Maximum Utility-Aware Capacity Partitioning in Cooperative Computing

Nitin Singha, *Member, IEEE*, Sanket S. Kalamkar, and Yatindra Nath Singh, *Senior Member, IEEE*

***Abstract*—In many networks, a user has to allocate the link capacity between upload and download. When such networks are used for cooperative computing, the user needs to maintain the division of upload and download capacities at an optimal value to receive the maximum utility. To determine this optimal value, we model upload-download partitioning as a resource maximization game. We show that a Nash equilibrium (NE) obtained for this game is socially optimal. Thus this NE acts as an upper bound on capacity partitioning and serves as a benchmark to analyze the efficiency and performance of various capacity partitioning algorithms. Specifically, using this upper bound and simulations, we examine the performance of different partitioning algorithms while considering the dynamics of resource requests.**

***Index Terms*—Capacity partitioning, cooperative computing, distributed system, game theory, Nash equilibrium (NE)**

I. Introduction

Many distributed systems, such as file sharing, streaming, and distributed computing, use cooperative computing [1], [2], where users share their resources, *e.g.*, capacity, computation power, or data files. They gain benefits by downloading resources contributed by others. There is no centralized control. Users are expected to upload and download resources at their discretion. But a user might be reluctant to upload due to inherent cost attached to it. Many incentive mechanisms have been proposed to promote uploading, where users receive resources based on the amount they upload [1]–[3].

Nowadays, users can access a backbone network using wireless links. But they have access links with different capacities due to variations in local wireless environment, such as path loss and fading. It may be possible that the uplink and downlink traffic flows through a common access link before entering the backbone network. WiFi, LTE, and WiMAX (in time-division duplex (TDD) mode) are some examples of such scenarios [4]–[6]. For instance, in the TDD mode, time slots allocated to upload and download can be adjusted to achieve the required partitioning of the link capacity between upload and download [7], [8]. In frequency-division duplex, the bandwidth of the access link is split between upload and download to achieve the required capacity partitioning. A user can alter the division of the link capacity between upload and download such that their sum is equal to the given link capacity [4], [5]. Reluctant to upload, a user prefers to allocate zero capacity to the upload, but the incentive mechanism forces the user to maintain a certain level of upload. Hence, the user seeks to partition the link capacity at an optimal value. In particular, the user maintains a minimal level of upload to receive just enough resources to meet requirements and thus receive maximum resources (utility) from the network.

Many capacity partitioning algorithms exist to maximize the user's utility [4], [5]. Instances of these algorithms run at each user and dynamically adjust user's upload and download capacities to maximize the utility. But these algorithms do not explore any partitioning strategy where a user receives the maximum utility. A partitioning strategy implies how a user divides the link capacity between upload and download. The strategy maximizing the utility can be used to analyze the performance of various algorithms employed to partition the capacity, which is the focus of this paper.

Related work. Game theory is widely used in cooperative computing, but not in the wireless setup under consideration [3], [9], [10]. To maximize received resources, [3] discusses resource auctioning. The work in [9] proposes an optimal bidding during auctions to receive maximum resources. Such payment-based incentives are generally not used in cooperative computing, because they require dedicated systems for accounting and payments, thereby placing an extra overhead on the network. The requester's past cooperative behavior based on the amount uploaded is commonly used to promote uploading. If all users increase their upload, it results in overall increase in received resources across the network. The work in [10] presents a game-theoretic analysis on upload behavior to maximize received resources. But this study is restricted to Asymmetric digital subscriber line (ADSL) links and does not consider flexibility in the link capacity partitioning. In [6], the authors obtain a partitioning strategy for wireless links under the constraint that users carry out only one upload and download at a time. This constraint restricts the maximum download rate (resources received) achieved by any user to half of the network's slowest link capacity. The maximum resource received using this analysis is much smaller than what is received in practice. Overall, to the best of our knowledge, no analysis is available to estimate the maximum resource that can be received in a wireless network with access links.

This paper makes the following contributions.

Resource maximization game. We model interaction between users performing multiple uploads and downloads simultaneously as a resource maximization game. In this game, users strategically assign their access link capacities for upload to derive the maximum utility from the network.

Upper bound on resources received. We obtain an upper bound on resources received by a user, which is shown to be a Nash equilibrium (NE).

Capacity partitioning algorithms. Using the obtained upper bound, we evaluate the performance of various capacity partitioning algorithms employed by users to increase received resources. Closer the partitioning to the NE, the higher the resources received by that algorithm.

