2005

Optimal Peer Selection for P2P Downloading and Streaming

Micah Adler
University of Massachusetts - Amherst

Rakesh Kumar
Polytechnic University - NY

Keith Ross
Polytechnic University - NY

Dan Rubenstein
Columbia University

Torsten Suel
Polytechnic University - NY

See next page for additional authors

Authors
Micah Adler, Rakesh Kumar, Keith Ross, Dan Rubenstein, Torsten Suel, and David D. Yao

Optimal Peer Selection for P2P Downloading and Streaming

Micah Adler*, Rakesh Kumar†, Keith Ross‡, Dan Rubenstein§, Torsten Suel¶ and David D. Yao‖
*Dept. of Computer Science, University of Massachusetts Amherst, MA; Email: micah@cs.umass.edu
†Dept. of Electrical Engineering, Polytechnic University, NY; Email: rkumar04@utopia.poly.edu
‡Dept. of Computer and Information Science, Polytechnic University, NY; Email: ross@poly.edu
§Dept. Of Electrical Engineering, Columbia University, NY; Email: danr@ee.columbia.edu
¶Dept. of Computer and Information Science, Polytechnic University, NY; Email: suel@poly.edu
‖IEOR Dept., Columbia University, NY; Email: yao@columbia.edu

***Abstract*— In a P2P system, a client peer may select one or more server peers to download a specific file. In a P2P *resource economy*, the server peers charge the client for the downloading. A server peer's price would naturally depend on the specific object being downloaded, the duration of the download, and the rate at which the download is to occur. The optimal peer selection problem is to select, from the set of peers that have the desired object, the subset of peers and download rates that minimizes cost. In this paper we examine a number of natural peer selection problems for both *P2P downloading* and *P2P streaming*. For downloading, we obtain the optimal solution for minimizing the download delay subject to a budget constraint, as well as the corresponding Nash equilibrium. For the streaming problem, we obtain a solution that minimizes cost subject to continuous playback while allowing for one or more server peers to fail during the streaming process. The methodologies developed in this paper are applicable to a variety of P2P resource economy problems.**

***Keywords* — Economics, Mathematical Programming/optimization.**

I. Introduction

Today many computers participate in peer-to-peer file sharing applications in which widely distributed nodes contribute storage and bandwidth resources [1], [2], [3], [4]. It is widely documented, however, that these P2P systems are havens for "free riders": a significant fraction of users do not contribute any significant resources, and a minute fraction of users contribute the majority of the resources [1], [5], [6]. Thus, to improve the performance of existing P2P file sharing systems, and to enable new classes of P2P applications, a compelling incentive system needs to be put in place to encourage users to make their resources available.

Now suppose the existence of an online marketplace where entities - such as peers, companies, users etc. - buy and sell surplus resources. In this market place, a peer might purchase storage and bandwidth from a dozen other peers for the purpose of remotely backing up its files; a content publisher might purchase storage and bandwidth from thousands of peers to create a peer-driven content distribution network; a biotechnology company might purchase CPU cycles from thousands of peers for distributed computation. If such a flourishing resource market existed, individual peers would be incited to contribute their resources to the marketplace, thereby unleashing the untapped resource pool.

We envision a *free-market resource economy* in which peers buy and sell resources directly from each other [7], [8]. In this market, selling peers are free to set the prices of their resources as they please. A client peer, interested in purchasing a specific resource, is permitted to "shop" the different server peers and choose the peers that best satisfy its needs at the best prices. The "money" paid by the client peers and earned by the server peers may be real money, or some pseudo-currency similar to frequent flyer miles. When a seller earns money, it can later spend the money in the resource market, obtaining resources from other seller peers.

In a P2P resource economy a client peer can select one or more server peers for downloading a file or streaming a stored audio/video object. In general, multiple server peers may have the object available, with each offering a different price. A serving peer's price will naturally depend on the specific object, the duration of the transmission, and the rate at which the transmission is to occur. The client may obtain different portions of the object in parallel from different serving peers, as is currently already the case with KaZaA and other file sharing systems. The optimal peer selection problem is to select, from the subset of the peers that have the desired object, a set of peers and downloading rates that minimize cost and/or delay.

