Cooperative Mobile Internet Access with Opportunistic Scheduling

Pouya Ostovari*, Jie Wu*, and Abdallah Khreishah†
*Department of Computer & Information Sciences, Temple University, Philadelphia, PA 19122
†Department of Electrical & Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102

***Abstract*—Ubiquitous and resilient Internet connection access is becoming a necessity of life. Moreover, the amount of data requested by mobile users is increasing rapidly. Cooperative mobile Internet access is a promising approach to addressing these demands, by giving the mobile devices the opportunity to use the help of other devices to access the Internet. The helpers can download the data requested by the other users, referred to as clients, through their cellular connections, and can transmit the downloaded data to the clients using WiFi. In this paper, we consider the problem of sharing the resources of helpers among a set of clients that request the assistance of the helpers. Opportunistic scheduling is an effective method that uses the dynamic channel conditions to elevate the systems' overall utilities. We propose an opportunistic scheduling algorithm to use the helpers efficiently and share them among the clients in a fair way. Through simulation results, we show the effectiveness of our cooperative downloading methods.**

Keywords*—Cooperative download, resilient communication, opportunistic scheduling, device-to-device communication, cellular network, network coding.

I. Introduction

With the rapid development of mobile device technology, such as smartphones and tablets, these devices can provide their users with a convenient way to access the Internet. Smartphone and tablet use is increasing rapidly, and resilient ubiquitous Internet access is becoming a necessity in people's lives. The users can browse the Internet, download data, or stream videos from anywhere through cellular connections. On the other hand, the cellular data traffic is growing rapidly, and download rates offered by cellular networks might not be sufficient for users. Moreover, the user's cellular channel quality might not be sufficient to meet the user's demand, and the data rate of the cellular network can dramatically change over time. Consequently, the users might not get the quality of service that they expect [1].

Cooperative downloading is an effective approach for addressing the increasing traffic demand of the mobile devices, and can provide resilient and ubiquitous Internet access [2]. Using cooperation among the users, we can use the idle resources of the users to provide Internet access to other users or improve their data rate. As shown in Fig. 1(a), the helpers use their cellular connections, e.g. 4G/LTE, to download the request of the clients, and transmit the downloaded data to the clients using WiFi connections.

Consider the example in Fig. 1(b). The AP, helpers, and clients are shown as B, H, and C, respectively. The delivery rate of the links are shown beside the links, and the other links are reliable. The bandwidth of each link is 10Mb/s, and utility

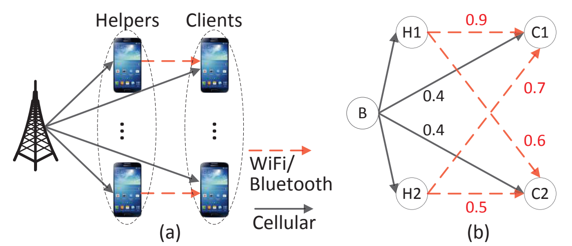

Fig. 1. (a) The system architecture. (b) Motivation example.

is defined as the total download rate of the clients. In the case of not using the helpers, the download rate of the clients and the total utility will be $10 \times 0.4 = 4$ and 8, respectively. Now consider the case of using the helpers. It is easy to check that assigning helpers 1 and 2 to clients 1 and 2 maximizes the total utility. In this case, the download rate of clients 1 and 2 will be 13 and 9, respectively. Not only does cooperation increase the download rate of the clients, but also provides resilient communications. The reason is that in the case of a broken link, the other links can replace it. However, is it clear that this is not a fair resource sharing, since client 2 receives less data than client 2. In a fair solution, the helper that serve the clients should be changed depending on the cumulative downloading rate of the clients in the previous time slots.

In order to have a fair resource sharing, we should use a fair matching algorithm to assign the helpers to the clients at each time slot. Moreover, it is typical to have a utility function that is not a linear function of the receiving data rate. The reason is that, as the receiving data rate of a client increases, the increasing rate of its satisfaction decreases. As a result, we need an optimization algorithm to find the optimal transmission rate of the helpers. Another challenge is the time-varying channel conditions due to fading, shadowing, and interference among the links. Opportunistic scheduling [3] is an effective approach used to deal with the dynamics of channel conditions and achieve higher network performance. The idea behind opportunistic scheduling is to consider the current channel condition of the users, and at each time slot, transmit to the user that maximizes the system performance and does not violate the fairness constraints.

In this work, we study the problem of providing cooperative Internet access to a set of clients through a group of helpers. We consider the unreliability of the links and time-varying channel condition of the links. In order to motivate the helpers to assist the clients, we use a credit-based incentive method. The credits that the clients need to pay the helpers depend on the effort of the helpers. In order to use the wireless resources more efficiently, we extend opportunistic scheduling to the case of multiple helpers, and use it in our cooperative

Internet access. We formulae the problem as an optimization problem, which can be solved in a distributed way.