Nitin Singha is with the Department of Electronics and Communication Engineering, National Institute of Technology, Delhi, India. Sanket S. Kalamkar is with Qualcomm Inc., San Diego, CA, USA. Yatindra Nath Singh is with the Department of Electrical Engineering, Indian Institute of Technology Kanpur, Kanpur, India. E-mail: nitinsingha@gmail.com, sanket.kalamkar.work@gmail.com, ynsingh@iitk.ac.in.

Sanket S. Kalamkar was with INRIA and ENS, Paris, France when this work was done.

II. System Model

Network model. Consider a network consisting of users, access links, and a backbone network [5], [6]. Each user connects to the backbone network using a wireless access link. Let C_i denote the capacity of the wireless access link for user i. Here C_i can be characterized by the well-known Shannon's formula $C_i = B_i \log_2(1+\mathsf{SNR}_i)$, where B_i is bandwidth and SNR_i is the signal-to-noise ratio (SNR) for user i. Users have access links with different capacities owing to different SNR values. The variation in SNR arises due to different wireless conditions, such as path loss and fading, that users experience. The capacity C_i is divided between upload s_i and download d_i capacities. This division can be changed by the user under the constraint $C_i = s_i + d_i$.[1] For the aforementioned Shannon's formula, the capacity partitioning at a user can be achieved by appropriately allocating the bandwidth to upload and download based on respective SNRs.

Request profile. Users randomly request resources from other users in the network. Their instantaneous requests may follow any distribution, but in the long run, they request from all users with the same probability [4], [5].

Resource allocation and incentives. The upload capacity s_j of a serving user j (user receiving a resource request) acts as a resource for requesters. Requesters use this capacity s_j to download files from user j. User j may receive multiple requests. When resources demanded are less than or equal to the upload capacity of user j, all requesters are satisfied. Otherwise, the upload capacity s_j of user j is allocated among the requesters based on their incentive levels [10]. The incentive level of user i is defined as [4], [5]

$$I_i \triangleq \frac{\text{resources shared by user } i}{\text{resources demanded by user } i}. \tag{1}$$

The incentive level is an estimate of the contribution of a user. A serving user estimates incentive levels of requesters before serving them. Specifically, the resource is allocated to requesters in a non-increasing order of their incentive levels. If a tie occurs and the available resource is insufficient to meet the demand of tied requesters, the resource is equally allocated between them. The total resources allocated to user i from serving users determine the user's utility.

Utility. The utility u_i of user i is equal to the portion of the total allocated resources r_i that the user receives. User i may lack the download capacity d_i to receive all the allocated resources, *i.e.*, $r_i > d_i$. Resources received cannot exceed the user's download capacity. Thus the utility of user i is given as

$$u_i = \min\left(r_i, d_i\right). \tag{2}$$

Users demand resources to maximize their utilities. User i does not demand more than the download capacity d_i (see (2)). Besides, since the incentive level is inversely proportional to the demand (see (1)), asking for more resources reduces the user's incentive level. On the other hand, demanding lesser resources leads to the underutilization of the download capacity of a user. Hence, we assume that a user's demand is equal to the download capacity. Accordingly, the incentive level of user i in (1) can be rewritten as $I_i = \frac{s_i}{d_i}$.

[1]The focus of this paper is to provide an optimal capacity partitioning strategy for a given value of C_i. Thus we do not consider the specifics of the modeling of path loss and fading. The value of C_i captures wireless characteristics through bandwidth and SNR.

III. Resource Maximization game

The utility of a user depends on the upload capacity of the user and the capacity shared by other users (constituting resources available for download). We model this strategic interaction between users as a resource maximization game to analyze their strategies. Users play this game repeatedly to receive resources from the network.

Let $G = [\mathcal{N}, \{\mathcal{S}_i\}, \{\mathcal{U}_i\}]$ denote a resource maximization game, where $\mathcal{N} = \{1, 2, \cdots, N\}$ represents the set of users, $\mathcal{S}_i$ and $\mathcal{U}_i$ are the sets of possible strategies and utilities for user i, respectively. The upload capacity of user i represents the user's strategy, *i.e.*, $s_i \in \mathcal{S}_i$ and $s_i = [0, C_i],\ \forall i \in \mathcal{N}$. The notation $u_i(s_i, \mathcal{S}_{-i}) \in \mathcal{U}_i$ denotes the utility received by user i on employing the strategy s_i when other users play the strategy $\mathcal{S}_{-i} = \{s_k \mid s_k \in \mathcal{S}_k,\ \forall k \in \mathcal{N}\backslash\{i\}\}$.

