$$V^j(k^j, p^j) = \max_{a^j \in \mathcal{A}(p^j)} \left\{ u^j\left(k^j, p^j, a^j\right) + \beta \mathbf{E}[V(k^{j'}, p^{j'})] \right\}, \tag{7}$$

Where $\mathbf{E}[\cdot]$ denotes an expectation over the next state distribution defined in (5) and β is an economic discount factor reflecting the fact that having a token now is more valuable than having one in the future. The key idea in (7) is that the optimal cooperation decision in state $s^j = (k^j, p^j)$ not only depends on the immediate utility, but also on the expected future utility. Indeed, a long-term optimization is required because, if the device only considers its immediate utility, it will never choose to act as a relay (because it will incur a cost c^j without any direct benefit to itself).

The optimal cooperation strategy/policy $\sigma^j(k^j, p^j)$ can be determined by taking the argument that maximizes (7). Assuming that the transition probability function and utility function are known a priori, the optimal value function can be computed using the well-known value iteration algorithm [8]. In practice, however, the cost and transition probability functions are unknown a priori (due to the fact that the ORDR λ^j , IRDR μ^j , and relay recruitment efficiency e^j are unknown), so device j cannot directly apply value iteration to find the optimal policy. Instead, it must learn its optimal cooperation strategy online based on experience. We discuss our proposed learning solution in Section III.

E. Balance of Outbound and Inbound Token Exchanges

On average, a device will only be able to get relay service as often as it provides relay service because it must earn as many tokens as it spends: specifically, the following condition must hold

$$\begin{aligned}
&\Pr(\text{Device } j \text{ receives inbound ACK [-1 Token]}) \\
&= \Pr(\text{Device } j \text{ sends outbound ACK [+1 Token]})
\end{aligned} \tag{8}$$

In this section, we characterize this balance after making several simplifying assumptions. First, we are only interested in the balance of token exchanges while the device has non-zero battery energy, so we assume that the power state is non-zero and fixed. Second, since the ORDR λ^j , IRDR μ^j , and relay recruitment efficiency e_j are time-varying, it is difficult to characterize the steady state behavior; hence, we assume that these parameters are fixed. Under these assumptions, (8) is equivalent to:

$$\underbrace{\lambda^j e^j \Pr(k^j > 0)}_{-1 \text{ Token}} = \underbrace{\mu^j \mathbf{E}[\sigma^j(k^j, p^j)]}_{+1 \text{ Token}}, \tag{9}$$

where the left hand side is the probability that device j uses a relay (and therefore spends one token), the right hand side is

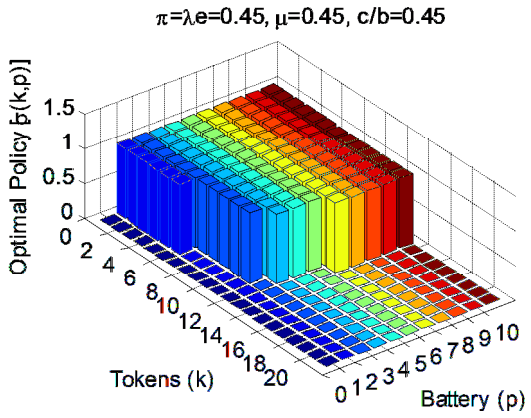

Fig. 2. Illustrative optimal cooperation policy $\sigma(k,p)$ generated using $\beta = 0.99$, $\lambda e = 0.45$, $\mu = 0.45$, and $c / b = 0.45$.

the probability that it acts as a relay (and therefore earns one token), $\Pr(k^j > 0)$ is the probability that it has at least one token, and $\mathbf{E}_{k^j}[\sigma^j(k^j, p^j)]$ is the probability that it ACK's an inbound relay request, where the expectation is taken over its steady-state token holding distribution.

