Incentive and Service Differentiation in P2P Networks: A Game Theoretic Approach

Richard T. B. Ma, Sam C. M. Lee, John C. S. Lui, *Senior Member, IEEE*, and David K. Y. Yau, *Member, IEEE*

***Abstract*—Conventional peer-to-peer (P2P) networks do not provide service differentiation and incentive for users. Therefore, users can easily obtain information without themselves contributing any information or service to a P2P community. This leads to the well known *free-riding* problem. Consequently, most of the information requests are directed towards a small number of P2P nodes which are willing to share information or provide service, causing the "tragedy of the commons." The aim of this paper is to provide service differentiation in a P2P network based on the amount of services each node has provided to the network community. Since the differentiation is based on nodes' prior contributions, the nodes are encouraged to share information/services with each other. We first introduce a resource distribution mechanism for all the information sharing nodes. The mechanism is distributed in nature, has linear time complexity, and guarantees *Pareto-optimal* resource allocation. Second, we model the whole resource request/distribution process as a competition game between the competing nodes. We show that this game has a Nash equilibrium. To realize the game, we propose a protocol in which the competing nodes can interact with the information providing node to reach Nash equilibrium efficiently and dynamically. We also present a generalized incentive mechanism for nodes having *heterogeneous* utility functions. Convergence analysis of the competition game is carried out. Examples are used to illustrate that the incentive protocol provides service differentiation and can induce productive resource sharing by rational network nodes. Lastly, the incentive protocol is adaptive to node arrival and departure events, and to different forms of network congestion.**

***Index Terms*—Contribution-based service differentiation, game theory, incentive protocol, peer-to-peer network.**

I. Introduction

THERE HAS BEEN a lot of recent interest in peer-to-peer (P2P) networks. As evidenced by traffic measurements of ISPs, a large percentage of existing network traffic is due to P2P applications [4], [10]. These applications aim to exploit the cooperative paradigm of information exchange to greatly increase the accessibility of information to a large population of network users. Unlike traditional client–server networking, the P2P paradigm allows individual users (or *nodes*) to play the roles of both server and client at the same time. Therefore, nodes in a P2P network can assist each other in file searching, file lookup [14], [15], [18], [20] and file transfer in an anonymous manner [3]. For file searching, P2P networks have evolved from a centralized file/directory lookup approach (e.g., Napster) to a distributed object query approach (e.g., Gnutella). Whereas distributed object queries can be effected by some form of controlled flooding, the new generation of P2P networks (e.g., Chord and CAN) use the method of *consistent hashing* to improve the efficiency of file lookup.

While much current research focuses on improving the performance of file searching/lookup in P2P networks, some fundamental and challenging issues remain unanswered about the basic cooperative paradigm of information exchange. *Free riding* and the *tragedy of the commons* are two such problems. It is reported in [1] that nearly 70% of P2P users do not share any file in a P2P community. Instead, these users simply free ride on other users who share information. Since there are few users who are willing to share information or provide services to others, nearly 50% of all file search responses come from the top 1% of information sharing nodes. Therefore, nodes that share information and resources are prone to congestion, which leads to the tragedy of the commons problem [9]. In short, existing P2P networks do not provide *service differentiation*; hence, there is no *incentive* for users to share information or provide file transfer services.

More recently, there are some preliminary mechanisms implemented in P2P software that encourage people to share information. For example, Kazaa [10] considers the "participation level" of each peer. The participation level is calculated as the ratio between a peer's recent uploads and downloads. However, this ratio is not accumulated over time, and provides differentiation for query requests only. Another P2P system, eMule [4], establishes a credit system in which credits are exchanged between any two specific nodes. In allocating its upload resources to competing requesting peers, an information providing node gives smaller queueing delays for the peers which have previously provided more uploads to the information providing node. No formal analysis of their mechanisms, including fairness for competing nodes, has been given for Kazaa and eMule.

In this paper, we propose a protocol to provide service differentiation based on the contribution levels of individual nodes. Our protocol targets the file transfer process, because the amount of data transferred per unit time is much higher than that of object lookup/query. In this context, a node which offers popular files for sharing and provides more service (via file upload) to the P2P community will achieve a higher contribution level. As a result, when such a node later asks for a file transfer, it will be granted a higher utility than competing nodes

Manuscript received November 2, 2004; revised August 8, 2005; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor J. Byers. This work was supported in part by the National Science Foundation under grant numbers CCR-9875742 and CNS-0305496. A preliminary version of this paper appeared in ACM SIGMETRICS 2004, New York, NY.

R. T. B. Ma, S. C. M. Lee, and J. C. S. Lui are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong (e-mail: tbma@cse.cuhk.edu.hk; cmlee@cse.cuhk.edu.hk; cslui@cse.cuhk.edu.hk).

D. K. Y. Yau is with the Department of Computer Science, Purdue University, West Lafayette, IN 47907 USA (e-mail: yau@cs.purdue.edu).

Digital Object Identifier 10.1109/TNET.2006.882904

having lower contribution levels. We address the challenges of incorporating such *incentive-compatible* resource distribution in the file transfer process such that we can: 1) encourage nodes to share information with or provide services for their peers; 2) achieve fair service differentiation between network users; and 3) maximize the social welfare [16] or the aggregate perceived utility of the users. It is important to point out that our incentive protocol can be adopted by various P2P systems which use either the distributed query (e.g., Gnutella) or the consistent hashing approach (e.g., Chord and CAN).

The proposed incentive-compatible resource distribution mechanism has the following properties:

1. Fairness: Nodes which have contributed more to the P2P network should gain more resources or achieve higher utility in the resource sharing.

2. Avoidance of resource wastage: The mechanism will not assign more resource to a node than it can consume. In case there is congestion in the communication path, the mechanism can adapt to the congestion level and re-distribute the resources accordingly.

3. Adaptability and Scalability: The mechanism can adapt to dynamic events such as node join/leave. Since the mechanism runs at each participating node, its performance is scalable as the size of the P2P network increases.

4. Maximization of individual and social utility: Each node is motivated to follow the incentive protocol in order to maximize its own utility in the game. In addition, the resource distribution maximizes the aggregate perceived utility from the point of view of an information sharing node.

As we will show, the proposed mechanism makes different requesting users bid for resources, thereby creating a *dynamic competitive game*. In order to assure that every node in the P2P network will follow the mechanism honestly, the dynamic game created should be *strategic-proof* and *κ-collusion-proof*. The first property implies that following the proposed mechanism is the *best* strategy for each user in the network. The second property implies that users cannot gain extra resources by cooperatively deceiving the system.

A. Related Work

Let us briefly present some related work. In [7], the authors address one possible mechanism for *centralized* P2P systems like Napster. Our work, on the other hand, can be applied to both centralized and distributed P2P networks. Zhong *et al.* [21] discuss shortcomings of *micro-payment* and reputation systems. They propose a cheat-proof, credit-based mechanism for mobile ad hoc networks. However, they do not address how to provide incentive and service differentiation in the P2P setting. In [6], the authors discuss the economic behavior of P2P storage networks. Our work, on the other hand, focuses on file-transfer and bandwidth allocation in a P2P network; we use the approach of *mechanism design* to create a desirable competition game for the network. In [11], the authors propose a budget-balance virtual money exchange mechanism to bring incentive into P2P networks. Our work uses a game-theoretic approach, and provides a stronger solution concept of *incentive compatibility*. Lastly, algorithmic mechanism design [13], [17] provides a theoretical framework for designing incentive mechanisms.

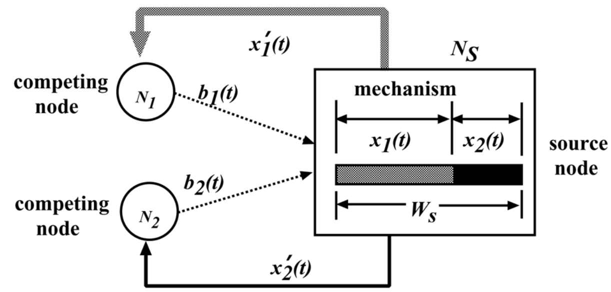


Fig. 1. Illustrating two competing nodes and a source node.

B. Paper Organization

The rest of this paper is organized as follows. In Section II, we give an overview of the interactions between an information providing node and other nodes requesting the information. In Section III, we present the resource distribution mechanism and its properties. In Section IV, we model the resource distribution as a dynamic game, and show how it can be applied in a P2P network. In Section V, we present a generalized mechanism that handles incentive for nodes with heterogeneous perceived utilities. Convergence analysis of the dynamic game is presented in Section VI. In Section VII, we present a performance evaluation of the proposed mechanism and competition game. Section VIII concludes.

II. Incentive P2P System Overview

Let us provide an overview of our incentive P2P system. In particular, we illustrate the interactions between different nodes during the file transfer process. In later sections, we will formally present the development of the resource distribution mechanism and its properties.

Each node in our incentive P2P network can play the roles of a server and a client at the same time. During a file transfer, the node which performs the service (i.e., uploading files to other nodes) is called the *source node*, which is denoted by $\mathcal{N}_s$. Nodes which request file download from $\mathcal{N}_s$ are called the *competing nodes*, which are denoted as $\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_N$, where N is the number of competing nodes. Each node in our incentive P2P network has a contribution value, which indicates how much service that node has provided to the whole P2P community. Due to the lack of space, we will not discuss architecture issues in realizing a scalable and secure contribution value reporting system. Please refer to [8] for related discussions.

A scenario, in which two competing nodes $\mathcal{N}_1$ and $\mathcal{N}_2$ request file download service from the source node $\mathcal{N}_s$, is illustrated in Fig. 1. The source node has an upload bandwidth resource of $\mathcal{W}_s$ (in units of bits/s). From time to time, these competing nodes send messages $b_1(t)$ and $b_2(t)$ (in units of bits/s) to $\mathcal{N}_s$, telling $\mathcal{N}_s$ how much transfer bandwidth they want. Upon receiving these messages, $\mathcal{N}_s$ will use a resource distribution mechanism (to be presented in Section III) to distribute its bandwidth resource $\mathcal{W}_s$ based on the values of $b_1(t)$, $b_2(t)$, and the requesting nodes' contribution values denoted by $C_1(t)$ and $C_2(t)$, respectively. As a result, $\mathcal{N}_s$ delivers file data to $\mathcal{N}_1$ and $\mathcal{N}_2$ with bandwidth $x_1(t)$ and $x_2(t)$, respectively. However, it is possible that there is network congestion along the communication path between $\mathcal{N}_s$ to $\mathcal{N}_1$ (or $\mathcal{N}_2$). Therefore, packets may

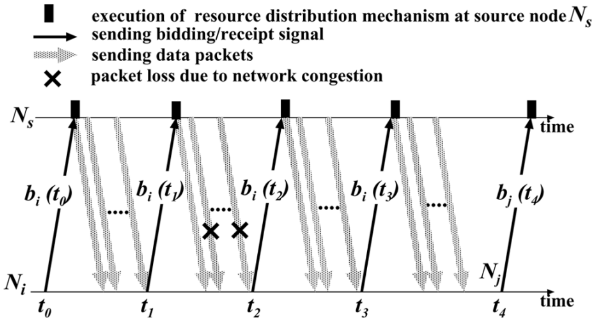


Fig. 2. Interaction between competing nodes and a source node.

be lost and the actual received bandwidths at nodes $\mathcal{N}_1$ and $\mathcal{N}_2$ are $x_1'(t) \leq x_1(t)$ and $x_2'(t) \leq x_2(t)$, respectively.