More specifically, when a client peer wants to obtain a specific object, the following steps may be taken:

(1) **Discovery:** The client first uses a look-up service to discover server peers that have a copy of the object. KaZaA is one example of such a lookup service, but structured DHTs could also be used

for this.

(2) **Pricing:** The client then queries the server peers for their prices. Alternatively, price information might be available via the lookup service.

(3) **Reputation:** The client may also use a reputation service to determine the reliability of each of the server peers. (Reputation services are beyond the scope of this paper; see [9], [10].)

(4) **Server and rate selection:** From the subset of reputable server peers offering the object, the client peer selects the server peers from which it obtains the object. The client obtains different segments of the object from each of the selected server peers. The servers may offer a segment at different upload rates, and advertise an upload price as a function of the upload rate. The client peer will naturally want to choose the server peers (and rates) to minimize cost and delay.

(5) **Payment:** Money is transferred from the client peer to the server peers. A protocol for transferring money in a P2P resource market is described in [7], [8].

In this paper we study the optimal peer selection problem for two delivery schemes: (i) **streaming**, where the portions of the object must arrive in a timely manner such that the client peer can render the object without glitches; (ii) **downloading**, where the client wants to receive the entire file as quickly and inexpensively as possible, but does not render the file during the download. For both schemes, there are many variations of the optimal peer selection problem. For example, for downloading, one can minimize the download delay subject to a cost constraint, or minimize the cost subject to a download delay constraint. We do not attempt to solve all possible variations in an encyclopedic manner. Instead, we have formulated a few problems that we feel are particularly representative and important. The techniques developed in this paper can be extended to other natural variations.

For the downloading problem, we formulate and solve the problem of minimizing the (parallel) downloading time subject to a budget constraint. We find that the optimal solution is a greedy one in which costly servers are fully excluded from downloading. We also determine the Nash equilibrium for the servers' prices. For the streaming problem, we consider the problem of minimizing cost subject to a continuous-playback constraint. Because server peers often fail (because of intentional or unintentional disconnects from the P2P system), we also consider peer failures in our formulation. We are able to find the optimal solution when any subset of f chosen peers may fail. We solve the streaming problem for both convex and concave cost functions.

For both schemes, a content publisher may also be an active component of the system. For example, CNN.com may contract with a large number of peers to store chunks of video files. When another peer, say Alice, asks CNN to see a video, CNN may select the peers on Alice's behalf. The selected peers would then either stream or upload the object, depending on the delivery scheme. The methodology developed in this paper is applicable when the client peer is to select the server peers, or when an intermediate peer (such as CNN) selects the server peers on the client's behalf.

The contribution of our work is the development of theoretical methodologies for these types of peer selection problems. To facilitate the analysis, we use a model of the network where delivery rates can more or less be guaranteed. As discussed in the next section, this assumption can be partially justified due to the abundance of bandwidth in the Internet core. Implementation of peer selection techniques in more accurate networking models, while not explicitly addressed in this paper, can more easily be accomplished by using our results as a guide or starting point.

II. PRICING MODEL

In this section we describe our pricing model. As mentioned in the introduction, each server is free to set its own prices. Consider a server peer i. As part of a delivery session, peer i will transfer a portion of the bytes of some object o to a client peer. For such a delivery, the server peer will fix an appropriate price that could naturally depend on:

- **The object itself:** For example, recently-released objects (e.g., videos) might be more expensive than older objects.
- **Rate of transfer:** The server may be able to transfer the object at different rates, and charge different prices for different rates. In this paper we suppose that peer i has a maximum transfer rate u_i and can transfer at any rate b in the interval $[0, u_i]$.
- **Duration of transfer:** The longer the transfer (at some constant rate), the more a server should charge. We typically expect the server's price to be proportional to the duration of the transfer.

For a particular object o, we consider pricing functions of the form

$$\text{price} = c_i(b_i) \cdot t_i$$

where $b_i \in [0, u_i]$ is the rate (in bytes per second) of the transfer from server i and t_i is the duration of the transfer for server i. Thus, for a given transfer rate b_i the price is proportional to the duration of transfer.