The remainder of the paper is organized as follows. Background on network coding and related work is presented in Section II. Section III introduces the setting and problem statement. In Section IV, we propose our cooperative Internet access method. We present our simulation results in Section V. Section VI concludes the paper.

II. Related Work and Background

The authors in [4] introduce a system for sharing the on-demand and live video streaming among a set of users in the same vicinity. It is assumed that the users are friends and they agree to share their content with each other, which might not be practical in the cases that the users are not friends. There are many previous works on cooperative downloading that are proposed to share the same content to the users [5]. However, in our work the data requested by the users can be different, and there is no restriction on the type of applications.

The authors in [1] propose a system for cooperative downloading. In their system, a client sends requests to its neighbors to form a network. The users that agree to help the client send a confirmation message to the client. After forming the network, the download process is started and each helper receives a payment depending on the amount of help provided to the client. The authors propose an energy efficient method for scanning the neighbors. In contrast with [1], we consider several clients that are interested in using the assistance of a set of helpers, and focus on optimizing the total utility of the clients subject to fair resource sharing.

Network coding (NC) is introduced in [6] to solve the bottleneck problem in wired networks. *Random linear network coding* (RLNC) is introduced in [7], and the authors show that we can achieve the capacity asymptotically with respect to the finite field size in the case that the relays select the coefficients of the coded packets randomly. In RLNC, a coded packet is a linear combination of the packets. In order to code the packets, the coefficients of the linear combinations are selected randomly, and the coded packets have a form of $\sum_{i=1}^{k} a_i \times P_i$. Here, the symbols a and P are the random coefficients and the packets. If we use RLNC to code k native packets together, with a large probability any k coded packets are sufficient to decode the coded packets and retrive the original packets. In other words, with a probability close to one, any k random linear coded packets are linearly independent. In order to decode the coded packets, Gaussian elimination can be used to solve a system of linear equations.

III. System Model and Problem Statement

A. System Model

We consider a set of mobile helpers $H = \{1, .., m\}$, e.g. smartphones and tablets, that are willing to cooperate in providing internet access to a set of mobile clients $I = \{1, .., n\}$. The helpers and clients can communicate with each other by forming a mesh network $G = (H \cup I, E)$, where E represents the set of links connecting them through WiFi. The helpers get access to the Internet through a base station, e.g 4G/LTE connection and provide the clients with the Internet using WiFi. We consider a time-slotted system, and during each time slot the channel conditions are fixed. We assume that the total number of WiFi channels are equal to m, i.e. each helper works on a different channel. Thus, each helper can serve only one client at a time. Moreover, each client cannot receive help from more than one helper at each time slot. The helpers and clients can use their cellular and WiFi connections simultaneously.

The clients and helpers consume energy while downloading or relaying data. The energy consumption of client i to receive one byte through cellular and WiFi connections are represented as e_i^c and e_i^w. Typically, the energy consumption of cellular connections is more than that of the WiFi connections. We represent the bandwidth of the link between helper j and client i at time t as b_{ji}^t. Also, b_i^t and b_j^t denote the bandwidth of the link from the base station to client i and helper j, respectively. The delivery rate of the links change over time and we denote the delivery rate of the link ϵ_{ji} at time t as p_{ji}^t. We represent the transmission rate from the base station to the client i at time t as x_i^t. Moreover, x_{ji}^t is defined as the transmission rate of helper j to client i at time slot t.

In our model, the links are not reliable. Therefore, some of the packets might be lost and need to be retransmitted. In order to eliminate the need for feedback messages, we use RLNC. The packets that should be transmitted to each client are partitioned into segments of equal sizes, and RLNC is performed among the packets of the same segment. The coding is done over a finite field, and a coded packet has a form of $\sum_{i=1}^{k} a_i \times P_i$. The symbols a and P are the random coefficients and the packets, respectively. The helpers receive coded packets from the base station, and recode the packets before transmitting them. The clients are able to decode the coded packets once they receive a sufficient number of coded packets, i.e. k linearly independent coded packets.

The use of RLNC enables a flow-based model of the content, which simplifies our proposed schemes. Without NC, we need to decide which packets should be transmitted by the base station or the helpers. However, with NC, we just need to find the rates at which the coded packets should be transmitted. It should be noted that other types of coding schemes, such as fountain codes [8], can also be applied on top of our solutions. However, for ease of description, we use RLNC in our proposed methods. An overview on linear NC is provided in Section II.