The message $b_i(t)$ plays two important roles. First, it can be regarded as a *bandwidth bidding* message from the perspective of the competing node $\mathcal{N}_i$. Another usage of $b_i(t)$ is that it is a *confirmation* to the source node $\mathcal{N}_s$ that $\mathcal{N}_i$ has received a certain amount of service (measured in units of bits/s). Therefore, $\mathcal{N}_s$ can use this message as evidence for updating its contribution. In general, the message $b(t)$ helps the source node to determine the proper bandwidth assignment. If a competing node is inactive or failed, the source node will assume that the competing node cannot receive any data. Therefore, it will not send any more packet to the competing node. The source node, on the other hand, can *adjust* the bandwidth resource assignment whenever it receives a bidding message. The justifications for this adjustment are: 1) a newly arriving competing node may request a new file download from $\mathcal{N}_s$; 2) an existing competing node finishes its file transfer service; and 3) because of network congestion, a competing node replies with different bidding message values during the file download session. To efficiently utilize the bandwidth resource $\mathcal{W}_s$ and to improve the rate of contribution increase for $\mathcal{N}_s$, the source node needs to adjust its bandwidth distribution among the competing nodes. Lastly, Fig. 2 illustrates the interactions between the competing nodes and the source node $\mathcal{N}_s$. At time t_0, the competing node $\mathcal{N}_i$ requests the transfer of a large file F_i, and sends a bidding message $b_i(t_0)$ to $\mathcal{N}_s$. After verifying the identity and contribution level of $\mathcal{N}_i, \mathcal{N}_s$ uses the resource distribution mechanism to determine the sending bandwidth $x_i(t_0)$, and delivers some data packets of F_i to $\mathcal{N}_i$ based on this rate allocation. After receiving these data packets, $\mathcal{N}_i$ sends another bidding/receipt $b_i(t_1)$ at time t_1. $\mathcal{N}_s$ then determines a new resource allocation and sends some additional data packets of file F_i based on $x_i(t_1)$. Note that in this round of the data delivery, some data packets are lost due to network congestion. Therefore, $\mathcal{N}_i$ sends a bidding/receipt $b_i(t_2)$ to $\mathcal{N}_s$ at time t_2, with $b_i(t_2) < b_i(t_3)$. The source node $\mathcal{N}_s$ adjusts the resource allocation and delivers additional data packets of file F_i to $\mathcal{N}_i$ at a lower rate. At time t_4, a new competing node $\mathcal{N}_j$ requests a transfer of the file F_j from $\mathcal{N}_s$, and sends its bidding message $b_j(t_4)$. $\mathcal{N}_s$ adjusts the resource allocation based on the latest biddings of the two competing nodes $\mathcal{N}_i$ and $\mathcal{N}_j$.

III. Resource Distribution Mechanism

In this section, we discuss how a source node, say $\mathcal{N}_s$, implements a mechanism to distribute its bandwidth resource $\mathcal{W}_s$ (in Mb/s) among all its competing nodes $\mathcal{N}_1, \ldots, \mathcal{N}_N$. For ease of presentation, we start with some simple mechanisms and discuss their shortcomings. Then we introduce more sophisticated features so as to provide service differentiation and incentive.

1) Resource Bidding Mechanisms (RBM): The objective of this mechanism is to avoid resource wastage that would occur in a naive mechanism that gives each competing node an equal resource share. Under RBM, every competing node is required to send a bidding message periodically to $\mathcal{N}_s$. Let $b_i(t)$ be the bidding message from the competing node $\mathcal{N}_i$ at time t indicating the *maximum* bandwidth (in units of bits/s) that $\mathcal{N}_i$ can receive at time t. Given all the bidding messages from the competing nodes, $\mathcal{N}_s$ has knowledge of the *upper bound* bandwidth assignments and will not assign any bandwidth higher than $b_i(t)$ to $\mathcal{N}_i$ at time t. Notice that it seems possible for some competing nodes to ask for more bandwidth than they really need; we will discuss the rational bidding values of competing nodes in Section IV.

One important property of the RBM mechanism is that it provides *max-min fairness* [2]. Suppose $\vec{x} = [x_1, \ldots, x_N]$ is the bandwidth allocation for all N competing nodes within the feasible domains $x_i \in [0, b_i]$ for $i = 1, \ldots, N$. Then a feasible allocation is max-min fair if and only if an increase of x_i within its domain of feasible allocation must be at the cost of a decrease of some x_j, where $x_j \leq x_i$. In other words, the max-min allocation gives the competing node with the smallest bidding value the largest feasible bandwidth while not wasting any resource for the source node $\mathcal{N}_s$. One can show that there exists a unique max-min fair allocation vector $\vec{x}$, and that it can be obtained by a *progressive water filling algorithm*. The algorithm initializes all $x_i = 0$. It will then increase all competing nodes' bandwidth allocations at the *same rate*, until one or several competing nodes hit their limits (i.e., $x_i = b_i$). When that happens, the amount of allocated resources for these competing nodes will not be increased any more. The algorithm will continue to increase the resources of other competing nodes at the same rate. The algorithm terminates when all the competing nodes have hit their limits, or the total resource $\mathcal{W}_s$ is fully utilized. Mathematically, we can express the max-min resource distribution as follows. Let $\mathcal{N}_{\hat{1}}, \ldots, \mathcal{N}_{\hat{N}}$ be N competing nodes sorted in non-decreasing order of b_i. The resource distribution of the RBM mechanism is

$$x_{\hat{k}} = \min\left\{b_{\hat{k}}, \frac{\mathcal{W}_s - \sum_{\hat{i}=1}^{\hat{k}-1} x_{\hat{i}}}{N - \hat{k} + 1}\right\}, \quad \hat{k} = 1, \ldots, N. \quad (1)$$

Fig. 3(a) illustrates the RBM with four competing nodes of $\vec{b} = [1, 2.5, 2.5, 4]$ (in units of Mb/s) and the resource bandwidth $\mathcal{W}_s = 7$ Mb/s. The resource allocation is $\vec{x} = [1, 2, 2, 2]$ (in units of Mb/s), which is depicted as the shaded regions in the figure. Although the RBM avoids resource wastage, it does not provide any *incentive* for nodes to share information. Two competing nodes with the same bidding values will obtain the same amount of resource regardless of their actual contributions to the P2P community.

2) Resource Bidding Mechanism With Incentive (RBM-I): To provide incentive, this mechanism takes the contribution levels of competing nodes into account. Let C_i be the contribution

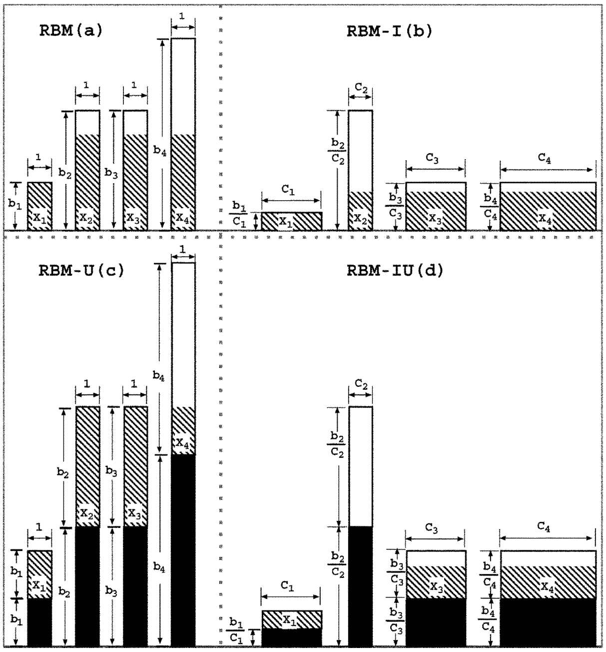


Fig. 3. Resource distribution mechanisms: (a) RBM; (b) RBM-I; (c) RBM-U; (d) RBM-IU. The shaded region represents the amount of resource allocation for individual node.

value of the competing node $\mathcal{N}_i$. This value reflects the amount of work that $\mathcal{N}_i$ has performed, for example, sharing and uploading files for other nodes. The contribution value C_i can be retrieved from a distributed database at the beginning of the file transfer process, or every time when the source node receives the bidding message $b_i(t)$ from the competing node $\mathcal{N}_i$.

One can implement the resource bidding mechanism with incentive (RBM-I) by enhancing the progressive filling algorithm as follows. We distribute bandwidth resources to all the competing nodes at the same time but at different rates. In particular, the competing node $\mathcal{N}_i$ will have a resource assignment rate of C_i. Also, once the assigned resource to $\mathcal{N}_i$ reaches its limit of b_i, $\mathcal{N}_i$ will be taken out of the resource distribution. Therefore, one can view the mechanism as a weighted max-min resource distribution. Mathematically, we can express the RBM-I as follows. Let $\mathcal{N}_{\hat{1}}, \ldots, \mathcal{N}_{\hat{N}}$ be N competing nodes sorted in non-decreasing order of b_i/C_i. The resource distribution is

$$x_{\hat{k}} = \min\left\{ b_{\hat{k}}, \frac{C_{\hat{k}}\left(\mathcal{W}_s - \sum_{\hat{i}=1}^{\hat{k}-1} x_{\hat{i}}\right)}{\sum_{j=\hat{k}}^{N} C_j} \right\}, \quad \hat{k} = 1, \ldots, N. \quad (2)$$

Using the previous example in RBM but now with contributions $\vec{C} = [2.5, 1, 2.5, 4]$, the resource allocation is $\vec{x} = [1, 0.8, 2, 3.2]$ (in units of Mb/s), which is shown in Fig. 3(b). One important property of this mechanism is that if two competing nodes have the *same* bandwidth bidding values and have not hit their limits, then the assigned bandwidth will be *proportional* to their contribution values (i.e., $\mathcal{N}_2$ and $\mathcal{N}_3$).

3) Resource Bidding Mechanism With Utility Feature (RBM-U): This mechanism addresses the efficiency of the resource allocation from the perspective of the competing nodes' satisfaction. Consider two competing nodes $\mathcal{N}_i$ and $\mathcal{N}_j$ which have the same contribution values. If the bandwidth resource at the source node is $\mathcal{W}_s = 1$ Mb/s and the two bidding messages are $b_i(t) = 10$ Mb/s and $b_j(t) = 1$ Mb/s, based on the RBM mechanism, they will receive a bandwidth resource of 0.5 Mb/s each. Although the resource at $\mathcal{N}_s$ is efficiently utilized, the degrees of satisfaction of the two competing nodes are obviously different. To overcome the problem, we use the concept of *utility* [16] to represent the degree of satisfaction of a competing node given a certain amount of allocated bandwidth.

We first define the family of utility functions we consider in this paper. Given an allocated bandwidth x, the utility of the node $\mathcal{N}_i$ is denoted by $U_i(x)$. The utility function we consider in this work satisfies the following three assumptions: 1) $U_i(x)$ is concave (or the marginal utility $dU_i(x)/dx$ is non-increasing $\forall\, x \geq 0$); 2) $U_i(0) = 0$; and 3) the utility depends on the ratio of x/b. In other words, $U_i(x_i) = U_j(x_j)$ whenever $x_i/rb_i = x_j/b_j$ for any two competing nodes $\mathcal{N}_i$ and $\mathcal{N}_j$. The justifications for the above assumptions are as follows. First, the utility function is concave, which is often used to represent *elastic traffic* such as file transfer [16]. Concavity implies that the marginal utility is non-increasing as one increases the allocated bandwidth resource x. This captures the physical characteristics of elastic traffic: the utility increases significantly when a competing node starts receiving service. The increase of utility becomes less significant when the receiving bandwidth is nearly saturated. Second, the utility is zero when a competing node is not allocated any bandwidth. Third, because utility measures the satisfaction of a competing node, naturally, it is a function of the *fraction* of the allocated resource over the bidding resource. Furthermore, this assumption normalizes the utility of all nodes so that we can *compare* the degrees of satisfaction of different nodes.

The objective of the RBM-U mechanism is to maximize the social (or aggregate) utility. Formally, we have

$$\max \sum_{i=1}^{N} U_i(x_i) \quad \text{s.t.} \quad \sum_{i=1}^{N} x_i \leq \mathcal{W}_s \text{ and } x_i \in [0, b_i] \quad \forall\, i.$$

It is important to point out that the implication of this maximization problem is to allocate resources to the competing node which currently has the *largest* marginal utility (i.e., largest $dU_i(x)/dx$). The allocation process starts with $x_i = 0$ for $i = 1, \ldots, N$. It then assigns resources to the node which has the largest marginal utility and ends when the resource $\mathcal{W}_s$ is used up, or all the competing nodes are fully satisfied with $x_i = b_i\ \forall\, i$.