A. Nash Equilibria

A strategy profile $\mathcal{S}^* = \{s_j^* \mid s_j^* \in \mathcal{S}_j,\ \forall j \in \mathcal{N}\}$ is an NE if no user can increase the utility by unilaterally changing the strategy.

Theorem 1. *In our resource maximization game, the following* NE exist.
1. Every user allocates zero capacity towards upload, *i.e.*, $\mathcal{S}^* = \{s_j \mid s_j = 0, \forall j \in \mathcal{N}\}$. Let us denote this NE by $\mathcal{S}^*_{\text{zero}}$. The utility of every user in $\mathcal{S}^*_{\text{zero}}$ is zero, *i.e.*, $u_i = 0,\ \forall i \in \mathcal{N}$.
2. Every user allocates half of the link capacity towards upload, *i.e.*, $\mathcal{S}^* = \left\{s_j \mid s_j = \frac{C_j}{2}, \forall j \in \mathcal{N}\right\}$. Let us denote this NE by $\mathcal{S}^*_{\text{hlf}}$. The utility earned by every user in $\mathcal{S}^*_{\text{hlf}}$ is half of the link capacity, *i.e.*, $u_i = \frac{C_i}{2},\ \forall i \in \mathcal{N}$ in $\mathcal{S}^*_{\text{hlf}}$.

Proof: Let S_{N} and D_{N} denote the total network upload and download capacities, *i.e.*, $S_{\text{N}} \triangleq \sum_{j\in\mathcal{N}} s_j$ and $D_{\text{N}} \triangleq \sum_{j\in\mathcal{N}} d_j$. Three relationships are possible between S_{N} and D_{N}: $S_{\text{N}} < D_{\text{N}}$, $S_{\text{N}} = D_{\text{N}}$, and $S_{\text{N}} > D_{\text{N}}$. We now explore the existence of an NE in each case.

Case 1. $\boldsymbol{S_{\text{N}} < D_{\text{N}}}$ **:** In this case, we need to consider two possibilities—either the network has no resources ($S_{\text{N}} = 0$) or resources are available ($S_{\text{N}} \neq 0$).

Case 1.1 "$S_{\text{N}} = 0$" has an NE.
The case $S_{\text{N}} = 0$ implies that the strategy profile $\mathcal{S}^*_{\text{zero}} = \{s_j \mid s_j = 0, \forall j \in \mathcal{N}\}$ exists in the network. Since no user uploads, we have

$$u_i(s_i, \mathcal{S}_{-i}) = 0, \quad \forall i \in \mathcal{N}. \tag{3}$$

Even if any user unilaterally deviates and uploads, the user does not receive any resources as other users are not uploading. Thus the user's utility remains zero. Since no user has an incentive to deviate, $\mathcal{S}^*_{\text{zero}}$ is an NE.

Case 1.2 "$S_{\text{N}} \neq 0$" has no NE.
The case $S_{\text{N}} \neq 0$ implies that non-zero resources are insufficient to meet download requirements of all users as $S_{\text{N}} < D_{\text{N}}$. At least one user in the network receives resources less than

the download capacity. This user can increase the incentive level by increasing the upload capacity. This, in turn, increases resources that the user receives and hence the utility of the user. Since the user receives a higher utility on deviation, an NE does not exist in this case.

Case 2. $S_{\mathrm{N}} = D_{\mathrm{N}}$: In this case, we need to consider the following two scenarios.

2.1 "$\forall i \in \mathcal{N} : s_i = d_i$" has an NE.
When upload and download capacities of each user are equal, the strategy profile becomes $\mathcal{S}^*_{\text{hlf}} = \left\{ s_j \mid s_j = \frac{C_j}{2}, \forall j \in \mathcal{N} \right\}$. We can calculate the change in the utility when a user changes the strategy unilaterally. By computing this change, we can determine whether $\mathcal{S}^*_{\text{hlf}}$ is an NE or not. User i can either increase or decrease the upload capacity to change the strategy. Since we consider a unilateral deviation, strategies of other users remain unchanged and are denoted by $\frac{\mathcal{C}_i}{2}_{-i} = \{s_j \mid s_j = \frac{C_j}{2},\ \forall j \in \mathcal{N}\backslash\{i\}\}$. Under the unilateral deviation, user i has three strategies available: 1) $s_i = \frac{c_i}{2}$, 2) $s_i > \frac{c_i}{2}$, or 3) $s_i < \frac{c_i}{2}$. The utility u_i of user i under these strategies are represented by u_i^1, u_i^2, and u_i^3, respectively, and are calculated as follows:

- u_i^1 *calculation:* For strategy $s_i = \frac{C_i}{2}$, the following capacity partitioning exists for all users in the network:

$$s_j = d_j = \frac{C_j}{2}, \quad \forall j \in \mathcal{N}. \tag{4}$$

Thus the total resources shared S_{N} and demanded D_{N} across the network are $S_{\mathrm{N}} = D_{\mathrm{N}} = \sum_{j \in \mathcal{N}} \frac{C_j}{2}$. Since there is a supply and demand balance across the network, the resource received by user i is equal to the download capacity d_i. Hence the utility of user i is

$$u_i^1 = u_i\left(\frac{C_i}{2}, \frac{\mathcal{C}_i}{2}_{-i}\right) = \frac{C_i}{2}. \tag{5}$$

- u_i^2 *calculation:* When user i plays strategy $s_i > \frac{C_i}{2}$, we have

$$s_i = \frac{C_i}{2} + \Delta_2,\ d_i = \frac{C_i}{2} - \Delta_2 \quad \text{for } \Delta_2 > 0$$

and

$$s_j = d_j = \frac{C_j}{2}, \quad \forall j \in \mathcal{N}\backslash\{i\}.$$

The incentive levels of users are $I_i > 1$ and $I_j = 1$, $\forall j \in \mathcal{N}\backslash\{i\}$. Due to the highest incentive level, user i gets priority over other users during resource allocation. User i may be eligible for more resources, but the download capacity $\left(d_i = \frac{C_i}{2} - \Delta_2\right)$ is smaller compared to the previous case $\left(d_i = \frac{C_i}{2}\right)$. As the user cannot receive resources higher than the download capacity, the utility received by user i is

$$u_i^2 = u_i\left(\frac{C_i}{2} + \Delta_2, \frac{\mathcal{C}_i}{2}_{-i}\right) = \frac{C_i}{2} - \Delta_2. \tag{6}$$

Comparing (5) and (6), we get $u_i^2 < u_i^1$. Since the strategy $s_i = \frac{C_i}{2}$ provides a higher utility than $s_i > \frac{C_i}{2}$, the strategy $s_i > \frac{C_i}{2}$ does not lead to an NE.

- u_i^3 *calculation:* When user i employs the strategy $s_i < \frac{C_i}{2}$,

$$s_i = \frac{C_i}{2} - \Delta_3,\ d_i = \frac{C_i}{2} + \Delta_3 \quad \text{for } \Delta_3 > 0 \tag{7}$$

and

$$s_j = d_j = \frac{C_j}{2}, \quad \forall j \in \mathcal{N}\backslash\{i\}. \tag{8}$$

This results in $I_i < 1$ and $I_j = 1,\ \forall j \in \mathcal{N}\backslash\{i\}$. Since $I_i < I_j,\ \forall j \in \mathcal{N}\backslash\{i\}$, user i is given the lowest priority during the distribution of available resources. The total available and demanded resources across the network are (see (7) and (8))

$$S_{\mathrm{N}} = \sum_{j \in \mathcal{N}} \frac{C_j}{2} - \Delta_3 \text{ and } D_{\mathrm{N}} = \sum_{j \in \mathcal{N}} \frac{C_j}{2} + \Delta_3.$$

Due to higher incentive levels of users $j \in \mathcal{N}\backslash\{i\}$, S_{N} is first distributed among them, and the remaining resources are awarded to user i. Resources consumed by users excluding user i are $D_{\mathrm{N}-\{i\}} = \sum_{j \in \mathcal{N}\backslash\{i\}} d_j = \sum_{j \in \mathcal{N}\backslash\{i\}} \frac{C_j}{2}$. Thus resources available for user i to download are $S_{\mathrm{N}} - D_{\mathrm{N}-\{i\}} = \frac{C_i}{2} - \Delta_3$. In the current case, although the download capacity of user i has increased $\left(d_i = \frac{C_i}{2} + \Delta_3\right)$ compared to the previous cases, available resources are reduced. Hence the utility of user i is

$$u_i^3 = u_i\left(\frac{C_i}{2} - \Delta_3, \frac{\mathcal{C}_i}{2}_{-i}\right) = \frac{C_i}{2} - \Delta_3. \tag{9}$$

Comparing (5) and (9), we get $u_i^3 < u_i^1$.