B. Problem Statement

In order to motivate the helpers to participate in the cooperation, we need to have an incentive mechanism. For this purpose, the clients pay the helpers based on the data rate that the helpers transfer to the clients. It should be noted that the rates at which the clients receive data might be less than the transfer rates of the helpers, which is due to the unreliable links. On the other hand, the clients consume energy to receive data from the base station and the helpers. Therefore, we define the utility of each client as a function of the data rate that it receives minus its energy consumption and the credits that the client needs to pay the helpers.

The utility of client i at time t is defined as $U(i,t) = f(x_{ji}^t p_{ji}^t + x_i^t p_i^t) - e(i,t) - x_{ji}^t z$. Here, z represents the credits that client i needs to pay a helper in order to transmit

TABLE I. THE SET OF SYMBOLS USED IN THIS PAPER.

Notation	Definition
x_{ji}^t	Transmission rate from helper j to user i at time t
x_i^t	Transmission rate from base station to user i at time t
y_i^t	Total download rate of user i at time t
p_{ji}^t	Delivery rate of the link from helper j to client i at time t
p_i^t/p_j^t	Delivery rate from base station to client i/helper j at time t
b_{ji}^t	Bandwidth of link between helper j and client i at time t
b_i^t	Bandwidth of the link from base station to client i at time t
b_j^t	Bandwidth of the link from base station to helper j at time t
e_i^w/e_i^c	Energy consumption of user i for receiving one byte from the helpers/base station

one byte. The transmission rate of helper j to client i is equal to x_{ji}^t, and the data rate that is delivered to client i is equal to $x_{ji}^t p_{ji}^t$. Moreover, the rate of the data that is delivered to client i from the cellular network equals $x_i^t p_i^t$. We represent the energy consumption of i as $e(i,t)$, which equals $x_{ji}^t p_{ji}^t e_i^w + x_i^t p_i^t e_i^c$. Function $f(.)$ is a strict concave, non-decreasing, and continuously differentiable function of the receiving data rate. The reason for the concavity assumption relies on the fact that as the receiving data rate of a client increases, the increasing rate of its satisfaction decreases. In economics, this fact is known as the "law of diminishing returns" [9]. Since the energy consumption and the credit that needs to be paid to the helpers increases as the receiving data rate increases, $U(i,t)$ might reduce. Assuming that the energy consumption and credit payments are linear functions of the transition rate, $U(i,t)$ becomes a strictly concave function. Therefore, it has a unique maximum.

Our objective is to maximize the aggregated utility of the clients. However, this maximization problem might result in an unfair resource sharing among the users. The clients with bad channel conditions might not be able to use the help of the helpers in the case that the number of helpers is less than that of the clients. As a result, in our optimization problem, we try to provide fairness, which is discussed in the next section.

IV. Cooperation Schemes

We first solve the optimization problem without considering the energy consumptions and the credits that need to be paid by the clients. We then extend our solution when considering the energy consumptions and the credit payments.

A. Transmission with WiFi

Fairness is a main part of scheduling problems in wireless networks. If we do not consider fairness, we can trivially optimize the system performance by finding a bipartite matching of the helpers to the clients such that the total utilities are maximized. However, this scheduling might not be fair as the clients with good channel conditions might keep the channels forever. Thus, the clients with poor channel conditions might not be able to use the assistance offered by the helpers.

To give an idea of opportunistic scheduling consider the example in Fig. 2. The channel condition of users B and C are shown in Fig. 2(b). The channel condition of node B is always better than that of the node C. As a result, if we want to maximize the data rate that the nodes receive, node A should always transfer data only to node B. Clearly, this is not a fair scheduling. Now assume that we assign half of each time slot to user B. Node B experiences 3 slots with bandwidth 8,

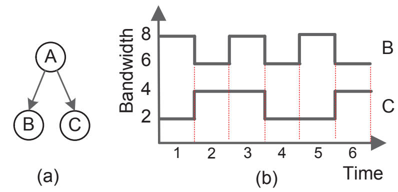

Fig. 2. Opportunistic scheduling; (a) Topology; (b) Time varying channel conditions.

and 3 slots with bandwidth 6; thus, its receiving rate will be $(8/2+6/2)\cdot 3 = 21$. Also, the receiving rate of node C will be $(4/2+2/2)\cdot 3 = 9$. However, if A assigns time slots 1, 3, and 5 to node B, and slots 2, 4, and 6 to node C, the receiving rates of B and C will be 24 and 10, respectively. In this case, not only does the total utility of the system increase, but the utility of each user is also enhanced.