F. Threshold Strategies

Device j's cooperation decision depends on its token and battery states. The optimal cooperation strategy $\sigma^j(k^j, p^j)$ is threshold in the token state k^j and the threshold depends on the battery state p^j: specifically,

$$\sigma^j(k^j, p^j) = \begin{cases} 1, & \text{if } k^j \le K_{\text{th}}(p^j), \\ 0, & \text{otherwise,} \end{cases} \qquad (10)$$

where the threshold $K_{\text{th}}(p^j)$ depends on the battery state, and $K_{\text{th}}(p^j)$ is non-decreasing in p^j. In other words, as the device's battery drains, it will want to conserve its energy and will therefore use a lower threshold. A proof of this result is left as future work; however, it has been shown in [5] that policies are threshold when there is unlimited battery energy.

A representative optimal cooperation policy is illustrated in Fig. 2. For computational reasons (i.e., to compute the optimal policy with value iteration), we quantize the battery state into eleven bins (one representing the dead state) and we limit the maximum number of tokens that the device can hold to 20.

III. Learning the Optimal Policy

All devices in the cellular network experience different environmental dynamics depending on their geographic location in the network, their distance from the nearest base station, the underlying network conditions, and, importantly, the locations, cooperation strategies, token holdings, and power states of the other network devices. These dynamics are captured by the a priori unknown and time-varying ORDR λ^j, IRDR μ^j, and relay recruitment efficiency e^j, for all $j \in \{1,\ldots,N\}$, as defined in Section II.C. In this section, we propose a hybrid offline/online learning algorithm that a device can deploy to learn the optimal collaboration strategy $\sigma^j(k^j, p^j)$ on-the-fly, despite these parameters being unknown and time-varying.

In the online phase of the proposed algorithm, device j estimates several unknown parameters (namely, its ORDR λ^j, IRDR μ^j, relay recruitment efficiency e^j, and transmission cost c^j). Then, in each time slot, device j selects its cooperation strategy based on the estimated values. In principle, device j could use these estimated values to populate the utility and transition probability functions defined in (6) and (5), respectively, and then use value iteration to compute the optimal cooperation strategy corresponding to the estimated parameters; however, recomputing the optimal policy in every time slot is computationally prohibitive.

For this reason, we propose to first compute a collection of cooperation policies *offline*, which correspond to a representative set of discretized network parameters, and then use a simple *online* table look-up to select device j's cooperation strategy in each time slot. To reduce the size of the look-up table, instead of estimating the ORDR λ^j and relay recruitment efficiency e^j independently, we directly estimate the outbound relay success rate $\pi^j = \lambda^j e^j \in [0,1/2]$.[4] Additionally, since the optimal policy is threshold in the token state as described in Section II.F, each policy can be represented compactly with only one (threshold) value per battery state. We now describe the offline and online phases in more detail.

Offline phase: Let $\Pi = \{\pi^1, \pi^2, \ldots, \pi^X\}$, $\mathrm{M} = \{\mu^1, \mu^2, \ldots, \mu^Y\}$, and $\mathcal{C} = \{c^1, c^2, \ldots, c^Z\}$ be finite sets containing representative values of the outbound relay success rate, IRDR, and relay cost, respectively. In the offline phase of the hybrid learning algorithm, we compute the collection of policies $\left\{\sigma(k, p \mid \pi^x, \mu^y, c^z) : \pi^x \in \Pi, \mu^y \in \mathrm{M}, c^z \in \mathcal{C}\right\}$. To compute the policy $\sigma(k, p \mid \pi^x, \mu^y, c^z)$, we first populate the utility and transition probability functions defined in (6) and (5), respectively, with π^x, μ^y, and c^z (in place of π^j, μ^j, and c^j). Subsequently, we use value iteration to compute the optimal cooperation strategy corresponding to the representative parameters.

Online phase: In the algorithm's online portion, the device maintains run-time estimates of the outbound relay success rate π^j, denoted by $\hat{\pi}_t^j$, and the IRDR μ^j, denoted by $\hat{\mu}_t^j$. These estimates can be made using an exponential average of successful relay transmissions and incoming relay requests, respectively. After evaluating the cost c^j required to provide a

[4] The maximum outbound relay success rate is ½. This is because, on average, a device can only use a relay as often as it relays.

relay transmission in slot t, the device follows the policy $\sigma^j(k^j, p^j \mid f(\hat{\pi}_t^j), g(\hat{\mu}_t^j), h(c^z))$, where $f : [0,1 / 2] \to \Pi$, $g : [0,1] \to \mathrm{M}$, and $h : [0,1] \to \mathcal{C}$ map $\hat{\pi}_t^j$, $\hat{\mu}_t^j$, and c^j to the nearest values in Π, M, and $\mathcal{C}$, respectively.