In reality, the utility function of a competing node may not be known to other nodes. However, after allocating a certain amount of bandwidth to a competing node, the source node can infer the *perceived* utility of the competing node. From now on, we will use the word "utility" to indicate the "perceived utility" by the source node. We will discuss how competing nodes can maximize their underlying true utilities in the next section.

Definition 1: We define the *perceived utility* of a node, say i, by any source node to be $U_i(x_i)$, where x_i is the assigned bandwidth to node i. This is an estimate of the true utility of node i by the source node.

Let us consider the following form of perceived utility function which satisfies the above three assumptions:

$$U_i(x_i) = \log\left(\frac{x_i}{b_i} + 1\right) \qquad \text{where } x_i \in [0, b_i].$$

The marginal utility is $U_i' = (x_i + b_i)^{-1}$. Therefore, the RBM-U mechanism tries to increase the resource allocation of the competing node which has the smallest value of $x_i + b_i$ at any time. Using the previous example of RBM of four competing nodes with $\vec{b} = [1, 2.5, 2.5, 4]$ and $\mathcal{W}_s = 7$ Mb/s, we use the above perceived utility function, and the resource allocation which maximizes the aggregate utility is $\vec{x} = [1, 2.5, 2.5, 1]$ (in units of Mb/s). This result is depicted in Fig. 3(c). The figure shows graphically how the mechanism works. Each competing node, say $\mathcal{N}_i$, has a lower limit height which is equal to b_i (i.e., the darkened region). The enhanced progressive filling algorithm distributes resources first to the competing node that has the *lowest* depth since that node has the *largest* marginal utility at that point. When the assigned resource to node $\mathcal{N}_i$ is equal to the node's maximum bidding b_i, node $\mathcal{N}_i$ is taken out from the resource distribution. The algorithm terminates when all nodes have reached their maximum allocations, or when the resource $\mathcal{W}_s$ is fully utilized.

4) Resource Bidding Mechanism With Incentive and Utility Feature (RBM-IU): One can view the RBM-IU mechanism as a *generalization* of the previously discussed mechanisms. This mechanism considers both the utilities of competing nodes and their contribution values. Each competing node, say $\mathcal{N}_i$, has its contribution value C_i and bidding message b_i. Mathematically, the RBM-IU performs the following constrained optimization:

$$\max \sum_{i=1}^{N} C_i \log\left(\frac{x_i}{b_i} + 1\right) \quad \text{s.t.} \sum_{i=1}^{N} x_i \le \mathcal{W}_s,\ x_i \in [0, b_i]\ \forall\, i.$$

The RBM-IU mechanism enhances the progressive filling algorithm as follows. 1) We treat the competing node $\mathcal{N}_i$ as a bucket of area b_i and width C_i. 2) The bucket of the competing node $\mathcal{N}_i$ is located at the height b_i/C_i; therefore, the upper limit of the bucket is at a height of $2b_i/C_i$. 3) At any time, the RBM-IU mechanism gives an additional amount of resource to the competing node's bucket which currently has the lowest height—in other words, the bucket which has the largest weighted marginal utility (i.e., weighted by the contribution value). It is interesting to observe that when competing nodes have the same contribution values, the RBM-IU is equivalent to the RBM-U mechanism. The spirit of this mechanism is to increase the amount of resource of the competing node which has the largest weighted marginal utility of $C_i/(b_i + x_i)$ at a rate of C_i. Fig. 3(d) illustrates the RBM-IU mechanism with $\vec{b} = [1, 2.5, 2.5, 4]$, $\vec{C} = [2.5, 1, 2.5, 4]$ and $\mathcal{W}_s = 7$ Mb/s. The final resource allocation is $\vec{x} = [1, 0, 2.3, 3.7]$ (in units of Mb/s). From the figure, one can observe that the mechanism fills the bucket of $\mathcal{N}_i$ at most up to its area limit of b_i at the resource distribution rate of C_i. The bucket of $\mathcal{N}_i$ at the "resource level" $(x_i + b_i)/C_i$ is guaranteed to have the marginal utility $C_i/(x_i + b_i)$. The algorithm terminates when all the competing nodes have reached their resource limit, or when the resource $\mathcal{W}_s$ is fully utilized.

The RBM-IU mechanism can be expressed by the following pseudo-code. The source node $\mathcal{N}_s$ maintains a *sorted list* of competing nodes in ascending b_i/C_i order.

RBM-IU Mechanism ()

1. **if** $\left(\sum_{i=1}^{N} b_i \le \mathcal{W}_s\right)$ **return** $\vec{x} = \vec{b}$; `/* no congestion */`

2. $l = 2$; $u = 1$; `/* upper and lower limits index */`

3. $v = C_1$; $w = \mathcal{W}_s$; `/* filling rate and resource capacity */`

4. level $= \dfrac{b_1}{C_1}$; `/* initialize resource level */`

5. **while** $(w > 0)$

6.   **if** $\left(\left(\min\left\{\dfrac{2b_u}{C_u}, \dfrac{b_l}{C_l}\right\} - \text{level}\right)^{*} v \ge w\right)$

7.   level $=$ level $+ w/v$; $w = 0$;

8.   **else if** $\left(\dfrac{2b_u}{C_u} < \dfrac{b_l}{C_l}\right)$

9. $w -= \left(\dfrac{2b_u}{C_u} - \text{level}\right)^{*} v$; level $= \dfrac{2b_u}{C_u}$; $v -= C_u$; $u{+}{+}$;

10.   **else**

11. $w -= \left(\dfrac{b_l}{C_l} - \text{level}\right)^{*} v$; level $= \dfrac{b_l}{C_l}$; $v += C_l$; $l{+}{+}$;

12. **for** (each i)

13. $x_i = \min\left\{\max\left\{0, \left(\text{level}\ \dfrac{b_i}{C_i}\right) * C_i\right\}, b_i\right\}$

14. **return** $\vec{x}$;

Based on the above code, $\mathcal{N}_s$ performs the filling algorithm when the total bidding is greater than the total available resource. In determining the final "resource level", we have three cases for the *while* loop in line 5: 1) When the resource is used up, the loop ends with the final "resource level" (lines 6–7). 2) If the next available resource level is at the upper limit (or bidding level) of some competing node, we adjust the remaining amount of available resource and reduce the filling rate by that competing node's contribution value C_i, since we will not give any more resource to that satisfied competing node (lines 8–9). 3) If the next available resource level is at a lower limit of some competing node, we adjust the remaining amount of available resource and increase the filling rate by that competing node's contribution value C_i (line 11). The reason is that this competing node will have the largest weighted marginal utility for its turn to gain the resource at a rate of C_i. Note that the algorithm has a *linear* time complexity of $O(N)$, where N is the number of competing nodes at the source node $\mathcal{N}_s$. Therefore, the resource distribution can be performed efficiently.

Theorem 1: The RBM-IU mechanism solves the following constrained optimization problem:

$$\max \sum_{i=1}^{N} C_i \log\left(\frac{x_i}{b_i} + 1\right) \quad \text{s.t.} \sum_{i=1}^{N} x_i \le \mathcal{W}_s,\ x_i \in [0, b_i]\ \forall\, i.$$

Proof: Let us consider an equivalent constrained optimization problem in a standard form as follows:

$$\max \sum_{i=1}^{N} C_i \log(x_i + b_i)$$
$$\text{s.t. } \sum_{i=1}^{N} x_i \le \mathcal{W}_s, \quad x_i \le b_i \,\forall\, i \text{ and } x_i \ge 0 \,\forall\, i.$$

We have the Lagrangian function

$$L(x,\lambda)=\sum_{i=1}^{N}C_i \log(x_i+b_i)-\sum_{i=1}^{N}\lambda_i(x_i-b_i)-\lambda_0\left(\sum_{i=1}^{N}x_i-\mathcal{W}_s\right)$$

where each λ_i is the Lagrangian multiplier associated with the according "less than" constraint.

For any optimal solution x^*, the Karush–Kuhn–Tucker (KKT) condition [19] requires that there exists a non-negative Lagrangian multiplier λ such that the following conditions are satisfied:

$$\frac{\partial L}{\partial x_i} = \frac{C_i}{x_i^* + b_i} - \lambda_i - \lambda_0 = 0 \text{ (or } \le 0 \text{ if } x_i^* = 0)$$
$$\frac{\partial L}{\partial \lambda_i} = x_i^* - b_i = 0 \text{ (or } \le 0 \text{ if } \lambda_i = 0)$$
$$\frac{\partial L}{\partial \lambda_0} = \sum_{i=1}^{N} x_i^* - \mathcal{W}_s = 0 \text{ (or } \le 0 \text{ if } \lambda_0 = 0).$$

First, if $\sum_{i=1}^{N} b_i \le \mathcal{W}_s$, the RBM-IU mechanism assigns $x_i = b_i$ for all i. By checking the KKT condition, λ_0 has to be 0, and $\lambda_i = C_i/(x_i+b_i)$ for all $i > 0$. Second, if $\sum_{i=1}^{N} b_i > \mathcal{W}_s$, the RBM-IU mechanism performs the resource filling at the level $h = (x_j + b_j)/C_j$ for some j, where $0 < x_j < b_j$. It uses up all the resources so that $\sum_{i=1}^{N} x_i - \mathcal{W}_s = 0$, and λ_0 can be positive. Make $\lambda_0 = 1/h$, and the KKT condition requires

$$\frac{C_i}{x_i^* + b_i} - \lambda_i = \frac{1}{h} \text{ (or } \le \text{ if } x_i^* = 0).$$

It is satisfied in all three cases: 1) When $0 < x_i < b_i$, $\lambda_i = 0$. This is the case when all nodes get the resource at the final resource level; so $C_i/(x_i^* + b_i) = 1/h$. 2) When $x_i = 0$, node i is not filled up because b_i/C_i is higher than h. Thus, $C_i/(x_i + b_i) - 1/h \le 0 \le \lambda_i$. 3) When $x_i = b_i$, $C_i/(x_i + b_i) - 1/h = \lambda_i \ge 0$, because the resource level $(x_i + b_i)/C_i$ of node i must be less than or equal to the final resource level h.

By the strong concavity of the objective function and the linearity of all the constraints, the above KKT condition also guarantees that the solution of RBM-IU is an optimal solution for the constrained optimization problem. ■

Moreover, the following two important theorems state some desirable properties of the RBM-IU mechanism.

Theorem 2: For any two competing nodes $\mathcal{N}_i, \mathcal{N}_j$, the mechanism RBM-IU gives the bandwidth assignments x_i and x_j such that

$$\text{if } \frac{C_i}{b_i} \ge \frac{C_j}{b_j} \Longrightarrow U_i(x_i) \ge U_j(x_j). \quad (3)$$

Proof: When $C_i/b_i \ge C_j/b_j$, the stated condition is equivalent to

$$\frac{b_i}{C_i} \le \frac{b_j}{C_j}. \quad (4)$$

So initially, node $\mathcal{N}_i$ has a lower resource level than node $\mathcal{N}_j$. Therefore, bucket i will hit its capacity faster than j. In the final bandwidth distribution, we have

$$\frac{x_i + b_i}{C_i} \le \frac{x_j + b_j}{C_j}. \quad (5)$$

When (5) meets the strictly less than condition, we have $x_i = b_i$. In this case, node $\mathcal{N}_i$ is fully satisfied and reaches its maximal utility value of $U_i(x_i) = \log 2$. Therefore, $U_i(x_i) \ge U_j(x_j)$. When (5) meets the equality condition, we divide (4) by $(x_i + b_i)/C_i = (x_j + b_j)/C_j$, which gives

$$\frac{b_i}{x_i + b_i} \le \frac{b_j}{x_j + b_j} \Longrightarrow \frac{x_i + b_i}{b_i} \ge \frac{x_j + b_j}{b_j}$$
$$\Longrightarrow \log\left(\frac{x_i + b_i}{b_i}\right) \ge \log\left(\frac{x_j + b_j}{b_j}\right)$$
$$\Longrightarrow U_i(x_i) \ge U_j(x_j).$$ ■

Remarks: The implication of this theorem is that a competing node which has the highest contribution per unit of resource request will receive the highest utility. Therefore, the RBM-IU provides incentive in a P2P system and balances the utilities among all the competing nodes.