Since u_i^1 is higher than both u_i^2 and u_i^3, user i has no incentive to unilaterally deviate from the strategy $s_i = \frac{C_i}{2}$. Hence, $\mathcal{S}^*_{\text{hlf}} = \left\{ s_j \mid s_j = \frac{C_j}{2}, \forall j \in \mathcal{N} \right\}$ is an NE.

2.2 "$\exists i \in \mathcal{N} : s_i \neq d_i$" has no NE.
In this case, there is at least one user whose upload capacity is not equal to the download capacity. To maintain $S_{\mathrm{N}} = D_{\mathrm{N}}$, the network will contain at least one user j whose $s_j < d_j$ and a user i with $s_i > d_i$. This establishes that there are some higher $(s_i > d_i)$ and lower $(s_j < d_j)$ incentive users in the network. Besides, since $S_{\mathrm{N}} = D_{\mathrm{N}}$, every user receives resources equal to the download capacity. Thus users with smaller incentive levels also receive resources. Users with higher incentive levels can acquire resources of those with lower incentive levels by increasing download capacities. These users will continue to increase their download capacities until their incentive levels remain higher than those of lower incentive users. As higher incentive-level users benefit from the deviation, this case is not an NE.

Case 3. $S_{\mathrm{N}} > D_{\mathrm{N}}$ has no NE.
Here, $S_{\mathrm{N}} - D_{\mathrm{N}}$ resources remain unutilized after fulfilling the download requirement of every user. Any user can receive these unutilized resources by increasing the download capacity. As every user has an incentive to deviate, no NE exists. ■

B. Equilibria Analysis

We have shown that two NE exist, namely, $\mathcal{S}^*_{\text{zero}}$ and $\mathcal{S}^*_{\text{hlf}}$. If the network is in $\mathcal{S}^*_{\text{zero}}$ state, every user receives zero utility. So they will start leaving the network. In real life, $\mathcal{S}^*_{\text{zero}}$ is infeasible, while $\mathcal{S}^*_{\text{hlf}}$ is feasible. The analysis in the proof of Theorem 1 is vital in a distributed setting, where one cannot be sure of a user's capacity partitioning since each user is free to decide the partition. Suppose a user needs more resources

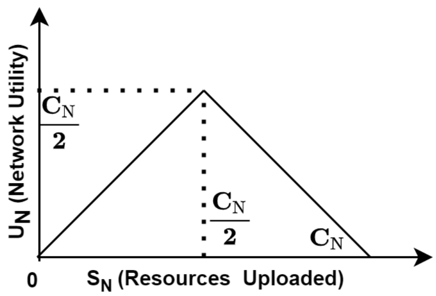

Fig. 1: Aggregate network utility variation with aggregate resources uploaded.

at a particular instant. Then the user may start allocating more portion of the link capacity towards download. After some time, the user will have to allocate more capacity for upload to compensate for extra download and to avoid decrease in the incentive level. To receive the maximum utility, whatever may be the instantaneous changes in capacity partitioning, its mean value over a long duration converges to $\mathcal{S}^*_{\text{hlf}}$.

IV. Maximizing aggregate utility of all users

Users in an NE consider only their benefits. Such a behavior may not yield the best outcome. During the best outcome, resources are divided between all users to maximize their aggregate utility. An NE of a resource maximization game may act as an upper bound if it is the best outcome. To check this, we calculate the aggregate utility received during the best outcome and compare it with the aggregate utility received during an NE.