In order to provide fairness, we can extend the idea in [3] to the case of multiple helpers. The idea is that instead of maximizing the total utility, we maximize $\sum_{i=1}^{n} \alpha_i^t U(i,t)$. Here, $U(i,t)$ is the utility of client i at time t, and α_i^t are real parameters that control the fairness. Similar to the work in [3], the idea behind parameter α is to give a chance to the client whose utility received in the previous time slots is low to use the assistance of the helpers. The α variable for the clients with large received utilities is lower than the other nodes. Thus, for each time slot, the relatively best clients are selected and assigned to the helpers. The clients with large α are the unfortunate nodes in that their channel conditions are worse than the other clients. We will discuss the calculation of α later. The problem of maximizing the total system utility subject to fairness can be formulated as follows:

$$\max \sum_{i\in I}\sum_{t=1}^{T} \alpha_i^t U(i,t) \quad (1)$$

$$s.t.\ x_{ji}^t \le p_j^t b_j^t / p_{ji}^t, \quad \forall i,j : j\in H, i\in I \quad (2)$$

$$x_{ji}^t \le b_{ji}^t, \quad \forall i,j : j\in H, i\in I \quad (3)$$

$$x_i^t \le b_i^t, \quad \forall i\in I \quad (4)$$

$$y_i^t \le x_i^t p_i^t + \sum_{j\in H} x_{ji}^t p_{ji}^t, \forall i\in I \quad (5)$$

$$[x_{ji}^t] \in R_1 \quad (6)$$

where, $U(i,t)$ is the utility of client i at time slot t. Also, x_{ji}^t and p_{ji}^t are the transmission rates of helper j to client i and the delivery rate of the link ϵ_{ji} at time t, respectively. We represent the total download rate of client i at time t as y_i^t. The objective function (1) is to maximize the total utility of the clients. Here, we consider $U(i,t) = f(y_i^t)$, where $f(y_i^t)$ is a monotonic increasing concave function.

The receiving rate of helper j equals $p_j^t b_j^t$. Therefore, client i cannot receive data from helper j at a rate greater than $p_j^t b_j^t$. On the other hand, the receiving rate of client i equals to $x_{ji}^t p_{ji}^t$; thus, we have $x_{ji}^t p_{ji}^t \le p_j^t b_j^t$, which is stated as the set of Constraints (2). The set of Constraints (3) and (4) are bandwidth constraints. Constraint (5) calculates the total receiving rate of the clients. Constraint (6) implies that the transmission rate should be feasible. Here, R_1 is the set of possible scheduling. In our model, a helper cannot transmit to more than one client at the same time slot and a client cannot receive data from multiple helpers concurrently.

Algorithm 1 Scheduling Algorithm

At each time slot t perform the following steps:
for each client i and helper j **do**
    $G(i,t) = f(min(b^t_{ji}p^t_{ji}, b^t_j p^t_j) + b^t_i p^t_i) - f(b^t_i p^t_i)$
    $\alpha^t_i = 1/\sum_{t'=1}^{t-1} G(i,t), \forall i \in I, \quad \alpha^1_i = 1, \forall i \in I$
    Assign $\alpha^t_i G(i,t)$ to link ϵ_{ji}
Find the MWBM using Hungarian algorithm

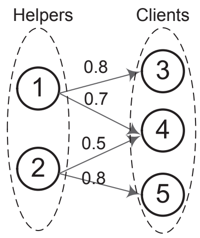

Fig. 3. Optimal scheduling.

Therefore, if for helper j and client i we have $x^t_{ji} > 0$, then $x^t_{j'i} = 0, \forall j' \neq j$ and $x^t_{ji'} = 0, \ \forall i' \neq i$.

Our scheduling algorithm for time slot t works as follows. First, for client i, we calculate its maximum utility when it uses only the cellular connection. We then calculate the increase in client i's maximum utility in the case of using helper j in addition to the cellular connection. Since we do not consider the energy consumption and the credit payments, the utility of each client is an increasing concave function. Therefore, the optimal rate assignment for client i is to use the full bandwidth of both of the cellular connection and helper j. The increase in the utility of client i in the case of using helper j is equal to $G(i,t) = f(min(b^t_{ji}p^t_{ji}, b^t_j p^t_j) + b^t_i p^t_i) - f(b^t_i p^t_i)$. The reason for taking the minimum of the two values is that the receiving rate of client i cannot exceed the bandwidth of the link ϵ_{ji} and the receiving rate of helper j. Then, we multiply each utility enhancement $G(i,t)$ by α^t_i, and assign the result to the link ϵ_{ji}. We run the Hungarian algorithm [10] to find the maximum weighted bipartite matching (MWBM) of the helpers to the clients. In order to provide fairness, in this paper we calculate α as $\alpha^t_i = 1/\sum_{t'=1}^{t-1} G(i,t)$, and we set $\alpha^1_i = 1, \forall i \in I$. The details are shown in Algorithm 1.