Theorem 3: The resource allocation $\vec{x}$ is Pareto optimal, which implies that the resource allocation vector cannot be improved further without reducing the utility of at least one competing node.

Proof: There are two cases for terminating the RBM-IU mechanism. One is when the aggregate bidding $\sum_{i=1}^{N} b_i \le \mathcal{W}_s$. In this case, $\vec{x} = \vec{b}$ and all nodes are equally satisfied. The second case is when the total resource $\mathcal{W}_s$ is fully utilized. In this case, no matter how efficient our resource allocation is, we have to decrease the resource of some competing node in order to improve the utility of another node. ■

IV. Resource Competition Game

In the proposed incentive P2P network, each competing node sends bidding messages to the source node. The source node then uses the mechanism RBM-IU for bandwidth resource distribution. The interactions between the competing nodes and the source node can be described in a game-theoretic framework [12]. We will explore the solution and some important properties of this game. Lastly, we discuss how the game can be realized in a P2P protocol such that it converges to Nash equilibrium.

A. Theoretical Competition Game

We model the resource bidding and distribution processes as a competition game among all the competing nodes. One basic postulate in game theory is that the game structure is *common knowledge* to all players. In our competition game, we assume that the total amount of bandwidth resource $\mathcal{W}_s$ and all the contribution values C_i's are common knowledge. This means that all nodes know the information, know that their rivals know the information, and know that their rivals know that they know the information, and so on. Also, we only consider the non-trivial situation of $\sum b_i > \mathcal{W}_s$. The competition game can be described as follows:

1. All the competing nodes are the players of the game.

2. The bidding message b_i is the strategy of the competing node $\mathcal{N}_i$. A bidding vector $\vec{b} = \{b_1, b_2, \ldots, b_N\}$ is a strategy profile where N is the number of competing nodes in the game.

3. The mechanism RBM-IU defines the rules and the structure of the game. We can regard mechanism RBM_IU as a mapping which has $\vec{C}$ and $\vec{b}$ as input parameters and returns $\vec{x}$ as output.

4. The outcome of the game is the vector $\vec{x}$ which represents the amount of bandwidth resource each competing node obtains.

5. The objective of each player is to maximize its assigned bandwidth x_i. We do not explicitly assume the utility function of each player. However, as long as the utility is a non-decreasing function in x_i, the objective to maximize x_i is equivalent to maximizing the player's underlying true utility.

Lemma 1: The mapping function RBM-IU: $\vec{C} \times \vec{b} \rightarrow \vec{x}$ is quasi-concave in each individual's strategy b_i.

Proof: Consider any strategy profile $\vec{b} = \{b_i, \vec{b}_{-i}\}$, where $\vec{b}_{-i}$ is any fixed strategy profile (or bidding messages) of players other than $\mathcal{N}_i$. We regard the resource allocation of $\mathcal{N}_i$ as a function of its bidding value, which is $x_i(b_i)$. When we increase b_i from zero gradually, $x_i(b_i)$ will also increase monotonically with $x_i(b_i) = b_i$. After that, when we continue to increase b_i, the marginal utility decreases, because the weighted marginal utility $C_i(x_i + b_i)^{-1}$ must be among the largest for b_i close to zero. At some point when the marginal utility is not among the largest, the bandwidth allocation satisfies $x_i(b_i) < b_i$ and will start to decrease monotonically in b_i until $x_i(b_i) = 0$. From the single peak property of $x_i(b_i)$, we know that the upper-level contour set is convex, and therefore the function is quasi-concave in b_i. ■

Theorem 4: There exists at least one Nash equilibrium in the competition game.

Proof: Note that the bidding values are finite because x_i becomes zero when b_i is larger than a certain threshold. Accordingly, the strategy set is convex and compact. The mechanism represents a continuous function of resource distribution, and from Lemma 1, it is quasi-concave in each b_i. Therefore, by [12, Prop. 8.D.3], the game has at least one Nash equilibrium. ■

Lemma 2: For any player, say $\mathcal{N}_i$, the strategy $b_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$ implies a resource allocation of $x_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$ for $i = 1, \ldots, N$.

Proof: Let W' be the least amount of resource which gives $\mathcal{N}_i$ the resource $x_i = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$. This outcome means that the "resource level" of $\mathcal{N}_i$ is at a height of $h' = 2b_i^*/C_i = 2\mathcal{W}_s / \sum_{j=1}^N C_j$. Any other player, say $\mathcal{N}_k$, may report its strategy b'_k in two possible cases: 1) When $b'_k \leq \mathcal{W}_s C_k / \sum_{j=1}^N C_j$, we have $2b'_k/C_k \leq 2\mathcal{W}_s / \sum_{j=1}^N C_j = h'$. Hence, $x'_k = b'_k \leq \mathcal{W}_s C_k / \sum_{j=1}^N C_j$. 2) When $b'_k > \mathcal{W}_s C_k / \sum_{j=1}^N C_j$, we have $b'_k/C_k > \mathcal{W}_s / \sum_{j=1}^N C_j$. Hence, $x'_k = (h' - b'_k/C_k) C_k < \mathcal{W}_s C_k / \sum_{j=1}^N C_j$. As a result, $W' = \sum_{k=1}^N x'_k \leq \sum_{k=1}^N \mathcal{W}_s C_k / \sum_{j=1}^N C_j = \mathcal{W}_s$. So with $\mathcal{W}_s$ amount of resource, we can at least distribute $\mathcal{W}_s * C_i / \sum_{j=1}^N C_j$ amount of resource to player $\mathcal{N}_i$, which is also the bidding value b_i. Therefore, the RBM-IU mechanism will allocate exactly $x_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$ amount of resource to player $\mathcal{N}_i$. ■

Remark: The importance of the above lemma is in guaranteeing that a player can gain its fair share of resources during the competition. Players who have small contribution values will not suffer from resource starvation. Free riders, however, will eventually gain zero resource in the competition.

Theorem 5: The strategy profile $b_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$ for player $\mathcal{N}_i$, where $i = 1, \ldots, N$, is a Nash equilibrium..

Proof: The aggregate bidding is $\sum_{i=1}^N b_i^* = \mathcal{W}_s$, so that $x_i^* = b_i^*$ for player $\mathcal{N}_i$, for $i = 1, \ldots, N$. From Lemma 2, any player $\mathcal{N}_k$ who insists on $b_k^* = \mathcal{W}_s C_k / \sum_{j=1}^N C_j$ gains $x_k = b_k^* = \mathcal{W}_s C_k / \sum_{j=1}^N C_j$. Therefore, regardless of the change of strategy from b_i^*, player $\mathcal{N}_i$ gains $x_i \leq \mathcal{W}_s - \sum_{k \neq i} x_k^* = x_i^*$. ■

Theorem 6: The strategy profile $b_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$ for player $\mathcal{N}_i$, where $i = 1, \ldots, N$, is a unique Nash equilibrium.

Proof: Suppose there exists another Nash equilibrium $\{\hat{b}_i, \hat{x}_i\}$, where each player i uses strategy $\hat{b}_i$ and gains $\hat{x}_i$ amount of resource. At least one of the players has $\hat{b}_i \neq b_i^*$. By Lemma 2, independent of the strategies used by the other players, strategy b_i^* induces $x_i^* = b_i^*$. Because the RBM-IU mechanism will not assign x_i larger than the bidding b_i, the first necessary condition for $\{\hat{b}_i, \hat{x}_i\}$ to be a Nash Equilibrium is $\hat{b}_i \geq b_i^* \; \forall \, i$. Otherwise, b_i^* can always get more bandwidth than any $\hat{b}_i < b_i^*$. For the same reason, if $\hat{x}_i < x_i^*$, strategy b_i^* performs better than $\hat{b}_i$ and strategy $\hat{b}_i$ cannot be a Nash strategy. Therefore, the second necessary condition for $\{\hat{b}_i, \hat{x}_i\}$ to be a Nash Equilibrium is $\hat{x}_i = x_i^*$ for each player i.

The condition $\hat{x}_i = x_i^* \; \forall \, i$ implies that the "resource height" is $\hat{x}_i / C_i = x_i^* / C_i = W_s / \sum_{j=1}^N C_j$, which is constant for all the players. By the water-filling algorithm, we know that the initial resource level $\hat{b}_i / C_i$ must also be the same for each player i. Therefore, $\{\hat{b}_i\}$ should only be in the form of $\hat{b}_i = (1 + \delta) b_i^*$ for all i, where $\delta > 0$ is some constant. But this cannot be a Nash equilibrium, because any player can unilaterally change $\hat{b}_i = (1 + \delta) b_i^*$ to be smaller in order to gain $\hat{x}_i > x_i^*$. For example, if any player i unilaterally changes its strategy $\hat{b}_i = (1 + \delta) b_i^*$ to be $\hat{b}_i = (1 + \delta/2) b_i^*$, its bandwidth is increased by $\frac{1}{2} \delta h C_i \left(1 - C_i / \sum_{j=1}^N C_j\right)$, where $h = \mathcal{W}_s / \sum_{j=1}^N C_j$. ■

Another important property of our protocol is that it can avoid one form of collusion attack.

Definition 2: κ-collusion occurs when a subset of competing nodes $\mathcal{N}_\kappa$ use strategy profile $b_i \neq b_i^* \; \forall \, i \in \mathcal{N}_\kappa$, and achieve $\sum_{i \in \mathcal{N}_\kappa} x_i > \sum_{i \in \mathcal{N}_\kappa} x_i^*$.

Theorem 7: Assuming that all honest competing nodes use the Nash equilibrium strategy $b_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$, the RBM-IU mechanism at the source node avoids κ-collusion.

Proof: Suppose some but not all players are dishonest. When the honest players play their Nash equilibrium strategy $b_i^* = \mathcal{W}_s C_i / \sum_{j=1}^N C_j$, by Lemma 2, the honest players are guaranteed to have $x_i^* = b_i^*$. Therefore, the aggregate resource received by the dishonest players are $\mathcal{W}_s - \sum_{\text{honest}} x_i^*$, which cannot exceed what they could have gained in the Nash equilibrium. ■

B. Practical Competition Game Protocol

In Section IV-A, we showed that the interactions between the source node and all its competing nodes can be modeled as a competition game, which has a Nash equilibrium solution. This solution assigns each competing node an amount of resource *proportional* to the node's contribution, efficiently utilizes all resource at the source node, and also prevents collusion among a group of competing nodes.

Although the theoretical competition game provides these attractive properties, there are gaps to fill so as to realize the theoretical competition game in an actual incentive P2P network. In particular, one needs to address the following issues:

- **I1** The information of contribution $\vec{C}$ and the amount of resource $\mathcal{W}_s$ is assumed to be common knowledge. How can this be implemented in a P2P system?
- **I2** In real life, a competing node, say $\mathcal{N}_i$, has its maximum download capacity, say w_i (in units of bits/s). Also, due to intermittent network congestions, the actual assigned bandwidth allocation x_i may be less than the actual received bandwidth x'_i. These two factors will change the Nash equilibrium derived under the theoretical competition game.
- **I3** In a dynamic environment like a P2P network, new competing nodes may arrive and request file download, while existing competing node may leave due to the termination of their file transfers. Under these situations, how can the system reach equilibrium when the number of competing nodes changes? (More challenges are addressed in [5].)

To address these issues, let us first consider the behavior of the source node. Based on a given strategy profile $\vec{b}$ and contribution values $\vec{C}$, the source node carries out the RBM-IU for bandwidth resource distribution. The *justification* that the source node is willing to use this mechanism is that the allocation result is *Pareto optimal* (based on Theorem 3). (Although other Pareto optimal solutions may exist, the source node has no incentive to switch because the other solutions are not more efficient.) This implies that following the RBM-IU mechanism, the source node can maximize its contribution value so that it can enjoy better service for its future file download requests. However, without perfect information for all the competing nodes, the game solution may oscillate and induce resource wastage. In order for the source node to maximize its contribution, it has *incentive* to help the competing nodes reach Nash equilibrium. In our practical game protocol, the source node will signal a competing node, say $\mathcal{N}_i$, with the value of $S_i = \mathcal{W}_s C_i / \sum_{j=1}^{N} C_j$, when $\mathcal{N}_i$ initiates its request for file download. This information exchange is inexpensive because: 1) the signal is sent only *once* for each competing node's arrival, and 2) the signal value is computed on the fly and it does not need global information of all the network nodes' contribution values. Hence, the issue **I1** is resolved.