IV. Simulation Results

In this section, we present our simulation results. In Section IV.A, we describe the simulation setup. In Section IV.B, we investigate how, within the token system, users' mobility and battery states impact their own performance, the performance of other users, and the overall network performance.

A. Cellular Network Simulation Setup

We assume that $N = 1500$ mobile transceivers are uniformly and randomly distributed in a 10 km x 10 km square area consisting of 100 cells with size 1 km x 1 km. There is one base station at the center of each cell. In each time slot, each transceiver moves to a nearby location according to a random waypoint mobility model and needs to receive data from the base station in the corresponding cell with probability 0.3. We consider path loss and shadow fading for the channel model such that [9],

$$P_{\mathrm{rx}} = P_{\mathrm{tx}} - PL(d_0) - 10\alpha \log(d \,/\, d_0) - \chi, \qquad (11)$$

where P_{rx} and P_{tx} are the receive and transmit powers (in dB), respectively, $PL(d_0)$ is the path loss of the reference distance d_0, d is the distance between the source and destination, α is the path loss factor, and χ is a Gaussian distributed random variable representing the effect of shadow fading. We assume that the maximum transmission power of a transceiver is 15 dBm, the channel bandwidth is 10 MHz, and the target data rate is 10 Mbps for all devices. If the target data rate cannot be achieved with the maximum power, then the device requests a relay transmission. Using the above parameter values, the average ORDR throughout the network is approximately $\lambda = 0.1$. The required relay transmission power is calculated as the minimum power in the optimization defined in (3). The (normalized) cost c is the ratio of the relay transmission power (in mW) to the expected benefit $\mathbf{E}b(r)$ of achieving rate r. All devices adapt their cooperation strategy using the hybrid offline/online learning algorithm. Finally, we assume that there are a total of $T = 9000$ tokens in the network that are uniformly and randomly distributed among the users at the start of the simulation.

B. Impact of Mobility

In this section, we study how each user's mobility impacts its own performance, the performance of other users, and the overall network performance. We use the simulation test-bed described in Section IV.A and assume that all users deploy the hybrid offline/online learning algorithm with $\pi^x \in \{0.05, 0.1, \ldots, 0.45\} = \Pi$, $\mu^y \in \{0.05, 0.1, \ldots, 0.45\} = \mathrm{M}$, and $c \in \{0.05, 0.10, \ldots, 0.95\} = \mathcal{C}$ to compute the collection of policies offline.. We consider two classes of users:

1. **High mobility users** move at speeds between 50 and 120 km/hour. For instance, users in motor vehicles on roads or highways are considered high mobility users. These users play different roles in the network over time because sometimes they will be far from the base station and have a high ORDR, and sometimes they will be closer to the base station where they will have a high IRDR.
2. **Low mobility users** move at speeds between 0 and 8 km per hour. For example, users in offices, restaurants, or on foot are considered to be low mobility users. Due to their limited mobility, these users typically do not switch roles over time and their ORDR and IRDR are relatively static.

Since cellular networks usually contain both high mobility and low mobility users, we now investigate how different mobility mixtures impact the network performance. We consider five mixtures in which the network comprises 10%, 30%, 50%, 70%, and 90% high mobility users, and the remaining users have low mobility. In total, there are $N = 1500$ users. We assume that all users start in the maximum battery state, which is defined such that each user can ACK an average of 1000 relay requests before entering the dead state. In Fig. 3, we plot the average ORDR λ, average IRDR μ, average *modified relay recruitment efficiency* $e\Pr(k > 0, p > 0)$ [5], and average *throughput gain*[6] over 3000 time slots for users in each mobility class and for all users combined. The lower and upper error bars in Fig. 3 indicate the 25th and 75th percentiles, respectively, over the class of users.