Consider the example in Fig. 3. For simplicity, we assume that the bandwidth of the links from the base station to the helpers are not bottleneck, i.e. their bandwidth and reliability are much higher than the links between the helpers and the clients. Also, we do not consider the direct links from the base station to the clients. The bandwidth of the links between the helpers and clients are equal to 5. The delivery rate of the links are shown in the figure. Assuming function $f(.) = \log(x+1)$, the utility of client 4 in the case of downloading from helper 1 and 2 becomes $\log(5 \times 0.7 + 1) = 1.5$ and $\log(5 \times 0.5 + 1) = 1.25$, respectively. Also, the utility of the clients 3 and 5 when they download through helpers 1 and 2 are equal to $\log(5 \times 0.8 + 1) = 1.61$. Thus, the bipartite matching that maximizes the total utility is assigning helper 1 to client 3, and helper 2 to client 5.

Assuming that the reliability and bandwidth of the links are fixed, in time slot 2 the utility of clients 3 and 5 are divided by the utility that they received at time slot 1. As a result, the weight of the links from helpers 1 and 2 to clients 3 and 5 becomes 1. The weight of the links from helpers 1 and 2 to client 4 are still 1.5 and 1.25, respectively. Therefore, the optimal weighted bipartite matching is to schedule clients 4 and 5 to receive from helpers 1 and 2, respectively.

Our proposed optimization can be implemented in a distributed way as follows. Calculating the utility of the client for each helper assignment can be performed in a distributed way by each client or helper. Also, the α variables are calculated by each client separately. The products of the $G(i,t)$ and α^t_i variables are used as the weight of the links ϵ_{ji}. Then, we can apply a distributed version of the MWBM algorithm, e.g. [11], to find the optimal scheduling. We refer to this method as the joint scheduling and rate control (JSRC) method.

B. Transmission with WiFi Considering the Costs

1) Formulation: In the previous subsection, we did not consider the energy consumption and the credit payments. Thus, the utility of the clients was an increasing function of their received data rate. In that case, the utility function of the clients is an increasing function of the received data rate. However, when we take the energy consumption and the credit payments into account, a client might prefer not to use the full capacity of the link that is assigned to it. The reason is that, as the receiving rate of a client increases, its energy consumption and the credits that need to be paid to the helper increase as well. These increase in the costs are linear to the download rate. However, the function $f(.)$ is not linear ($f(.)$ is strictly concave). Consequently, the whole utility might decrease depending on the receiving rate. In this case, the utility function becomes $U(i,t) = f(y^t_i) - x^t_i p^t_i e^c_i - \sum_{j \in H}[x^t_{ji}p^t_{ji}e^w_i + zx^t_{ji}]$. The objective function and the other constraints are the same as those in the previous subsection. This optimization contains two sub-problems: (1) scheduling the links, (2) finding the optimal data rates. In order to find the optimal solution, we decompose the optimization into scheduling and rate optimization. Assuming n helpers and n clients, there are $n!$ possible matchings; thus, the time complexity of checking the total utility of all of the possible matchings is exponential.

In this problem, there is no need for checking all of the possible matchings. In our model, each client can download from one of the m helpers, and each helper cannot serve more than one client. Therefore, the utility of each client only depends on the matching, and is independent of the download rate and the utility of the other clients. Thus, in our polynomial time algorithm, we calculate the difference in the optimal utilities of client i in the cases of using the assistance of helper j and not using any helpers, denoted as $G(i,t)$. We then assign $\alpha^t_i G(i,t)$ as the weight of link ϵ_{ji}, and run the Hungarian algorithm [10] to find the maximum weighted matching of the helpers to the clients. The Hungarian algorithm can find the maximum weighted matching of a bipartite graph in a polynomial time. Other MWBM algorithms can be used instead of the Hungarian algorithm.

2) Optimization: In order to find the optimal rate allocations for each helper assignment, we need to perform an optimization algorithm. Consider client i, which is scheduled to receive from helper j at the current time slot t. The optimal transmission rates from the base station and helper j to client

Algorithm 2 Calculation of x_{ji} and x_i (for client node i)

$\gamma_i = \lambda_4^i p_{ji} - \lambda_1^j - \lambda_2^{ji} - z - p_{ji} e_i^w$
if $\gamma_j^i > 0$ **then** $x_{ji} = min(b_{ji}, b_j p_j / p_{ji})$ **else** $x_{ji} = 0$
if $\lambda_4^i p_i - \lambda_3^i - p_i e_i^c > 0$ **then** $x_i = b_i$ **else** $x_i = 0$

i can be found by solving the following convex optimization:

$$\max U(i,t) = f(y_i^t) - x_i^t p_i^t e_i^c - x_{ji}^t p_{ji}^t e_i^w - z x_{ji}^t$$
$$s.t\ x_{ji}^t \leq p_j^t b_j^t / p_{ji}^t;\ x_{ji}^t \leq b_{ji}^t;\ x_i^t \leq b_i^t;\ y_i^t \leq x_i^t p_i^t + x_{ji}^t p_{ji}^t$$