It is natural to assume that $c_i(0) = 0$ and $c_i(\cdot)$ is non-decreasing for all $i = 1, \ldots, I$. Furthermore, depending on the broader context, a cost-rate function $c_i(\cdot)$ may be either convex or concave. For example, if a server is sharing its upstream bandwidth resources between the P2P downloading application and other applications,

then the marginal cost to the server of allocating more bandwidth to the streaming application can be increasing with b_i, in which case $c_i(\cdot)$ is convex. On the other hand, the server may prefer to sell in bulk to individual clients (because of client acquisition costs), in which case $c_i(\cdot)$ is naturally concave. We therefore analyze both the convex and concave cases for both the downloading and streaming problems.

Note that if a server sends at rate b_i bytes/sec for a duration of t_i seconds, then the server transfers $x_i = b_i t_i$ bytes. This implies that the price can also be defined in terms of the rate b_i and the object size x_i since the price is equal to $c_i(b_i)t_i = [c_i(b_i)/b_i]x_i$. Thus, $c(b)$ is the cost per unit time for data streamed at rate b, and $c(b)/b$ is the cost per byte for data streamed at rate b.

Before proceeding, let us examine more carefully what it means for a server to be able to transfer bytes at a specific rate b. A server i will have access to the Internet with some upstream rate u_i. At any given time, the server peer i could be transferring files to multiple peers, with each file transfer taking place at its negotiated rate. In order to meet its commitment, server i, of course, must ensure that the sum of all the committed transfer rates does not exceed its upstream access rate u_i. In today's Internet (and in the foreseeable future), the bottleneck is typically in the access and not in the Internet core. Furthermore, in most broadband residential connections today (including cable modem and ADSL), the upstream rate is significantly less than the downstream rate. Thus, in many cases the bandwidth bottleneck between server and client is the server's upload rate. It is therefore reasonable to assume that a server can provide an offered rate b as long as the sum of server's committed ongoing rates is less than u_i. Even when this assumption is unreasonable, the formulations and results of this paper provide a framework and starting point for studying scenarios without the assumption.

There will be situations, however, when the server will not be able to honor its commitment due to unusual congestion or service failures in the core. In this case, the client peer may want some form of a refund. Furthermore, either the server or the client may be dishonest and may not agree on whether the service was actually rendered. Thus, some form of arbitration - preferably lightweight - may be needed in a realistic P2P resource market; see, e.g., [7]. In Section IV, we will describe a client strategy that allows one or more of the contracted peers to fail, either because of technical problems or dishonesty.

III. Optimal Peer Selection for Downloading

As discussed in the Introduction, in this paper we explore the optimal peer selection problem for two delivery schemes, streaming and downloading. In this section we consider the downloading problem, in which case the client wants to receive the entire file as quickly and inexpensively as possible, but does not render the file while downloading.

Naturally, a client desiring a specific object o would like to obtain the object as quickly as possible and at lowest possible cost. These two objectives will often be conflicting, as servers that provide high transfer rates may also demand high per-byte transfer costs. There are many ways to formulate an optimization problem that takes into account these conflicting goals. In this section, we consider one natural formulation: the client selects the peers and rates in order to minimize the total download time subject to a budget constraint for the download. (Although not considered here, the problem of minimizing the cost subject to a constraint on the download time is also tractable.)

We can now define the optimal downloading problem. Consider a client peer that wants to download a file o. Let F be the size (in bytes) of the file. As described in the introduction, the client peer uses a location service to find the set of peers, denoted in the following as $\{1, \cdots, I\}$, that have a copy of the file. Each server peer i in this set advertises a price function $c_i(b_i)$, $b_i \in [0, u_i]$. We assume that the client peer has a budget K for this particular download, that is, the client peer is prepared to spend up to K units on the download.