Let U_{N}, S_{N}, D_{N}, and C_{N} denote the aggregate utility, upload, download, and link capacities, respectively across the network. Note that $C_{\text{N}} \triangleq \sum_{j\in\mathcal{N}} C_j$ is fixed because user j cannot change the link capacity C_j. As s_j (upload capacity) and d_j (download capacity) are not fixed for user j, $S_{\text{N}} \triangleq \sum_{j\in\mathcal{N}} s_j$ and $D_{\text{N}} \triangleq \sum_{j\in\mathcal{N}} d_j$ can vary within the constraint $S_{\text{N}} + D_{\text{N}} = C_N$. Here S_{N} contributes to download resources. The amount of resources downloaded cannot exceed the download capacity D_{N}. Thus, the aggregate network utility is given by $U_{\text{N}} = \min\{S_{\text{N}}, D_{\text{N}}\}$. Substituting $D_{\text{N}} = C_{\text{N}} - S_{\text{N}}$ in this equation, the problem of finding the peak of U_{N} reduces to the following optimization problem:

$$\begin{aligned} &\max\ U_{\text{N}}, \\ &\text{s.t. } \mathrm{S_N} \le \mathrm{C_N},\ \mathrm{U_N} \le \mathrm{S_N},\ \text{and } \mathrm{U_N} \le \mathrm{C_N} - \mathrm{S_N}. \end{aligned} \tag{10}$$

Solving this problem, we get the maximum utility as $C_{\text{N}}/2$. To find the maximum U_{N}, we observe its variation with S_{N} as shown in Fig. 1. As S_{N} varies in the range $\left[0, \frac{C_{\text{N}}}{2}\right)$, U_{N} increases linearly. It achieves the peak value of $C_{\text{N}}/2$ at $S_{\text{N}} = C_{\text{N}}/2$. Thereafter, U_{N} linearly decreases and becomes 0 at $S_{\text{N}} = C_{\text{N}}$. The initial increase in U_{N} occurs because network resources increase with S_{N}. After $S_{\text{N}} = \frac{C_{\text{N}}}{2}$, $D_{\text{N}} = C_{\text{N}} - S_{\text{N}}$ becomes smaller than S_{N}. Thus, in this region, $U_{\text{N}} = D_{\text{N}}$, and it decreases with an increase in S_{N}.

The network utility achieved in the states $\mathcal{S}^*_{\text{zero}}$ and $\mathcal{S}^*_{\text{hlf}}$ are 0 and $\frac{C_{\text{N}}}{2}$, respectively (refer Theorem 1), whereas the maximum achievable network utility is $\frac{C_{\text{N}}}{2}$. Hence users in the state $\mathcal{S}^*_{\text{hlf}}$ receive the maximum utility. Thus, $\mathcal{S}^*_{\text{hlf}}$ can act as an upper bound on the utility that can be received in the network. We now show how the state $\mathcal{S}^*_{\text{hlf}}$ is achieved.

Algorithm 1 Link Capacity Partition Strategy

```
repeat
    if (r_i^t < d_i^t) then
        s_i^{t+1} ← s_i^t + δ_i^t
    else if (r_i^t > d_i^t) then
        s_i^{t+1} ← s_i^t − δ_i^t
    else if (r_i^t == d_i^t) then
        s_i^{t+1} ← s_i^t − δ_i^t
    end if
    d_i^{t+1} ← C_i − s_i^{t+1}
    t ← t + 1
    return s_i^{t+1} and d_i^{t+1}
until User i exits the network
```

V. Algorithm for resource maximization game

We now provide Algorithm 1 that users can use to achieve the NE $\mathcal{S}^*_{\text{hlf}}$ in a practical network. Algorithm 1 runs independently at each user. It describes how a user continuously adjusts capacity partitioning such that it finally converges to the partitioning value in the state $\mathcal{S}^*_{\text{hlf}}$.

The network is assumed to advance in intervals of fixed time duration. During any time interval t, user i adjusts the partitioning of the link capacity C_i between upload s_i^t and download d_i^t capacities by amount δ_i^t. Here δ_i^t is decided by capacity partitioning algorithms discussed in the next section. The adjustment is done in such a way that it maximizes the utility u_i^t received by user i at time t. As u_i^t depends on resources allocated r_i^t and the download capacity d_i^t (refer (2)), Algorithm 1 progresses in each time interval by comparing r_i^t and d_i^t. If $r_i^t < d_i^t$, to increase the utility, r_i^t needs to be increased (refer (2)). r_i^t can be increased by increasing the incentive level, in turn by increasing the contribution s_i^t level. Thus s_i^t is increased by δ_i^t for the next period. If $r_i^t > d_i^t$, user i does not have enough download capacity to receive all allocated resources. To increase the utility, the download capacity d_i^t needs to be increased. As $d_i^t + s_i^t = C_i$, s_i^t is reduced by δ_i^t for the next period. If $r_i^t = d_i^t$, user i further tries to enhance the utility by increasing the download capacity. Unlike the earlier two scenarios, when $r_i^t = d_i^t$, the utility does not always increase with a change in s_i; it may also decrease depending on the capacity allocated by other users. Among many capacity allocation scenarios possible, utilities of all users simultaneously decrease for any deviation when the aggregate utility $\sum_{i\in\mathcal{N}} u_i$ of the network attains the maximum value, which happens at an NE (see Section IV).