For the behavior of the competing nodes, let us see how the signals sent by the source node may help the game reach its equilibrium. Suppose that a competing node, say $\mathcal{N}_i$, has a maximum download capacity of w_i and a *signal variable* s_i. Initially, s_i encodes the signal value sent by the source node; i.e., $s_i = S_i = \mathcal{W}_s C_i / \sum_{j=1}^{N} C_j$. The competing node $\mathcal{N}_i$ sends its initial bidding message $b_i = \min\{w_i, s_i\}$ to the source node. After each round of data transfer, $\mathcal{N}_i$ measures x'_i, the amount of bandwidth resource it receives from the source node, and stores it as the current signal value s_i; i.e., $s_i = x'_i$, To start the next round of data transfer, $\mathcal{N}_i$ sends a new bidding message $b_i = \min\{w_i, s_i\}$ to the source node. This bidding strategy assumes that the source node uses the RBM-IU mechanism, so that all the competing nodes reach Nash equilibrium through feedback on their strategies. In the bidding message, a competing node informs the source node of 1) its download bandwidth limit, and 2) whether there is any congestion along the data transfer path.

The behavior of competing nodes described above is an attempt to resolve the issues of **I2** and **I3**. However, one can show that using this protocol, the system may *not* be able to reach Nash equilibrium. Consider the following illustrative example. Initially, the source node $\mathcal{N}_s$ has resource $\mathcal{W}_s = 6$. There is one competing node $\mathcal{N}_1$ with $w_1 = 10$ and $C_1 = 1$. The source node sends $\mathcal{N}_1$ a signal of $S_1 = 6$. Therefore, the initial bidding message from $\mathcal{N}_1$ is $b_1 = \min\{10, 6\} = 6$ and the resource allocation is $x_1 = 6$ (which is a Nash equilibrium point). Afterwards, a new competing node $\mathcal{N}_2$ arrives with $w_2 = 1$ and $C_2 = 1$. The source node sends $\mathcal{N}_2$ a signal of $S_2 = 3$. Therefore, the initial bidding message from $\mathcal{N}_2$ is $b_2 = \min\{1, 3\} = 1$. The final resource allocation is $\vec{x} = [5, 1]$ (which is also a Nash equilibrium point). Now a new competing node $\mathcal{N}_3$ arrives with $w_3 = 10$ and $C_3 = 1$. The source node sends $\mathcal{N}_3$ a signal of $S_3 = 2$. Therefore, the initial bidding message from $\mathcal{N}_3$ is $b_3 = \min\{10, 2\} = 2$. The final resource allocation is $\vec{x} = [3, 1, 2]$. Note that this equilibrium point is *not* a Nash equilibrium since there is some degree of unfairness between the two homogeneous nodes $\mathcal{N}_1$ and $\mathcal{N}_3$, and $\mathcal{N}_3$ could have received a higher bandwidth if had increased its bidding. Another scenario in which the final resource allocation is not a Nash equilibrium is when some of the competing nodes experience network congestion such that $x'_i < x_i$. When these nodes feed back their new biddings $b_i = x'_i$ for the resource allocation, some resource at the source node will not be utilized and may remain idle. This condition persists even if these competing nodes are relieved from network congestion at a later time. In other words, they cannot gain back the amount of resource they could have obtained in the Nash equilibrium. In summary, each competing node needs to behave more *aggressively* in order to get the proper amount of resource and also help the system reach a new Nash equilibrium efficiently.

To properly resolve issues **I2** and **I3**, we propose the following extension protocol. Each competing node, say $\mathcal{N}_i$, enhances its bidding by sending

$$b_i = \min\{w_i, (1+\delta)s_i\} \tag{6}$$

where δ is a small positive constant for all the competing nodes. The purpose of reporting a slightly larger bidding value is to explore the possibility of whether there is some idle resource at the source node. The Nash equilibrium result $\vec{x}^*$ in the theoretical model does not change except that the strategy profile is changed to be $\vec{b}^* = (1+\delta)\vec{x}^*$. In case there are idle resources or temporarily unfair resource allocations in the system, competing nodes which gain a smaller amount resource can increase their biddings and push the system to the new Nash equilibrium point. Hence, their subsequent bidding values will increase. Eventually, a new equilibrium is reached when each competing node bids $b_i = \min\{w_i, (1+\delta)s_i\}$ and receives $x'_i = s_i$.

From now on, we will assume that all the competing nodes in the incentive P2P network send bidding messages according to (6). Obviously, all competing nodes interacting with the source node will achieve a different allocation result in equilibrium as compared with the Nash equilibrium in the theoretical context. We classify these competing nodes into three categories at equilibrium. When the bidding is $b_i = w_i$, the competing node receives $x_i' = w_i$, and the allocated resource must be $x_i = w_i$. This implies that the competing node does not encounter any network congestion. When the bidding is $b_i = (1+\delta)x_i'$, there are two cases to consider: 1) There is a bottleneck (with available bandwidth v_i) along the path between the competing node and the source node. Therefore, no matter how large the contribution value of the competing node or its bidding value, the competing node can only receive v_i amount of bandwidth resource. So we have $b_i = (1+\delta)x_{i'} = (1+\delta)v_i$. 2) The competing node competes with other competing nodes for the resource at the source node. Therefore, the bottleneck is on the source node side. So we know $b_i = (1+\delta)x_{i'} = (1+\delta)x_i$. Defining the three categories of competing node at equilibrium to be the sets $\mathcal{N}_\alpha$, $\mathcal{N}_\beta$, and $\mathcal{N}_\gamma$, respectively, we have the following results:

Lemma 3: For any equilibrium of the dynamic game

$$x_i/C_i = x_j/C_j$$

for all $\mathcal{N}_i, \mathcal{N}_j \in \mathcal{N}_\gamma$. ■

Proof: For competing nodes $\mathcal{N}_i \in \mathcal{N}_\gamma$, the bottleneck is on the source node side. Following the equilibrium condition $b_i = (1+\delta)x_i$ and $x_i' = x_i$ for each competing node in $\mathcal{N}_\gamma$, the final "resource allocation level" in the RBM-IU mechanism should be $(x_i + b_i)/C_i = (2+\delta)x_i/C_i$ for all the competing nodes in $\mathcal{N}_i \in \mathcal{N}_\gamma$.

Lemma 4: For any equilibrium of the dynamic game

$$x_i/C_i + \frac{1}{2}\delta x_i/C_i \geq x_j/C_j$$

for all $\mathcal{N}_i \in \mathcal{N}_\gamma$ and $\mathcal{N}_j \in \mathcal{N}_\alpha \cup \mathcal{N}_\beta$.

Proof: Suppose we have a competing node $\mathcal{N}_i \in \mathcal{N}_\gamma$ and $\mathcal{N}_j \in \mathcal{N}_\alpha \cup \mathcal{N}_\beta$. For some competing node $\mathcal{N}_j$, the bottleneck is at the client side or at an intermediate link. The final resource allocation level in the RBM-IU mechanism, which is $(x_i + b_i)/C_i$, must be higher than or equal to the resource allocation level of any $\mathcal{N}_j$, which is $(x_j + b_j)/C_j$. Therefore, we have

$$(x_i + b_i)/C_i \geq (x_j + b_j)/C_j.$$

From the equilibrium condition 1) $b_j = x_j = x_j' = w_j$ for $\mathcal{N}_j \in \mathcal{N}_\alpha$; 2) $b_j = x_j = (1+\delta)x_j' = v_j$ for $\mathcal{N}_j \in \mathcal{N}_\beta$; and 3) $b_i = (1+\delta)x_i = (1+\delta)x_i'$ for $\mathcal{N}_i \in \mathcal{N}_\gamma$. Therefore, we have $(2+\delta)x_i/C_i \geq 2x_j/C_j$. ■

Theorem 8: The dynamic game equilibrium described above has the bandwidth allocation solution

$$x_i = \begin{cases} w_i, & \text{if } x_i \in \mathcal{N}_\alpha \\ v_i, & \text{if } x_i \in \mathcal{N}_\beta \\ \dfrac{C_i}{\sum\limits_{j\in\mathcal{N}_\gamma} C_j}\left(\mathcal{W}_s - \sum\limits_{j\in\mathcal{N}_\alpha} w_j - \sum\limits_{j\in\mathcal{N}_\beta} v_j\right), & \text{if } x_i \in \mathcal{N}_\gamma. \end{cases} \tag{7}$$

In addition, it becomes a Nash equilibrium solution when δ approaches zero.

Proof: $x_i = w_i$ if $x_i \in \mathcal{N}_\alpha$ and $x_i = v_i$ if $x_i \in \mathcal{N}_\beta$ follow directly from the equilibrium condition. Since all the competing nodes in $\mathcal{N}_\gamma$ are not saturated, they use up all the remaining resource $\mathcal{W}_s - \sum_{j\in\mathcal{N}_\alpha} w_j - \sum_{j\in\mathcal{N}_\beta} v_j$. Follow Lemma 3, the last equation holds.

When δ approaches zero, the strategy profile in equilibrium approaches:

$$b_i = \begin{cases} w_i, & \text{if } x_i \in \mathcal{N}_\alpha \\ v_i, & \text{if } x_i \in \mathcal{N}_\beta \\ \dfrac{C_i}{\sum\limits_{j\in\mathcal{N}_\gamma} C_j}\left(\mathcal{W}_s - \sum\limits_{j\in\mathcal{N}_\alpha} w_j - \sum\limits_{j\in\mathcal{N}_\beta} v_j\right), & \text{if } x_i \in \mathcal{N}_\gamma. \end{cases} \tag{8}$$

By Lemma 4, $x_i/C_i \geq x_j/C_j$ for all $\mathcal{N}_i \in \mathcal{N}_\gamma$ and $\mathcal{N}_j \in \mathcal{N}_\alpha \cup \mathcal{N}_\beta$. Physically, it implies that all $\mathcal{N}_i \in \mathcal{N}_\gamma$ have the final resource "water level" higher than or equal to that of nodes in $\mathcal{N}_\alpha \cup \mathcal{N}_\beta$. This strategy profile's solution is a Nash equilibrium: 1) For the competing nodes $\mathcal{N}_j \in \mathcal{N}_\alpha \cup \mathcal{N}_\beta$, they gain the maximum resource so that there exists no other better strategy for them to deviate. 2) For the competing nodes $\mathcal{N}_i \in \mathcal{N}_\gamma$, they will not bid $b_i' < b_i$ since x_i always less than or equal to b_i in RBM-IU. If they bid $b_i' > b_i$, consider the sub-game when $\mathcal{W}_s' = \mathcal{W}_s - \sum_{j\in\mathcal{N}_\alpha} w_j - \sum_{j\in\mathcal{N}_\beta} v_j$ and the set of competing nodes is $\mathcal{N}_\gamma$. From Theorem 5, we know that the strategy b_i is a Nash equilibrium which means that a competing node cannot become better off by deviating from the strategy. ■

Remark: Although the equilibria in the dynamic game are not strictly Nash, they are close to Nash when δ is small. The allocation results from these equilibria are the same as the equilibrium allocations when $\delta = 0$. Therefore, we can regard the game as reaching Nash equilibrium if all the players play the Nash strategy profile.

V. Generalized Mechanism and Game

In the last two sections, we discussed a specific RBM-IU mechanism and its corresponding resource competition game. In this section, we generalize the resource distribution mechanism with respect to incentive and utility. For incentive, we give a parametric manipulation of the contribution values C_i used, so that we can control the degree of incentive provided in the mechanism. For utility, we explore heterogeneous nodes which have diverse utility functions (i.e., not necessarily given by $U_i(x_i, b_i) = \log(x_i/b_i + 1)$ as assumed in Section III). We will analyze the properties of the competition games corresponding to the generalized mechanisms.