Let t_i be the transfer time for server i in this download, i.e., the amount of time that i participates in the download. If the client peer does not select server peer i, then $t_i = 0$. The number of bytes transferred by server peer i is $b_i t_i$. Because the client wants to obtain the entire file, we have $b_1 t_1 + \cdots + b_I t_I = F$. Our optimal peer selection problem is to determine b_i, $i = 1, \ldots, I$, and t_i, $i = 1, \ldots, I$, that minimize the total download time subject to the budget constraint. Because the client is downloading from multiple server peers in parallel, the total download time is the maximum of the t_i. Thus, the optimization problem is

$$\min \max\{t_1, \ldots, t_I\} \tag{1}$$

subject to

$$c_1(b_1)t_1 + \cdots + c_I(b_I)t_I \le K \tag{2}$$

$$b_1 t_1 + \cdots + b_I t_I \ge F \tag{3}$$

$$0 \le b_i \le u_i \quad i = 1, \ldots, I \tag{4}$$

$$t_i \ge 0 \quad i = 1, \ldots, I \tag{5}$$

Note that since the the cost functions $c_i(b)$ are non-decreasing, any optimal solution must make the constraint in (3) binding. (For otherwise, we can decrease b_i and/or t_i for some i, while maintaining feasibility and not increasing the objective value.)

Also note that without the budget constraint (2), the optimal solution is given by $b_i = u_i$, $t_i = F/(u_1 + \cdots +$

u_I), $i = 1, \dots, I$, and the resulting minimal download time is $F/(u_1 + \cdots + u_I)$. In other words, without the budget constraint, in the optimal solution, the client downloads from all of the servers at their maximum rates until all client as received all the bytes in the file. Thus $F/(u_1 + \cdots + u_I)$ is a lower bound for the value of the optimization problem (1).

A. *Concave Pricing Functions*

As discussed in Section II, depending on the broader context, a cost function $c_i(\cdot)$ may be either convex or concave. We first consider the scenario when $c_i(b)$, $b \in [0, u_i]$ is concave, for all $i = 1, \dots, I$. As it will become evident below, this scenario also provides the solution for the case when each server is capable of transmitting at only the rate u_i at cost $c_i = c_i(u_i)/u_i$ per byte.

Lemma 1: Suppose for all $i = 1, \dots, I$, $c_i(b)$ is concave for $b \in [0, u_i]$. Then, there exists an optimal solution such that for each $i = 1, \dots, I$: $b_i = 0$ if $t_i = 0$, and $b_i = u_i$ if $t_i > 0$.

Proof: If $t_i = 0$ for some i, then letting $b_i = 0$ will not affect either the constraints or the objective value.

To argue the case of $t_i > 0$ implying $b_i = u_i$, we first note that the concavity of $c_i(b)$, along with $c_i(0) = 0$, implies that $c_i(b)/b$ is non-increasing in b. Hence, start with any optimal solution $(b_i, t_i)_{i=0}^I$, if $t_j > 0$ and $b_j < u_j$ for some j, we can then modify the solution to

$$b'_j = u_j, \qquad t'_j = b_j t_j / u_j \leq t_j.$$

This way, $b'_j t'_j = b_j t_j$, hence, the constraint in (3) remains intact, while the constraint in (2) continues to hold since

$$c_j(b'_j) t'_j \leq c_j(b_j) t_j,$$

which is equivalent to

$$c_j(b'_j)/b'_j \leq c_j(b_j)/b_j,$$

i.e., the non-increasing property of $c_j(b)/b$ mentioned above. Furthermore, since $t'_j \leq t_j$, the objective value will not increase. ♣

The above lemma implies that the optimal decision on the rates, b_i's, follows directly from the optimal t_i's, and hence can be eliminated from the problem formulation. Specifically, letting $c_i := c_i(u_i)/u_i$, the original problem can be reduced to the following equivalent linear program (LP):

$$\min \; y \tag{6}$$

$$\text{s.t.} \quad \sum_i c_i x_i \leq K, \tag{7}$$

$$\sum_i x_i \geq F, \tag{8}$$

$$0 \leq x_i \leq u_i y, \quad \forall i. \tag{9}$$

To see, the equivalence, suppose (b_i^*, t_i^*), $i = 1, \dots, I$, is an optimal solution to the original problem. Then, letting

$$y^* = \max_i \{t_i^*\}; \quad x_i^* = b_i^* t_i^*, \; i = 1, \dots, I$$

results in a feasible solution to the LP, taking into account $b_i^* = 0$ or u_i for all i. Conversely, if $(y^*, x_1^*, \dots, x_I^*)$ is the optimal LP solution, then a feasible solution to the original problem is obtained by letting for each i, $b_i = t_i = 0$ if $x_i = 0$, and $b_i = u_i$ and $t_i = x_i/u_i$ if $x_i > 0$.