Let the network operate close to the state $\mathcal{S}^*_{\text{hlf}}$ at time t. If user i decreases the upload capacity, the user's incentive level becomes smaller than those of other users. Thus the user receives a smaller utility in the next period. Then the user again increases the upload capacity and returns to the state $\mathcal{S}^*_{\text{hlf}}$. After reaching $\mathcal{S}^*_{\text{hlf}}$ in $t+2$ period, the user again decreases the upload capacity in subsequent rounds as $r_i^{t+2} = d_i^{t+2}$.

This process repeats, and the capacity partitioning value of the user oscillates around the value in $\mathcal{S}^*_{\text{hlf}}$. Hence, when a user employs Algorithm 1, initially, the capacity partition between upload and download changes until the network reaches the state $\mathcal{S}^*_{\text{hlf}}$. At $\mathcal{S}^*_{\text{hlf}}$, the user's capacity partition marginally stabilizes and starts oscillating around its optimal value at $\mathcal{S}^*_{\text{hlf}}$. Thus Algorithm 1 continues until the user is in the network. The instantaneous value of the capacity partitioning of a user may not always be equal to the value at $\mathcal{S}^*_{\text{hlf}}$, but when it is averaged over a long period, it approaches the value at $\mathcal{S}^*_{\text{hlf}}$. This phenomenon is later verified by simulation results.

During one time interval, Algorithm 1 performs one comparison and executes three addition/subtraction operations to update three variables. For a single interval, the complexity of this algorithm is $\mathcal{O}(1)$ for each user.

VI. Performance of Partitioning Algorithms

In real-life networks, users employ capacity partitioning algorithms [4], [5] that adjust the step size (δ_i^t in Algorithm 1) during the link capacity partition. These algorithms progressively modify the user's upload capacity until the utility received is maximum. A user has two choices to modify the partition of the link capacity: fixed step size and adaptive step size. Thus there can be two representative algorithms for simulations: Fixed Step Size (FSZ) [4] and Adaptive Step Size (ASZ) [5]. We use the NE $\mathcal{S}^*_{\text{hlf}}$ as a baseline to analyze the performance of FSZ and ASZ algorithms.

Let us consider a network consisting of 100 users. They are equally divided into 5 groups of different link capacities, namely, $4, 5, 6, 7$, and 8 Mb/s. The network is simulated for 2000 rounds. At the end of 500th, 1000th, and 1500th round, 20 new users join the network. After joining, an acquaintance period of 20 rounds is provided to a user. During this period, irrespective of the upload amount, the user is given a fixed initial reputation,[2] *i.e.*, the user is awarded resources independent of the contribution level. The rest of the simulation setup is in line with models used in [4], [5]. We set the step size in the FSZ algorithm as $\frac{C_i}{10}$, $\forall i \in \mathcal{N}$. In [4], for this value, a good trade-off between the performance and the convergence rate has been observed. In contrast, the ASZ algorithm modifies the partitioning of the link capacity in an adaptive step size.

The capacity partitioning data are recorded in Table I. The column about "Nash" depicts the state $\mathcal{S}^*_{\text{hlf}}$, while the columns corresponding to the algorithms ASZ and FSZ present their average values of capacity partitioning. The value $|\Delta|$ in a square bracket denotes the difference between upload and download. At $\mathcal{S}^*_{\text{hlf}}$, the upload and download capacities are equal. Thus, lower the value of $|\Delta|$ for a partitioning scheme, closer that algorithm's output to $\mathcal{S}^*_{\text{hlf}}$, and thus higher the utility received. The partition achieved by ASZ is much closer to $\mathcal{S}^*_{\text{hlf}}$ than that by FSZ. This results in a higher utility achieved by ASZ for the same link capacity. This fact is verified by the utility output of these algorithms in Table I. The FSZ performance is worse than that of ASZ, because the former modifies the capacity partitioning in fixed-sized steps. Thus once a user reaches the state $\mathcal{S}^*_{\text{hlf}}$, the capacity partitioning oscillates[3] around $\mathcal{S}^*_{\text{hlf}}$ with the magnitude proportional to the step size. This decreases the utility. The ASZ uses dynamically adjusted step sizes, whose value approaches zero as the user's partitioning value approaches $\mathcal{S}^*_{\text{hlf}}$. Hence the amplitude of oscillations approaches zero at $\mathcal{S}^*_{\text{hlf}}$. In this way, the ASZ achieves a higher utility that for FSZ.