We can find the optimal rate allocation by solving the Lagrangian dual of the problem using the gradient method. In this way, we gradually update the transmission rates, based on the Lagrange variables. Since the Slater condition holds in this problem (see reference [12]), there is no duality gap between the primal and the dual problems. Let λ_1^j, λ_2^{ji}, λ_3^i, and λ_4^i be the Lagrange variables for the constraints. For simplicity, we do not show the t superscripts. The Lagrange function becomes:

$$\begin{aligned} L(x_i, x_{ji}, y_i, \vec{\lambda}) &= f(y_i) - x_i p_i e_i^c - x_{ji} p_{ji} e_i^w - z x_{ji} \\ &- \lambda_1^j (x_{ji} - p_j b_j / p_{ji}) - \lambda_2^{ji} (x_{ji} - b_{ji}) \\ &- \lambda_3^i (x_i - b_i) - \lambda_4^i \big[y_i - x_i p_i - x_{ji} p_{ji} \big] \end{aligned}$$

By rearranging the terms and removing the constants, we have:

$$L(x_i, x_{ji}, y_i, \vec{\lambda}) = f(y_i) - \lambda_4^i y_i \tag{7}$$
$$+ x_{ji} (\lambda_4^i p_{ji} - \lambda_1^j - \lambda_2^{ji} - z - p_{ji} e_i^w) \tag{8}$$
$$+ x_i (\lambda_4^i p_i - \lambda_3^i - p_i e_i^c) \tag{9}$$

The objective function of the dual problem is $D(\vec{\lambda}) = \max_{x_i, x_{ji}, y_i} L(x_i, x_{ji}, y_i, \vec{\lambda})$. The dual problem is $\min_{\lambda} D(\vec{\lambda})$. We can solve the dual optimization problem using the gradient method. The updates of the Lagrange variables are as follows:

$$\lambda_1^j(\tau+1) = \big[\lambda_1^j(\tau) + \beta(x_{ji}(\tau) - p_j b_j / p_{ji})\big]^+$$
$$\lambda_2^{ji}(\tau+1) = \big[\lambda_2^{ji}(\tau) + \beta(x_{ji}(\tau) - b_{ji})\big]^+$$
$$\lambda_3^i(\tau+1) = \big[\lambda_3^i(\tau) + \beta(x_i(\tau) - b_i)\big]^+$$
$$\lambda_4^i(\tau+1) = \big[\lambda_4^i(\tau) + \beta\big(y_i(\tau) - x_i(\tau) p_i - x_{ji}(\tau) p_{ji}\big)\big]^+$$

The projection on $[0, +\infty)$ is represented as $[.]^+$. Also, β is the step size. In order to find the y_i that maximizes (7), we set the first derivative of (7) with respect to y_i equal zero. If we consider $f(y_i) = \log(y_i + 1)$, the optimal y_i becomes $y_i = 1/\lambda_4^i - 1$. For iteration τ, if y_i becomes infinity, we set y_i to $p_i b_i + p_{ji} b_{ji}$. Equations (8) and (9) are linear functions of x_{ji} and x_i. Thus, in order to maximize them, when the multipliers of x_{ji} and x_i are greater than zero, their value should be set to the maximum possible value, which depends on the bandwidths. On the other hand, in the case of negative multipliers, we should set x_{ji} and x_i to zero. Algorithm 2 illustrates the computation of x_{ji} and x_i. Here, $\gamma_i = \lambda_4^i p_{ji} - \lambda_1^j - \lambda_2^{ji} - z - p_{ji} e_i^w$ is the multiplier of x_{ji} in Equation (8). After finishing the iterations, the final values of x_{ji} and x_i are set to the average calculated values of the iterations.