Therefore, it suffices to solve the LP problem. To this end, re-order the server peers such that

$$0 < c_1 < \cdots < c_I. \tag{10}$$

Also, denote

$$B_j := \sum_{i=1}^{j} u_i, \quad \beta_j := \sum_{i=1}^{j} u_i c_i. \tag{11}$$

It is easy to verify that β_j/B_j is increasing in j, since

$$\frac{\beta_j}{B_j} \leq \frac{\beta_{j+1}}{B_{j+1}} \quad \text{iff} \quad \beta_j \leq B_j c_{j+1},$$

and the last inequality follows from (10). We note that if $\beta_1/B_1 > K/F$, that is, $K < F c_1$, then there is no feasible solution to the LP. Henceforth, we assume $\beta_1/B_1 \leq K/F$.

Theorem 1: Suppose for all $i = 1, \dots, I$, $c_i(b)$ is concave for $b \in [0, u_i]$. Then the solution to the LP, and hence the original downloading problem, takes the following form: (a) If $K/F \geq \beta_I/B_I$, then $x_i = u_i y$ for all i, where $y = F/B_I$. (b) Otherwise, suppose for some $j \leq I$ we have

$$\frac{\beta_j}{B_j} > \frac{K}{F} \geq \frac{\beta_{j-1}}{B_{j-1}}.$$

Then,

$$x_i = u_i y, \quad i \leq j-1; \qquad x_j = F - y B_{j-1};$$

$$x_{j+1} = \cdots = x_n = 0;$$

where

$$y = \frac{F c_j - K}{c_j B_j - \beta_j}.$$

In both cases, y is the optimal objective value.

Proof: If $\beta_I/B_I \leq K/F$, then it is easily seen that $x_i = u_i y$ for all i, where $y = F/B_I$, is a feasible solution to the LP, with a download time equal to the lower bound F/B_I. Thus, this solution is clearly optimal.

Next, consider the case of $\beta_I/B_I < K/F$. The dual of the above LP is as follows, with the dual variables v and w corresponding, respectively, to the constraints in (7) and (8), and z_i corresponding to $x_i \leq u_i y$ in (9):

$$\min Fw - Kv \tag{12}$$
$$\text{s.t.} \quad w - z_i - c_i v \le 0, \tag{13}$$
$$\sum_i u_i z_i \le 1, \tag{14}$$
$$v \ge 0,\ w \ge 0,\ z_i \ge 0,\ \forall i.$$

Below, we start deriving a dual feasible solution, which then leads to a primal feasible solution via complementary slackness. Once these are verified — dual and primal feasibility and complementary slackness — the problem is completely solved.

Letting the constraints in (13) and (14) be binding, we get:

$$z_i = w - c_i v, \tag{15}$$
$$w = \frac{1 + v\beta_I}{B_I}. \tag{16}$$

Then, the dual objective becomes

$$Fw - Kv = \frac{F}{B_I} + \Big(\frac{F\beta_I}{B} - K\Big)v.$$

Consider, for the time being, $\frac{\beta_I}{B_I} > \frac{K}{F} \ge \frac{\beta_{I-1}}{B_{I-1}}$. Then $v = v_I := 1/(c_I B - \beta)$. Note $c_I B_I > \beta_I$ follows from (10); and for all i,

$$z_i = \frac{1}{B_I} + \Big(\frac{\beta_I}{B_I} - c_i\Big)v_I \ge 0$$

follows from

$$v_I \le \frac{1}{c_i B_I - \beta_I}, \quad \forall i:\ c_i B > \beta_I,$$

since $c_n \ge c_i$. Also note that $z_n = 0$.