TABLE I: Comparison of partitioning algorithms (FSZ, ASZ)

Link Capacity (Mb/s)	Upload, download capacities (Mb/s) [Deviation $\|\Delta\|^*$]			Utility (Mb/s)		
	Nash	FSZ	ASZ	Nash	FSZ	ASZ
4	2,2 [0]	3.08, 0.91 [2.17]	2.03, 1.97 [0.06]	2	0.78	1.54
5	2.5,2.5 [0]	4.04, 0.96 [3.08]	2.57, 2.43 [0.14]	2.5	0.86	2.01
6	3,3 [0]	4.91, 1.09 [3.82]	3.11, 2.89 [0.22]	3	0.98	2.52
7	3.5,3.5 [0]	5.76, 1.24 [4.52]	3.69, 3.31 [0.38]	3.5	1.13	3.00
8	4,4 [0]	6.68, 1.32 [5.36]	4.37, 3.63 [0.74]	4	1.23	3.32

* Deviation $|\Delta|$ from the NE = |Upload Capacity - Download Capacity|

VII. Conclusions

From the two Nash equilibria (NE) we found, only one is usually practically feasible, where all users divide their link capacities equally between upload and download. In this NE state, users receive maximum resources. This NE can be used as a benchmark to compare the performance of various partitioning algorithms, such as Adaptive Step Size (ASZ) and Fixed Step Size (FSZ) algorithms. We observe that the ASZ algorithm performs better than the FSZ algorithm.

References

[1] K. Shin, C. Joe-Wong, S. Ha, Y. Yi, I. Rhee, and D. S. Reeves, "T-chain: A general incentive scheme for cooperative computing," *IEEE/ACM Trans. Netw.*, vol. 25, pp. 2122–2137, Aug. 2017.

[2] C. Joe-Wong, Y. Im, K. Shin, and S. Ha, "A performance analysis of incentive mechanisms for cooperative computing," in *IEEE International Conference on Distributed Computing Systems*, 2016, pp. 108–117.

[3] Y. Zhang and M. Guizani, Eds., *Game theory for wireless communications and networking*. Boca Raton, Florida : CRC Press, 2011.

[4] A. Satsiou and L. Tassiulas, "Reputation-based resource allocation in P2P systems of rational users," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, pp. 466–479, Apr. 2010.

[5] N. Singha, Y. N. Singh, and R. Gupta, "Adaptive capacity partitioning in cooperative computing to maximize received resources," *IEEE Access*, vol. 8, pp. 3551–3565, 2020.

[6] M. Meo and F. Milan, "A rational model for service rate allocation in peer-to-peer networks," *in IEEE INFOCOM*, pp. 1–5, 2006.

[7] C. H. Chiang, W. Liao, and T. Liu, "Adaptive downlink/uplink bandwidth allocation in IEEE 802.16 (WiMAX) wireless networks: A cross-layer approach," *in IEEE GLOBECOM*, pp. 4775–4779, 2007.

[8] D. G. Jeong and W. S. Jeon, "CDMA/TDD system for wireless multimedia services with traffic unbalance between uplink and downlink," *IEEE J. Sel. Areas Commun.*, vol. 17, pp. 939–946, May 1999.

[9] R. Maheswaran and T. Basar, "Efficient signal proportional allocation (ESPA) mechanisms: decentralized social welfare maximization for divisible resources," *IEEE J. Sel. Areas Commun*, vol. 24, pp. 1000–1009, 2006.

[10] A. Goswami, R. Gupta, and G. S. Parashari, "Reputation-based resource allocation in P2P systems: A game theoretic perspective," *IEEE Commun. Lett.*, vol. 21, pp. 1273–1276, 2017.

[2] The initial reputation is the average reputation of all users in the network.

[3] As shown in Section V, oscillations occur because the user further wants to enhance the utility at $\mathcal{S}^*_{\text{hlf}}$.