The optimal rate allocation in the case of not using the assist of helpers can be found by setting the first derivative of $f(x_i) - x_i p_i e_i^c$ respect to x_i equal to zero. Then, $G(i,t)$

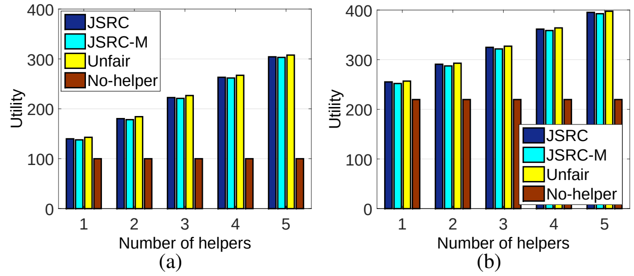

Fig. 4. Total utility of the clients, delivery rates $\in [0.5, 1]$, $n = 10$, $T = 50$, $b_j^t \in [2, 4]$, $b_{ji}^t \in [1, 2]$. (a): $b_i^t \in [0.2, 0.4]$; (b): $b_i^t \in [0.5, 1]$.

is equal to the difference of the utilities in the case of using helper j and not using any helper. We assign $\alpha_i G(i,t)$ to link ϵ_{ji}, and select the MWBM using the Hungarian algorithm, and pick its respective optimal rates.

The optimization in the case of considering the download costs can be implemented in a distributed way. First, each client uses the proposed gradient approach to find the optimal rate when helper j is assigned to it, calculates $G(i,t)$, and multiplies $G(i,t)$ by α_i. Then, the result is assigned as the weight of link ϵ_{ji}. Finally, the distributed version of the MWBM algorithm, e.g. [11], is run to find the scheduling that maximizes the total utility. We refer to this method as the joint scheduling and rate control with payments (JSRCP).

V. Simulations

A. Simulation Setting

We develop a simulator in the MATLAB environment, and compare each proposed method with the optimal solutions in the case of not considering the fairness, refereed to as the Unfair method. In order to remove the fairness constraint from the optimization constraints, we set $\alpha_i^t = 1\ \forall i \in I$. As a result, at each iteration, the matching that maximizes the total utility of the clients is selected as the optimal solution. We also modify our JSRC and JSRCP methods by setting α_i^t to the amount of increment in the utility of the client i in the previous time slots that is due to using the helpers. We refer to these methods as JSRC-M and JSRCP-M, respectively. In our simulations, we consider $f(y_i) = \log(y_i + 1)$. The reason to add one to y_i is to set the utility function $U(i,t)$ equal to zero in the case that the download rate of user i from the base station and the helpers is equal to zero. We run each simulation 200 times and report the average results.

B. Simulation Results

1) WiFi without Credit Payments: In Fig. 4, we evaluate the utility. Fig. 4(a) shows that the total utility without using the helpers is less than that of the other methods. Also, the Unfair method has the highest utility compared to the other methods. However, the utility of the JSRC method is about only 5% less than that of the Unfair method. The figure illustrates that the utility of the JSRC-M method is about 6% lower than that of the JSRC approach. As we increase the number of helpers, more resources are provided, which increases the utility of the JSRC, JSRC-M, and the Unfair methods. In Fig. 4(b), we increase the bandwidth of the links between the base station and the clients to the range of $[0.5, 1]$ and repeat the pervious experiment. The total utility of all of the methods increase when the clients can download more data directly from the

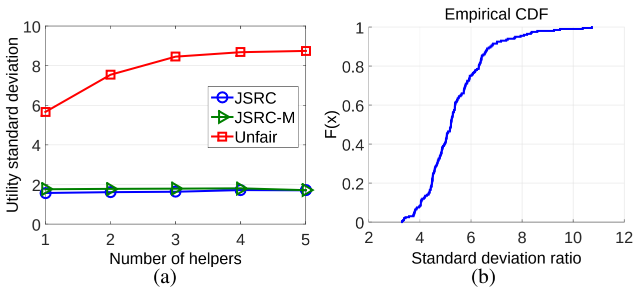

Fig. 5. Total utility of the clients, delivery rates $\in [0.5, 1]$, $n = 10$, $T = 50$, $b^t_j \in [2, 4]$, $b^t_{ji} \in [1, 2]$. (a): $b^t_i \in [0.2, 0.4]$; (b): $b^t_i \in [0.5, 1]$.

base station. In Fig. 4(b), the difference between the utility of the No-helper method and the other methods is less than those in Fig. 4(a). When the cellular connection of the clients has sufficient bandwidth, there is not need for the helpers.

The previous experiments show that there is no huge difference between the utility of the Unfair, JSRC, and JSRC-M methods. These results illustrate the effectiveness of our opportunistic scheduling in using the resources. In order to check if our opportunistic scheduling mechanisms can provide fairness, we measure the standard deviations of the total utility that the different clients receive. Fig. 5 depicts the utility standard deviation of the scheduling methods. The figure shows that the standard deviation of the JSRC and JSRC-M methods are almost the same. However, the standard deviation in the case of using the Unfair method is between 3 to 4.5 times that of the JSRC and JSRC-M methods. The utility standard deviation of the Unfair method increases as we increase the number of helpers. The reason is that, more helpers will increase the chance that some of the clients are connected to the helpers through channels with good conditions. Therefore, these clients receive much more utility than the other clients.