The dual feasible solution results in a dual objective value as follows:

$$\frac{F}{B_I} + \Big(\frac{F\beta_I}{B_I} - K\Big)\cdot\frac{1}{c_I B_I - \beta_I} = \frac{Fc_I - K}{c_I B_I - \beta_I}. \tag{17}$$

For the corresponding primal solution, consider the following:

$$x_i = u_i y, \quad \forall i \ne I; \qquad x_I = F - yB_{I-1}; \tag{18}$$

where y is the primal objective value, obtained via substituting the above solution into (7) and making the latter an equality:

$$y\beta_{I-1} + c_I\Big(F - yB_{I-1}\Big) = K,$$

from which we can obtain

$$y = \frac{Fc_I - K}{c_I B_{I-1} - \beta_{I-1}} = \frac{Fc_I - K}{c_I B_I - \beta_I}, \tag{19}$$

i.e., the primal objective value is equal to the dual objective value in (17).

We still need to verify primal feasibility and complementary slackness. First note that $y \ge 0$ follows from (17): both terms on its LHS are positive. Then $x_n \ge 0$ is equivalent to

$$\frac{F}{B_{I-1}} \ge \frac{F}{B_I} + \Big(\frac{F\beta_I}{B_I} - K\Big)v_I,$$

which simplifies (with some algebra) to $\frac{K}{F} \ge \frac{\beta_{I-1}}{B_{I-1}}$, the assumed condition in Case (ii). Other aspects of primal feasibility hold trivially. Complementary slackness is readily verified: all primal variables are positive, and all dual constraints are binding; all dual variables except z_I are positive, and all primal constraints except $x_I \le u_I y$, the I-th constraint in (9), are binding.

Next suppose K/F falls into the following range:

$$\frac{\beta_{I-1}}{B_{I-1}} > \frac{K}{F} \ge \frac{\beta_{I-2}}{B_{I-2}}.$$

Then, the dual solution is:

$$v = v_{I-1} := \frac{1}{(c_{I-1}B_{I-1} - \beta_{I-1})}, \quad w = \frac{1 + v_{I-1}\beta_{I-1}}{B_{I-1}},$$

and

$$z_i = w - c_i v, \quad i \le I-1; \qquad z_I = 0.$$

The primal solution is:

$$x = u_i y, \quad i \le I-2; \quad x_{I-1} = F - yB_{I-2}, \quad x_I = 0;$$

and

$$y = \frac{Fc_{I-1} - K}{c_{I-1}B_{I-2} - \beta_{I-2}} = \frac{Fc_{I-1} - K}{c_{I-1}B_{I-1} - \beta_{I-1}}.$$

Feasibility (primal and dual) and complementary slackness can be verified as before. ♣

Roughly speaking, Theorem 1 indicates that the client downloads in parallel from the least expensive servers at their maximum rates. The number of parallel servers is determined from the budget constraint. To meet the budget constraint with equality, one of the selected servers transmits for less time than the other parallel servers. Again, this result is true for two important special cases: (i) when the the cost per byte is linear in b for all servers; and (ii) when each selected server can transmit only at one rate.

B. Convex Pricing Functions

We now consider the downloading problem for convex pricing functions. Specifically, in this section we suppose that $c_i(b)$ is convex with respect to b for all $i = 1, \dots, I$. We'll see that this scenario gives rise to a completely different form for the optimal solution. In particular, for many natural pricing functions, all I servers will be selected with none of servers transmitting at its maximal rate. Let t^* denote the minimal download time for the downloading problem.

Theorem 2: Suppose $c_i(b)$ is convex in b for all $i = 1, \ldots, I$. Then, there exists an optimal solution for the downloading problem with $t_i = t^*$ for all i.

Proof: First note that the convexity of $c_i(b)$ implies that $c_i(b)/b$ is non-decreasing in b. Suppose $t_i < t^*$ for some i. Then, we can increase t_i to t^* while decreasing b_i to $b_i' = b_i t_i / t^*$. This way, there is no increase in the objective value; the constraint in (3) remains intact; and the constraint in (2) still holds, since

$$c_i(b_i)t_i \geq c_i(b_i')t^*,$$

i.e.,

$$c_i(b_i)/b_i \geq c_i(b_i')/b_i',$$

which follows from the non-decreasing property of $c_i(b)/b$ mentioned above, as $b_i' \leq b_i$. ♣

We remark that Theorem 2 does not necessarily imply that all peers are selected when the cost functions are convex. Indeed, for an optimal solution with $t_i = t^*$ for all i, we may have $b_j = 0$ for one or more server peers j. The peers with $b_j = 0$ are not selected.