ORDR and IRDR: Fig. 3(a) and Fig. 3(b) illustrate the average ORDR and average IRDR, respectively, under the various user mobility mixtures. Recall that, at the beginning of the simulation, users are uniformly distributed throughout the network. Since the low mobility users do not significantly deviate from their starting positions, and the high mobility users move many places throughout the network, the average ORDR and IRDR for users in each class is approximately the same; however, the variation of ORDRs and IRDRs across users within each class varies significantly. In particular, the low mobility users have a much larger range of ORDRs and IRDRs because if they are at the periphery of a cell, then they will have large ORDRs and small IRDRs, and if they are in the core of a cell, then they will have small ORDRs and large IRDRs. In contrast, the high mobility users move between the peripheries and cores of various cells over time, and therefore experience less deviation in these parameters across the population.

Modified relay recruitment efficiency and ACK rate: Fig. 3(c) illustrates the average modified relay recruitment efficiency under the various user mobility mixtures. The device's modified relay recruitment efficiency

[5] The device's modified relay recruitment efficiency is the probability that its outbound requests are ACK'd given that a relay is required [see (4)].

[6] The *throughput gain* is the ratio of the actual throughput to the direct transmission throughput. Since the actual throughput is always greater than or equal to the direct throughput, the minimum throughput gain is 1.

$e\Pr(k>0, p>0)$ is the conditional probability that its outbound requests are ACK'd given that a relay is required [see (4)]. Importantly, this parameter is intimately tied to each user's ORDR and IRDR. On average, low mobility users have lower modified relay recruitment efficiencies and more variation of these parameters across the population. These effects emerge because low mobility users have strongly imbalanced ORDRs and IRDRs unless they are lucky enough to be situated between the core and periphery of a cell. If a device's ORDR is larger than its IRDR, i.e., $\lambda^j > \mu^j$, then it tends to run out of tokens (i.e., $\Pr(k^j > 0)$ in (9) will be small). Without tokens, it cannot recruit relays, which reduces its modified relay recruitment efficiency. Note that this also hurts other users because it reduces opportunities to earn tokens. On the other hand, if a device's ORDR is smaller than its IRDR, i.e., $\lambda^j < \mu^j$, then it will tend to collect a surplus of tokens and not have any incentive to ACK incoming relay requests, which in turn reduces the relay recruitment efficiencies of the other devices in the network. Note that the ACK rate, i.e., the probability that a device ACKs an inbound request [the expectation $\mathbf{E}[\sigma^j(k^j, p^j)]$ in (9)], is directly proportional to the modified relay recruitment efficiency as shown in (9); consequently, we omit it from Fig. 3.

Throughput gain relative to direct transmission: Fig. 3(d) illustrates the average throughput gain under the various user mobility mixtures. Clearly, having a higher fraction of high mobility users in the network improves the average network throughput (relative to direct transmission only) because these users play different roles in the network over time, which enables a continuous exchange of tokens between devices in the periphery and core of each cell.

V. Conclusion

Cooperative communication and D2D technologies are central to the LTE-Advanced standard. However, these technologies assume that users are willing to share their limited battery energy to relay data for other users, even when doing so may reduce their utility. In this paper, we use token exchanges to provide self-interested users with incentives to relay data for other users. We formulate each device's decision problem, i.e., whether or not to relay, as a Markov decision process. We propose a simple and effective learning algorithm that a device can deploy to learn its optimal cooperation strategy online based on its experience. We evaluate the network performance in various deployment scenarios involving both high mobility and low mobility users. Our simulation results indicate that individual devices have the greatest incentive to cooperate when the network contains many highly mobile users (e.g., users in motor vehicles).

References

[1] T. C.-Y Ng and W. Yu, "Joint optimization of relay strategies and resource allocations in cooperative cellular networks," *IEEE Trans. on Select. Areas in Comm.*, vol. 25, no. 2, Feb. 2007.