A. Generalized Mechanism With Incentive

Recall the mechanism RBM-I in (2). We introduce incentive by distributing the resource in linear proportion to each competing node's contribution value C_i. In general, contribution values can be weighted by an exponent $r \geq 0$, and the resource distribution becomes

$$x_{\hat{k}} = \min\left\{ b_{\hat{k}}, \frac{C_{\hat{k}}^r \left(\mathcal{W}_s - \sum_{\hat{i}=1}^{\hat{k}-1} x_{\hat{i}} \right)}{\sum_{j=\hat{k}}^{N} C_j^r} \right\} \quad \hat{k} = 1, \ldots, N. \quad (9)$$

It is easy to observe that when $r = 0$, this mechanism is equivalent to the mechanism RBM-I. On the other hand, when r tends to infinity, this mechanism becomes a strict priority service mechanism which serves the requests by their contribution values in descending order. Generally speaking, the larger the value of r, the higher the amount of the allocated resource this mechanism provides based on contribution values. Therefore, the parameter r provides some degree of freedom for the mechanism designer to balance between incentive and fairness in the P2P system.

Similarly, by generalizing C_i to C_i^r, RBM-IU becomes

$$\max \sum_{i=1}^{N} C_i^r \log\left(\frac{x_i}{b_i} + 1\right) \text{ s.t. } \sum_{i=1}^{N} x_i \leq \mathcal{W}_s,\ x_i \in [0, b_i]\ \forall\ i.$$

Because the new mechanism linearly weights each contribution by C_i^r, the implementation of the new mechanism can be easily extended by changing the original filling algorithm. In extending both the RBM-I and the RBM-IU, we make the filling rate of each competing node to be C_i^r instead of C_i.

Lastly, this extension maintains the properties of all the previous theorems and the corresponding resource competition game. Two generalized versions of previous theorems are as follows.

Theorem 9: For any two competing nodes $\mathcal{N}_i, \mathcal{N}_j$, the generalized RBM-IU assigns the bandwidth x_i and x_j such that

$$\text{if } \frac{C_i^r}{b_i} \geq \frac{C_j^r}{b_j} \Longrightarrow U_i(x_i) \geq U_j(x_j). \quad (10)$$

Proof: The proof is similar to that of Theorem 2. ∎

Theorem 10: The strategy profile $b_i^* = \mathcal{W}_s C_i^r / \sum_{j=1}^{N} C_j^r$ for player $\mathcal{N}_i$, where $i = 1, \ldots, N$, is the unique Nash equilibrium.

Proof: The proof is similar to that of Theorem 6. ∎

B. Generalized Mechanism With Utility

In RBM-IU, we assume a special form of the perceived utility function: $U_i(x_i) = \log(x_i/b_i + 1)$. Although this form can be reasonable in practice, we would like to consider the more general situation in which the competing nodes have heterogeneous utility functions. We will first design a mechanism for the general situation, and then discuss the properties of the corresponding competition game (e.g., existence and characterization of any equilibrium solution).

Similar to RBM-U, our new mechanism tries to maximize the social utility. As in the context of resource bidding, the (perceived) utility function of node i should depend on the bidding b_i. We design the new mechanism to solve the following distributed optimization problem:

$$\max \sum_{i=1}^{N} U_i(x_i, b_i) \text{ s.t. } \sum_{i=1}^{N} x_i \leq \mathcal{W}_s \text{ and } x_i \in [0, b_i]\ \forall\ i.$$

The utility function of a node, say i, is denoted by $U_i(\cdot)$ and depends on x_i and b_i. Let us consider $U_i = x_i/b_i$. We can regard x_i/b_i (with range $[0,1]$) as node i's "fraction of satisfaction."

Theorem 11: There exists at least one Nash equilibrium in the competition game induced by the following mechanism:

$$\max \sum_{i=1}^{N} U_i\left(\frac{x_i}{b_i}\right) \text{ s.t. } \sum_{i=1}^{N} x_i \leq \mathcal{W}_s \text{ and } x_i \in [0, b_i]\ \forall\ i$$

where $U_i(\cdot)$ is any concave function for all i.

Proof: Consider any strategy profile $\vec{b} = \{b_i, \vec{b}_{-i}\}$, where $\vec{b}_{-i}$ is any fixed strategy profile (of bidding messages) of all the players other than $\mathcal{N}_i$. The resource allocation $\mathcal{N}_i$ is a function of i's bidding value, denoted by $x_i(b_i)$. By the constraint $x_i \leq b_i$, when $x_i(\hat{b_i}) = \hat{b_i}$ for some $\hat{b_i}$, x_i may increase for node i only if i bids $b_i > \hat{b_i}$. Again, the mechanism solves the optimization problem by giving a resource increment to the competing node which currently has the largest marginal utility. Also, the marginal utility $dU_i/dx_i = U_i'(x_i/b_i)/b_i$ is decreasing when we increase b_i. Hence, we need to consider three cases of the function $x_i(b_i)$: 1) The marginal utility of player i when $x_i = 0$ for any $b_i > 0$ (i.e., $U_i'(0)/b_i$) is less than that of any other player when the resource is used up. We have $x_i = 0\ \forall\ b_i > 0$. 2) The marginal utility when $x_i = W_s$ for any $b_i > 0$ (i.e., $U_i'(W_s/b_i)/b_i$) is always the largest among all the players. We have $x_i = \min\{b_i, W_s\}\ \forall\ b_i > 0$. 3) We increase b_i gradually from zero to infinity. x_i increases from zero when its marginal utility is among the largest and $x_i(b_i) = b_i$. After that, when we continue to increase b_i, the marginal utility decreases and x_i decreases until $x_i = 0$. When we have $x_i(\hat{b_i}) = 0$ for some $\hat{b_i} > 0$, we have $x_i(b_i) = 0\ \forall\ b_i \geq \hat{b_i}$. From the single peak property of $x_i(b_i)$, we know that the upper-level contour set is convex. Therefore, the function is quasi-concave in b_i. By similar arguments used in Theorem 4, we know that there exists at least one Nash Equilibrium in the competition game. ∎

Theorem 12: Suppose $\vec{b^*}$ and $\vec{x^*}$ is a Nash equilibrium in Theorem 11. For any $x_i^*, x_j^* > 0$, we have

$$U_i'\left(\frac{x_i^*}{b_i^*}\right) : U_j'\left(\frac{x_j^*}{b_j^*}\right) = b_i^* : b_j^*.$$

Proof: Assume $U_i'\left(x_i^*/b_i^*\right) : U_j'\left(x_j^*/b_j^*\right) > b_i^* : b_j^*$ for some $x_i^*, x_j^* > 0$. Therefore, $U_i'\left(x_i^*/b_i^*\right)/b_i^* > U_j'\left(x_j^*/b_j^*\right)/b_j^*$, which implies that the marginal utility of player i is higher than that of player j. Therefore, the resource distribution mechanism can increase the aggregate utility by shifting some resource from player j to player i. So the mechanism does not solve the maximization of the aggregate utility. We have a contradiction. ∎

Remarks: The implication of this theorem is that the bidding of each player in equilibrium should be proportional to their marginal utility (or shadow price) at that equilibrium point.

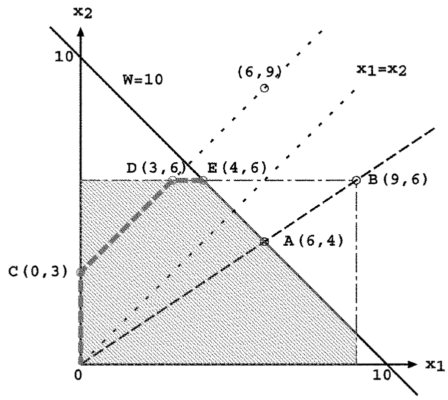


Fig. 4. Convergence illustration where $N = 2$, $\mathcal{W}_s = 10$, and $\delta = 0.5$.

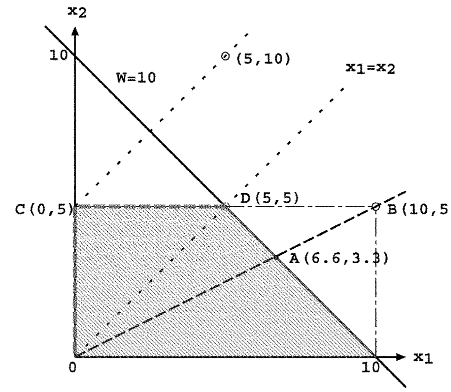


Fig. 5. Convergence illustration where $N = 2$, $\mathcal{W}_s = 10$ and $\delta = 0.5$.

VI. Convergence Analysis

In this section, we investigate the convergence of the practical competition game described in Section IV. We show that by using a small positive value of δ, the solutions of the practical game converges to a neighborhood of the Nash equilibrium in the theoretical competition game. Without loss of generality, we assume that all competing nodes have the same contribution values. We will also focus on the non-trivial case when the aggregate demand is larger than the resource bandwidth, i.e., $\sum_{i=1}^{N} b_i > \mathcal{W}_s$.

We start from a simple example shown in Fig. 4. There are two competing nodes and a source node with bandwidth resource $\mathcal{W}_s = 10$ Mb/s. Suppose at some moment the resource allocation is at point A (6,4). Each competing node sends a bidding value $b_i = (1+\delta)x_i$, where $\delta = 0.5$. We have the new bidding at point B (9,6). The shaded area in the figure is the feasible region for the new allocation, which physically implies that: 1) each competing node's allocation will be non-negative and no larger than its bidding value, and 2) the aggregate allocation will not exceed the total bandwidth resource $\mathcal{W}_s$.

The mechanism progressively moves the resource allocation from the origin to point C (0,3), giving the bandwidth resource to the competing nodes offering smaller bidding values. After that, the mechanism shares the bandwidth resource evenly until the second competing node reaches its bidding value at point D (3,6). The mechanism then completes the allocation process at point E (4,6). Therefore, the allocation result oscillates between points A and E near the equilibrium point (5,5). We can imagine that the smaller the value of δ, the shorter the corresponding convergence diameter.

Also, notice that by choosing an initial condition according to the value δ, we can achieve the Nash equilibrium solution. In Fig. 5, the initial allocation is at point A (6,6,3.3) and the following bidding is at point B (10,5). This bidding pair leads the two players to reach the Nash equilibrium solution D (5,5).

Theorem 13: For two players with bidding $b_i = (1+\delta)x_i$, the allocation solution converges to the neighborhood of equilibrium point $(\mathcal{W}_s/2, \mathcal{W}_s/2)$, where $|x_i - (\mathcal{W}_s/2)| \leq (\delta/2)(\mathcal{W}_s/2)$ for $i = 1, 2$.

Proof: We define

$$\begin{aligned} e(t) &= |x_1(t) - (\mathcal{W}_s/2)| \\ &= |x_2(t) - (\mathcal{W}_s/2)| \;\text{ for } t = 0,1,2\ldots \end{aligned}$$

$$\begin{aligned} \text{and } \limsup e &\equiv \lim_{t\to\infty} \sup e(t) \\ &\equiv \inf_{t\geq 0}\{\sup\{e(k)|k \geq t\}\}. \end{aligned}$$

Without loss of generality, we assume $x_1 > x_2$ at some time t. Hence, we have $(x_1(t), x_2(t)) = ((\mathcal{W}_s/2) + e(t), (\mathcal{W}_s/2) - e(t))$ and $(b_1(t), b_2(t)) = ((1+\delta)x_1(t)), (1+\delta)x_2(t))$. Consequently, we gain the resource allocation at time $t+1$ as follows:

1. If $e(t) \leq (\delta/2(1+\delta))(\mathcal{W}_s/2)$, then $x_1(t+1) = (\mathcal{W}_s/2) - (1+\delta)e(t)$ and $x_2(t+1) = (\mathcal{W}_s/2) + (1+\delta)e(t)$.
2. If $e(t) > (\delta/2(1+\delta))(\mathcal{W}_s/2)$, then $x_1(t+1) = (1-\delta)(\mathcal{W}_s/2) + (1+\delta)e(t)$ and $x_2(t+1) = (1+\delta)(\mathcal{W}_s/2) - (1+\delta)e(t)$.