Theorem 2 leads to a recipe for identifying the optimal solution. To this end, again let t^* denote the optimal download time and let b_i, $i = 1, \ldots, I$, be the corresponding optimal rates. By Theorem 2 and (2)-(5) we know that these rates must satisfy

$$c_1(b_1)t^* + \cdots + c_I(b_I)t^* \leq K \tag{20}$$

$$b_1 + \cdots + b_I = F/t^* \tag{21}$$

$$0 \leq b_i \leq u_i \quad i = 1, \ldots, I \tag{22}$$

Thus, to find the optimal solution we can search over all values of $t \geq F/(u_1 + \cdots + u_I)$; for each value of t we solve the optimization problem:

minimize

$$c_1(b_1)t + \cdots + c_I(b_I)t \tag{23}$$

subject to

$$b_1 + \cdots + b_I = F/t \tag{24}$$

$$0 \leq b_i \leq u_i \quad i = 1, \ldots, I \tag{25}$$

The optimal t^* is found by finding the smallest t such that the objective value for (23)-(25) is no greater than K. The optimization problem is a marginal analysis problem; it can be efficiently solved with the techniques in Section IV. Moreover, since the cost functions are convex, the value of the optimization problem (23)-(25) is convex in t (see Section IV). Thus, the optimal t can be found with a binary search.

We remark in passing that another interesting aspect to consider is a restriction on the maximum download rate d to the client. In this case we would have the additional constraint $b_1 + \cdots + b_I \leq d$ at all times. With asymmetric access (as in ADSL and most cable modem access systems), the downstream bandwidth is typically larger than the upstream bandwidth. However, a client peer downloading from multiple server peers would eventually saturate the client's downstream bandwidth d. We note that if each server peer can transmit only at one rate u_i then the problem of selecting an optimal set of peers and rates for downloading a file under a client bandwidth restriction can be easily shown to be NP-Complete, by a reduction from the Knapsack problem [11]. The problem of selecting a set of servers with aggregate bandwidth as close as possible, but not larger than, the client bandwidth is basically just the Knapsack problem.

From the discussions in the last two subsections, in particular the proofs of Lemma 1 and Theorem 2, it is clear that the required concavity or convexity of the cost functions $c_i(b)$ can be relaxed to the weaker condition of $c_i(b)/b$ being non-increasing or non-decreasing, respectively. Note that a function $f(x)$ with $x \geq 0$ and $f(0) = 0$ is termed "star-shaped" if $f(x)/x$ is non-decreasing. This is a standard property in reliability theory, refer to [12]. It is well-known (and easy to verify) that a convex function is star-shaped, but a star-shaped function need not be either convex or concave.

C. Nash Equilibrium

Suppose a client is interested in downloading the file o of size F, and it makes its budget K known to the server peers. Suppose now that each peer server i is free to set its pricing function $c_i(b)$, $0 \leq b \leq u_i$. We now turn our attention to the problem of what pricing function a server should propose to this client. To simplify the analysis, suppose each server can transmit at either rate 0 or u_i. Let $c_i := c_i(u_i)/u_i$. We refer to $\mathbf{c} = (c_1, \ldots, c_I)$, consisting of all the proposed prices, as the pricing vector.

For a given pricing vector $\mathbf{c}$, the peer client will determine the optimal number of bytes to allocate to each server. Let $(x_1(\mathbf{c}), \ldots, x_I(\mathbf{c}))$ denote the optimal allocation for pricing vector $\mathbf{c} = (c_1, \ldots, c_I)$. For a given $\mathbf{c}$, server i earns revenue $R_i(\mathbf{c}) = c_i x_i(\mathbf{c})$. Assuming that the servers are rational, a server i would modify its cost c_i if it could increase its revenue $R_i(\mathbf{c})$. A pricing $(c_1^*, c_2^*, \ldots, c_I^*)$ is said to be a *Nash equilibrium* if

$$R_i(c_1^*, \ldots, c_i^* + \delta, \ldots, c_I^*) \leq R_i(c_1^*, \ldots, c_i^*, \ldots, c_I^*)$$

for all δ and all $i = 1, \ldots, I$. In other words, $(c_1^*, c_2^*, \ldots, c_I^*)$ is a Nash equilibrium if no server can improve its revenue by unilaterally changing its price.