Fig. 5(b) shows the empirical CDF of the utility standard deviation in the case of using the JSRC method to that of the Unfair method. For each run, we divide the utility standard deviation of the Unfair method by that of the JSRC method, and plot the empirical CDF. The figure shows that in 50% of the cases, the utility standard deviation of the Unfair method is up to 5 times that of the JSRC method. Moreover, in 20% of the cases, the utility standard deviation of the Unfair method is more than 6 times that of the JSRC method.

2) WiFi with Credit Payments: We repeat our first experiment to evaluate the utility in the case of considering the costs. Fig. 6(a) depicts that the total utility without using the helpers is up to 60% less than the other methods. The Unfair method has the highest utility compared to the other methods. Also, the utility of the JSRCP method is about only 5% less than that of the Unfair method. The utility of the JSRCP-M method is about 2% lower than that of the JSRCP approach. In the JSRCP-M method, we use the utility enhancement due to the received help from the helpers to calculate the α_i variable. Clearly, as the number of helpers rises, more help is provided to the clients, and the utility of the methods increases.

We compare the standard deviation of the utility that the clients receive in Fig. 6(b). The standard deviation of the JSRCP and JSRCP-M methods are very close. However, the standard deviation of the JSRCP-M method is less than that of the JSRCP method, which means JSRCP-M is more fair than the JSRCP method. This is due to calculating α_i based on the

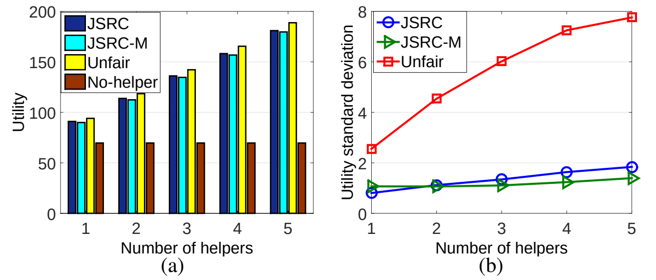

Fig. 6. Delivery rates $\in [0.5, 1]$, $n = 10$, $T = 50$, $b^t_j \in [2, 4]$, $b^t_i \in [0.2, 0.4]$, $b^t_{ji} \in [1, 2]$, $e^c_i = 0.3$, $e^w_i = 0.1$, $z = 0.1$. (a): Total utility of the clients; (b): Standard deviation of the clients' utilities.

utility enhancement that client i receives from the helpers. The standard deviation in the case of using the Unfair method is up to 4 times that of our proposed methods. From the results in Figs. 6(a) and (b), we can conclude that using the JSRCP and JSRCP-M methods we can provide fairness at a cost of about 2-5% reduction in the total performance.

VI. Conclusion

Ubiquitous Internet connection access is becoming a requirement of our lives. Also, the amount of data requested by the mobile users is increasing rapidly. An effective approach to address these two demands is to use cooperative mobile Internet access. In this work, we consider the problem of providing an Internet connection to a set of clients with the cooperation of a set of helpers. In order to increase the total utility of the clients, we use opportunistic scheduling to use the resources efficiently. The reported simulation results show the effectiveness of our proposed methods.

References

[1] 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, pp. 1545–1553.

[2] G. Iosifidis, L. Gao, J. Huang, and L. Tassiulas, "Enabling crowd-sourced mobile internet access," in *IEEE INFOCOM*, 2014, pp. 451–459.

[3] X. Liu, E. Chong, and N. Shroff, "A framework for opportunistic scheduling in wireless networks," *Computer Networks*, vol. 41, no. 4, pp. 451–474, 2003.

[4] L. Keller, A. Le, B. Cici, H. Seferoglu, C. Fragouli, and A. Markopoulou, "Microcast: cooperative video streaming on smartphones," in *ACM MobiSys*, 2012, pp. 57–70.

[5] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Communications Surveys and Tutorials*, vol. 99, no. PP, p. 1, 2013.

[6] R. Ahlswede, N. Cai, S. Li, and R. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[7] S. Li, R. Yeung, and N. Cai, "Linear network coding," *IEEE Transactions on Information Theory*, vol. 49, no. 2, pp. 371–381, 2003.

[8] M. Luby, "LT codes," in *The 43rd Annual IEEE Symposium on Foundations of Computer Science*, 2002, pp. 271–280.

[9] X. Lin and N. Shroff, "Joint rate control and scheduling in multihop wireless networks," in *IEEE CDC*, 2004, pp. 1484–1489.

[10] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[11] J. Schwartz, A. Steger, and A. Weißl, "Fast algorithms for weighted bipartite matching," in *Experimental and efficient algorithms*, 2005, pp. 476–487.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge Univ Press, 2004.