In the first case, $e(t+1) = (1+\delta)e(t) > e(t)$. In the second case, when $e(t) \in ((\delta/2(1+\delta))(\mathcal{W}_s/2), (\delta/2+\delta)(\mathcal{W}_s/2))$, we have $e(t+1) > e(t)$; otherwise, $e(t+1) \leq e(t)$. Therefore, $e(t+1) > e(t)$ only if $e(t) < (\delta/2+\delta)(\mathcal{W}_s/2)$ for all t. It is easy to show that we also have

$$\sup\left\{e(t+1)|e(t) < \frac{\delta}{2+\delta}\frac{\mathcal{W}_s}{2}\right\} = \frac{\delta}{2}\frac{\mathcal{W}_s}{2}. \qquad \blacksquare$$

One can easily extend this theorem to cover more than two competing nodes.

VII. Numerical Examples

In this section, we present numerical results to illustrate the performance and the incentive property of our resource distribution protocol. In particular, we show that our protocol can properly adapt to dynamic join/leave of competing nodes, and to various conditions of network congestion.

Example A (Incentive Resource Distribution): We consider a source node $\mathcal{N}_s$ with resource $\mathcal{W}_s = 2$ Mb/s. There are four competing nodes $\mathcal{N}_1$ to $\mathcal{N}_4$. Their maximum download bandwidths are $\vec{w} = [2, 1.5, 1, 0.5]$ (in Mb/s). The arrival times of $\mathcal{N}_1$, $\mathcal{N}_2$, $\mathcal{N}_3$, and $\mathcal{N}_4$ are $t = 20, 40, 60$, and 80 s, respectively. Unless stated otherwise, the propagation delay between a competing node and the source node $\mathcal{N}_s$ is one second and all the competing nodes use $\delta = 0.1$ in (6). We consider three scenarios, each using different contribution values for the four competing nodes. In **A.1**, we have $\vec{C} =$ [100,100,100,100]; in **A.2**, we have $\vec{C} =$ [400,300,200,100]; in **A.3**, we have $\vec{C} =$ [400,100,200,300]. Fig. 6 illustrates the *instantaneous* bandwidth allocation for all the competing nodes for $t \in$ [0,100]. One can make the following observations:

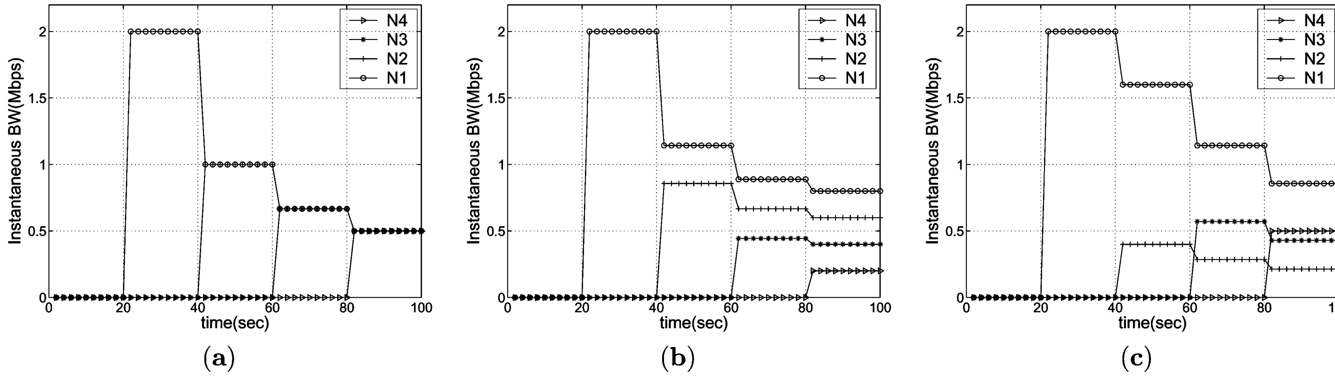


Fig. 6. Instantaneous bandwidth allocations: (a) $\vec{C} = [100,100,100,100]$; (b) $\vec{C} = [400,300,200,100]$; (c) $\vec{C} = [400,100,200,300]$.

• For $t \in [40, 60]$, the resource is evenly shared by $\mathcal{N}_1$ and $\mathcal{N}_2$ since they have the same contribution values. When all the four competing nodes are present ($t \in$ [80,100])), each node will get a resource amount $x = 0.5$ Mb/s.

• Fig. 6(b) shows that the bandwidth assignment is *proportional* to the contribution value of a competing node. When all four competing nodes are present ($t \in$ [80,100]), the resource allocation vector is $\vec{x} = [0.8, 0.6, 0.4, 0.2]$ (Mb/s). Hence, RBM-IU provides service differentiation, such that nodes have incentive to share information and to provide services.

• Fig. 6(c) shows that the protocol will not waste any resource at the source node. Given $\vec{C} = [400,100,200,300]$, the resource distribution should be $\vec{x} = [0.8, 0.2, 0.4, 0.6]$ (Mb/s). But since the maximum download bandwidth of $\mathcal{N}_4$ is $w_4 = 0.5$ Mb/s only, the remaining resource (0.1 Mb/s) will be distributed *proportionally* to $\mathcal{N}_1, \mathcal{N}_2$ and $\mathcal{N}_3$. The final resource distribution is $\vec{x} = [0.86, 0.21, 0.43, 0.5]$ (Mb/s).

In summary, these examples show that the RBM-IU can provide incentive service differentiation and will efficiently utilize resources at the source node.

Example B (Adaptivity to Dynamic Join/Leave of Competing Nodes): We consider a source node $\mathcal{N}_s$ with resource $\mathcal{W}_s = 2$ Mb/s. There are four competing nodes $\mathcal{N}_1$ to $\mathcal{N}_4$ with contributions $\vec{C} = [400,300,200,100]$ and maximum download bandwidths $\vec{w} = [2, 1.5, 1, 0.5]$ (in Mb/s). There is a propagation delay of one second between a competing node and the source node.

We consider two scenarios of arrival and departure patterns: **B.1:** $\mathcal{N}_1$ arrives and departs at $t = 40$ and $t = 160$, $\mathcal{N}_2$ arrives and departs at $t = 60$ and $t = 100$, $\mathcal{N}_3$ arrives and departs at $t = 80$ and $t = 120$, and $\mathcal{N}_4$ arrives and departs at $t = 20$ and $t = 140$. **B.2:** $\mathcal{N}_1$ arrives and departs at $t = 20$ and $t = 100$, $\mathcal{N}_2$ arrives and departs at $t = 80$ and $t = 120$, $\mathcal{N}_3$ arrives and departs at $t = 60$ and $t = 140$, and $\mathcal{N}_4$ arrives and departs at $t = 40$ and $t = 160$. Fig. 7 illustrates the instantaneous bandwidth allocation for time $t \in$ [0,180]. One can make the following observations:

• The protocol can assign the proper amount of resource to competing nodes without wastage. For example, for time $t \in [20, 40]$, Fig. 7(a) shows that $\mathcal{N}_4$ obtains 0.5 Mb/s (since this is its maximum download bandwidth). But for the same time period, Fig. 7(b) shows that $\mathcal{N}_1$ can get 2.0 Mb/s, its maximum download bandwidth and the full resource of the source node.

• Both Fig. 7(a) and (b) show that the protocol can fully utilize the source resources. For example, for period $t \in$ [40,120], the source node distributes the resource proportionally to the contribution values of the competing nodes. The assignment is independent of the number of competing nodes and their arrival patterns.

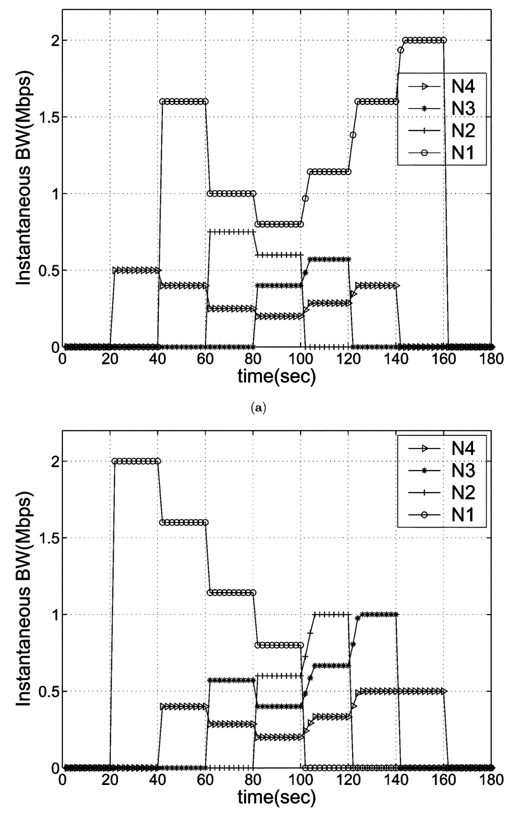


Fig. 7. Instantaneous bandwidth allocations for arrival and departure patterns. (a) B.1; (b) B.2.

• The protocol can reach the same equilibrium point, *independent* of the arrival and departure sequences of **B.1** or **B.2**. For example, consider the time period $t \in [80,100]$. The resource distribution for both cases is $\vec{x} = [0.8, 0.6, 0.4, 0.2]$ (in Mb/s), which is also the Nash equilibrium point.

In summary, these examples show that the protocol is adaptive to the arrival and departure sequence, and it provides service differentiation to different competing nodes having different contribution values.

Example C (Adaptivity to Network Congestion): We consider one source node $\mathcal{N}_s$ with resource $\mathcal{W}_s = 2$ Mb/s. At time

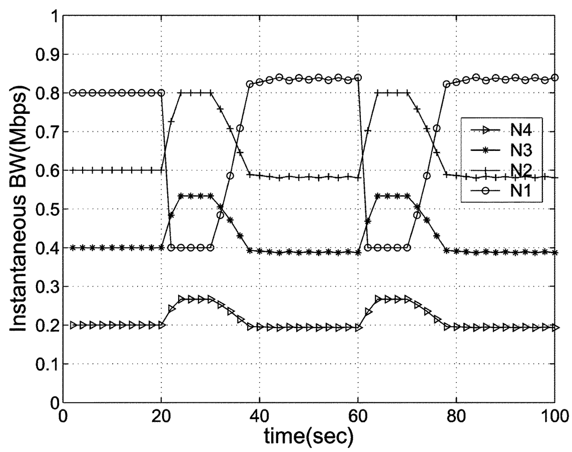


Fig. 8. Instantaneous bandwidth allocations for four competing nodes; congestion occurs at $t = [30, 40]$ and $t = [50, 60]$.

$t = 0$, there are four competing nodes $\mathcal{N}_1$ to $\mathcal{N}_4$ in the system. These nodes have contribution values $\vec{C} = [400,300,200,100]$ and maximum download bandwidths of $\vec{w} = [2, 1.5, 1, 0.5]$ (in Mb/s). There is a propagation delay of one second from each competing node to the source node. In this example, we consider the dynamic congestion situation. In particular, congestion occurs along the communication path between $\mathcal{N}_1$ and the source node $\mathcal{N}_s$. Congestion occurs twice, at times $t = [30, 40]$ and $t = [50, 60]$. During the congestion, the available bandwidth along the communication path is reduced to 400 kb/s.

Fig. 8 illustrates the instantaneous bandwidth allocation of all four competing nodes for time $t \in [0,100]$. One can make the following observations:

- At time $t = 0$, the system starts at Nash equilibrium with a resource allocation of $\vec{x} = [0.8, 0.6, 0.4, 0.2]$ (in Mb/s).
- Between time $t \in [30, 40]$ (or $t = [50, 60]$), since there is network congestion, the competing node $\mathcal{N}_1$ receives less transfer bandwidth from the source node. Other competing nodes $\mathcal{N}_2$ to $\mathcal{N}_4$ can discover this idle bandwidth resource of 0.4 Mb/s via their bidding messages. The source node $\mathcal{N}_s$ will distribute this excessive bandwidth resource to the other three competing nodes proportionally to their contribution values. New Nash equilibria are reached ($t \in [35 - 40]$ and $t \in [55 - 60]$).
- When the congestion disappears, the competing node $\mathcal{N}_1$ can gain back its proper resource amount of $x_1 = 0.8$ Mb/s. Also, the new Nash equilibrium can be quickly reached and the final resource allocation is $\vec{x} = [0.8, 0.6, 0.4, 0.2]$ Mb/s.