[2] O. Alay, P. Liu, Y. Wang, E. Erkip, and S. S. Panwar, "Cooperative layered video multicast using randomized distributed space time codes," *IEEE Trans. on Multimedia*, vol. 13, no. 5, pp. 1127-1140, Oct. 2011.

[3] N. Mastronarde, F. Verde, D. Darsena, A. Scaglione, and M. van der Schaar, "Transmitting important bits and sailing high radio waves: a decentralized cross-layer approach to cooperative video transmission," *IEEE J. on Select. Areas in Comm. Cooperative Networking – Challenges and Applications*, vol. 30, no. 9, pp. 1597-1604, Oct. 2012.

[4] L. Buttyan and J.-P. Hubaux, "Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks,' EPFL techinical report, Jan. 2001.

[5] J. Xu and M. van der Schaar, "Token system design for autonomic wireless relay networks," *IEEE Trans on Communications*, to appear, 2013.

[6] A. S. Ibrahim, A. K. Sadek, W. Su, and K. J. R. Liu, "Cooperative communications with relay-selection: when to cooperate and whom to cooperate with?," *IEEE Trans. on Wireless Communications*, vol. 7, no. 7, pp. 2814-2827, July 2008.

[7] R. Madan, N. Mehta, A. Molisch, and J. Zhang, "Energy-efficient cooperative relaying over fading channels with simple relay selection," *IEEE Trans. on Wireless Communications*, vol. 7, no. 8, pp. 3013-3025, Aug. 2008.

[8] M. L. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming. New York: John Wiley & Sons, 1994.

[9] R. Jain, "Channel models: a tutorial," 2007. Available online: http://www.cse.wustl.edu/~jain/cse574-08/ftp/channel_model_tutorial.pdf

[10] B. Sirkeci-Mergen and A. Scaglione, "Randomized space-time coding for distributed cooperative communication," *IEEE Trans. on Signal Processing*, vol. 55, pp. 5003-5017, Oct. 2007.

[11] M. Franklin and M. Reiter, "Fair exchange with a semi-trusted third party," *ACM conference on Computer and Communication Security*, 1997.

[12] Y. Zhao, R. Adve, T. J. Lim, "Improving amplify-and-forward relay networks: optimal power allocation versus selection," *IEEE Trans. Wireless Commun*., vol. 6, no. 8, 2007.

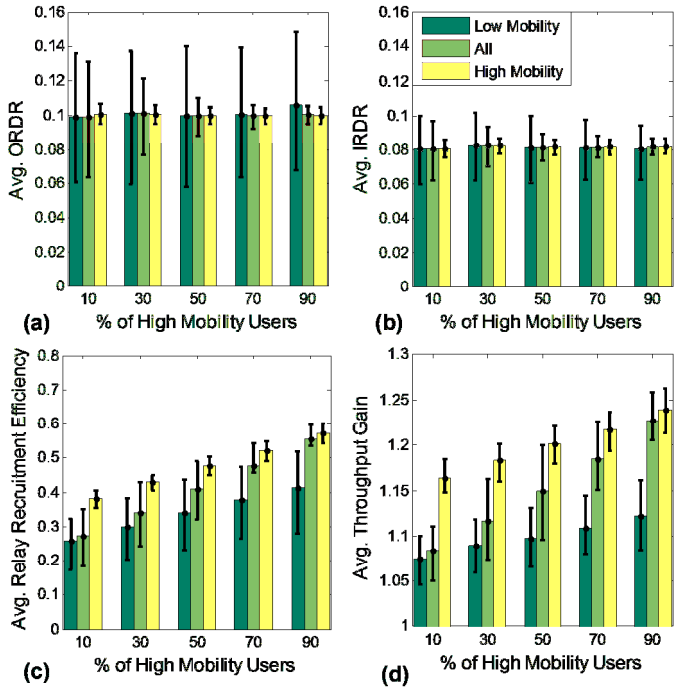

Fig. 3. Impact of different mobility mixtures on different user classes. (a) Average ORDR λ. (b) Average IRDR μ. (c) Average modified relay recruitment efficiency $e\Pr(k>0, p>0)$. (d) Average throughput gain.