In summary, this example shows that the protocol is adaptive to network congestion. During network congestion, the resource at the source node will not be wasted but rather distributed proportionally to other competing nodes.

Example D (Relationship Between the Step Size δ and the Equilibrium Allocation): We consider one source node $\mathcal{N}_s$ with resource $\mathcal{W}_s = 2$ Mb/s. At time $t = 0$, there are four competing nodes $\mathcal{N}_1$ to $\mathcal{N}_4$ in the system, but node $\mathcal{N}_1$ leaves the system at time 30. These nodes have contribution values $\vec{C} = [400,300,200,100]$ and maximum download bandwidths of $\vec{w} = [2, 1.5, 1, 0.5]$ (in Mb/s). There is a propagation delay of one second from each competing node to the source node.

We consider four scenarios, each using different step size values δ for the four competing nodes. In **D.1**, we have $\vec{\delta} = [0.01, 0.01, 0.01, 0.01]$; in **D.2**, we have $\vec{\delta} = [0.1, 0.1, 0.1, 0.1]$; in **D.3**, we have $\vec{\delta} = [0.01, 0.01, 0.1, 0.01]$; in **D.4**, we have $\vec{\delta} = [0.1, 0.1, 0.01, 0.1]$. Fig. 9 illustrates the *instantaneous* bandwidth allocation for all the competing nodes for $t \in [0,120]$. One can make the following observations:

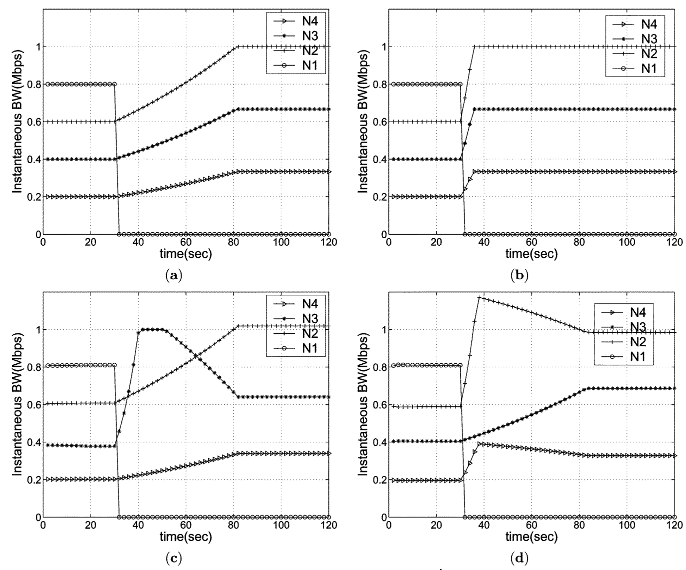


Fig. 9. Instantaneous bandwidth allocations: (a) $\vec{\delta} = [0.01, 0.01, 0.01, 0.01]$; (b) $\vec{\delta} = [0.1, 0.1, 0.1, 0.1]$; (c) $\vec{\delta} = [0.01, 0.01, 0.1, 0.01]$; (d) $\vec{\delta} = [0.1, 0.1, 0.01, 0.1]$.

- In Fig. 9(a) and (b), all competing nodes have the same value of δ (0.01 or 0.1). They show the same equilibrium allocation for $\mathcal{N}_2$, $\mathcal{N}_3$ and $\mathcal{N}_4$ after $\mathcal{N}_1$ leaves the system. The difference in these two scenarios is that from the time $\mathcal{N}_1$ leaves the system, it takes around 50 seconds to reach the new equilibrium in (a), but 5 seconds in (b). It is intuitive that a larger δ value enables faster convergence to the new equilibrium point.
- Fig. 9(c) shows the scenario when all competing nodes have δ value 0.01 except node $\mathcal{N}_3$, which has $\delta_3 = 0.1$. From $t = 30$, node $\mathcal{N}_3$ gains its resource much faster than $\mathcal{N}_2$ and $\mathcal{N}_4$, and gains even more resource than it gains in equilibrium from time 35 to 80. On the other hand, by comparing Fig. 9(c) with (a), we find that both the original equilibrium and the new equilibrium are different. Node $\mathcal{N}_3$ gains less resource in (c) than in (a). Theoretically, when all the competing nodes have the same δ value, the actual equilibrium should coincide with the theoretical Nash equilibrium. But a larger δ value achieves less resource in actual equilibrium.
- Fig. 9(d) shows the opposite of the previous experiment, when $\mathcal{N}_3$ now has a *smaller* δ value than the other competing nodes. In this case, node $\mathcal{N}_3$ reaches its new equilibrium slower, but gains more resource than in the theoretical Nash equilibrium.

In summary, these examples show that δ affects both the equilibrium solution and the rate of convergence to the equilibrium. In general, the larger the value of δ, the faster the convergence to the new equilibrium, but with less resource gained at the equilibrium point.

VIII. Conclusion

In this paper, we have presented a framework for building incentive P2P networks. The framework consists of the resource allocation mechanism RBM-IU and a network protocol for competing nodes to reach equilibria of the competition game induced by RBM-IU. Our solution is efficient: 1) RBM-IU can be implemented by a linear time algorithm; 2) the feedback bidding messages used by the competing nodes are simple; and 3) RBM-IU achieves Pareto-optimality allocation results. The robustness of the solution is evidenced by the fact that all the competing nodes can reach the equilibrium solutions of the competition game. The justification for the source node to use our protocol is its guarantee of the Pareto optimality. On the other hand, competing nodes are motivated to use the protocol because it guarantees Nash equilibrium. We show that the protocol can be extended to heterogeneous nodes with different utility functions. Convergence analysis is carried out to show the existence of the Nash equilibrium. Lastly, we also show that the protocol is adaptive to various nodes arrival and departure events, as well as in different forms of network congestion.

ACKNOWLEDGMENT

The authors would like to thank the anonymous referees for their insightful and helpful comments.

REFERENCES

[1] E. Adar and B. Huberman, "Free riding on Gnutella," *FirstMonday*, vol. 5, no. 10, Oct. 2000 [Online]. Available: http://firstmonday.org/issues/issue5_10/adar/index.html

[2] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[3] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong, "Freenet: A distributed anonymous information storage and retrieval system," *Lecture Notes in Computer Science*, vol. 2009, p. 46, 2001.

[4] eMule Peer-To-Peer File Sharing Client. [Online]. Available: http://www.emule-project.net

[5] E. J. Friedman and D. C. Parkes, "Pricing WiFi at Starbucks: Issues in online mechanism design," in *Proc. 4th ACM Conf. Electronic Commerce*, 2003, pp. 240–241.

[6] A. C. Fuqua, T. Ngan, and D. S. Wallach, "Economic behavior of peer-to-peer storage networks," presented at the Workshop on Economics of Peer-to-Peer Systems, Berkeley, CA, Jun. 2003.

[7] P. Golle, K. Leyton-Brown, I. Mironov, and M. Lillibridge, "Incentives for sharing in peer-to-peer networks," in *Proc. 3rd ACM Conf. Electronic Commerce*, Tampa, FL, Oct. 2001, pp. 264–267.

[8] M. Gupta, P. Judge, and M. Ammar, "A reputation system for peer-to-peer networks," in *Proc. 13th NOSSDAV*, Monterey, CA, Jun. 2003, pp. 144–152.

[9] G. Hardin, "The tragedy of the commons," *Science*, vol. 162, pp. 1243–1248, 1968.

[10] Kazaa Peer-To-Peer File Sharing Client. [Online]. Available: http://www.kazaa.com

[11] R. Ma, S. Lee, J. Lui, and D. Yau, "An incentive mechanism for P2P networks," in *Proc. Int. Conf. Distributed Computing Systems (ICDCS)*, 2004, p. 2.

[12] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic Theory*. Oxford, U.K.: Oxford Univ. Press, 1995.

[13] N. Nisan and A. Ronen, "Algorithmic mechanism design," in *Proc. 31st ACM Symp. Theory of Computing*, 1999, pp. 129–140.

[14] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content addressable network," in *Proc. ACM SIGCOMM*, 2001, pp. 161–172.

[15] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," *Lecture Notes in Computer Science*, vol. 2218, pp. 329–350, 2001.

[16] S. Shenker, "Fundamental design issues for the future Internet," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 7, pp. 1141–1149, Sep. 1995.

[17] J. Shneidman and D. Parkes, "Rationality and self-interest in peer to peer networks," in *Proc. 2nd Int. Workshop on Peer-to-Peer Systems (IPTPS'03)*, Feb. 2003, pp. 139–148.

[18] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for Internet applications," in *ACM SIGCOMM*, 2001, pp. 149–160.

[19] R. K. Sundaram, *A First Course in Optimization Theory*. Cambridge, U.K.: Cambridge Univ. Press, 1996, ch. 7.

[20] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph, Tapestry: An infrastructure for fault-tolerant wide-area location and routing Univ. California, Berkeley, Tech. Rep. UCB/CSD-01-1141, 2001.

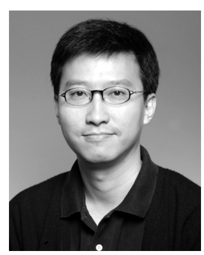
[21] S. Zhong, Y. Yang, and J. Chen, "Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks," in *Proc. IEEE INFOCOM*, 2003, pp. 1987–1997.

Richard T. B. Ma received the B.Sc. (first class honors) degree in computer science in 2002 and the M.Phil. degree in computer science and engineering in 2004, both from the Chinese University of Hong Kong. He is now pursuing the Ph.D. degree at Columbia University, New York, NY.

His research interests are in incentive protocols and stochastic modeling.

Sam C. M. Lee received the B.Sc. and M.Phil. degrees in computer science from the Chinese University of Hong Kong, where he is now pursuing the Ph.D. degree.

His research interests are in network communication and mathematical optimization theory.



John C. S. Lui (M'93–SM'02) received the Ph.D. degree in computer science from the University of California at Los Angeles (UCLA).

He is currently Chairman of the Department of Computer Science and Engineering (CSE) at the Chinese University of Hong Kong (CUHK). While a Ph.D student, he spent a summer in the IBM T. J. Watson Research Laboratory. Upon graduation, he joined the IBM Almaden Research Laboratory/San Jose Laboratory and participated in various research projects on file systems and parallel I/O architectures. He later joined the CSE department at CUHK. His research interests span both systems and theory/mathematics. His current research interests are in theoretic/applied topics in data networks, distributed multimedia systems, network security, OS design issues and mathematical optimization and performance evaluation theory.

Prof. Lui has received various departmental teaching awards and the CUHK Vice-Chancellor's Exemplary Teaching Award in 2001. He is a co-recipient of the IFIP WG 7.3 Performance 2005 Best Student Paper Award. Currently, he is an associate editor of the *Performance Evaluation Journal*, a member of ACM, and an elected member of the IFIP WG 7.3. He was the TPC co-chair of ACM SIGMETRICS 2005 and is currently on the SIGMETRICS Board of Directors.

David K. Y. Yau (M'97) received the B.Sc. (first class honors) degree from the Chinese University of Hong Kong in 1989, and the M.S. and Ph.D. degrees from the University of Texas at Austin in 1992 and 1997, respectively, all in computer science.

From 1989 to 1990, he was with the Systems and Technology group of Citibank, NA. He was the recipient of an IBM graduate fellowship, and is currently an Associate Professor of Computer Sciences at Purdue University, West Lafayette, IN.

Prof. Yau received a National Science Foundation (NSF) CAREER Award in 1999, for research on network and operating system architectures and algorithms for quality of service provisioning. His other research interests are in network security, value-added services routers, and mobile wireless networking. He is a member of the ACM and IEEE. He serves on the editorial board of the IEEE/ACM TRANSACTIONS ON NETWORKING. He is TPC Co-chair of IEEE International Workshop on Quality of Service (IWQoS) 2006 and Vice General Chair of IEEE International Conference on Network Protocols (ICNP) 2006.