To analyze the Nash equilibrium, we remark that when $\hat{c}_i = K/F$ for all $i = 1, \ldots, I$, then $\hat{x}_i = u_i F/(u_1 + \cdots + u_I)$, $i = 1, \ldots, I$, is optimal for the downloading problem. To see this, note that $(\hat{x}_1, \ldots, \hat{x}_I)$ is feasible for the pricing vector $\hat{\mathbf{c}} = (\hat{c}_1, \ldots, \hat{c}_I)$ and gives the

minimal cost $F/(u_1 + \cdots + u_I)$. We can now state the following corollary of Theorem 1.

Corollary 1: $\hat{c}_i = K/F$ for all $i = 1, \ldots, I$ is a Nash equilibrium.

Proof: We first observe that when $\hat{c}_i = K/F$ for all $i = 1, \ldots, I$, then $\hat{x}_i = u_i F/(u_1 + \cdots + u_I)$, $i = 1, \ldots, I$ is optimal for (6). To see this, note that $(\hat{x}_1, \ldots, \hat{x}_I)$ is feasible for the pricing vector $\hat{\mathbf{c}} = (\hat{c}_1, \ldots, \hat{c}_I)$ and gives the minimal cost $F/(u_1 + \cdots + u_I)$.

Now let $\tilde{\mathbf{c}}$ be a pricing vector that is identical to $\hat{\mathbf{c}}$ for all components except component j, for which $\tilde{c}_j = \hat{c}_j + \delta$. It suffices to show that for $\delta \neq 0$,

$$R_j(\tilde{\mathbf{c}}) < R_j(\hat{\mathbf{c}}). \tag{26}$$

First consider the case $\delta < 0$. For the pricing vector $\tilde{\mathbf{c}}$, the allocation $(\hat{x}_1, \ldots, \hat{x}_I)$ is optimal since it is feasible and it gives the minimal cost $F/(u_1 + \cdots + u_I)$. Thus $R_j(\tilde{\mathbf{c}}) = (\hat{c}_j + \delta)\hat{x}_j < \hat{c}_j \hat{x}_j = R_j(\hat{\mathbf{c}})$, establishing (26). Now suppose $\delta > 0$, so that $\tilde{c}_j > \tilde{c}_i$ for all $i \neq j$. Reorder the indices so that $j = I$. With the pricing vector $\tilde{\mathbf{c}}$, we have $\beta/B > K/F = \beta_{I-1}/B_I$. Thus, from Theorem 1, $\tilde{x}_I = F - yB_{I-1}$ where $y = (F\hat{c}_I - K)/(\hat{c}_I B_I - \beta_I)$. It is straightforward to show that $c_I B_I - \beta_I = \delta B_{I-1}$ and $FC_I - K = \delta F$. Thus, $y = F/B_{I-1}$ and $\tilde{x}_I = 0$. Thus $R_I(\tilde{\mathbf{c}}) = (\hat{c}_I + \delta)\tilde{x}_I = 0 < R_I(\hat{\mathbf{c}})$, again establishing (26). ♣

The Nash equilibrium in Corollary 1 has several notable proprerties:

1) The price $\hat{c}_i$ does not depend on u_i, the upload rate of server i.
2) All peers have the same price in the Nash equilibrium.
3) For each peer, the Nash price is exactly equal to the price per byte that the client is willing to pay, namely, K/F.

IV. Optimal Selection for Streaming

In this section, we consider streaming of encoded (compressed) audio or video. The delivery constraints are more stringent than for downloading: in order to prevent glitches in playback, the servers must continuously deliver segments of the object on or before their scheduled playout times.

An important parameter for the streaming delivery is the object's playback rate, denoted by r. For an object of size F with playback rate r, the viewing time is $T = F/r$ seconds. Suppose the user at the client begins to view the video at time 0. A fundamental constraint in the streaming problem is that for all times t with $0 \leq t \leq T$, the client must receive the first $r \cdot t$ bytes of the object. We refer to this constraint as the "continuous-playback" constraint. Thus, when selecting the server peers and the object portions to be obtained from each server peer, the client must ensure that this continuous-playback constraint is satisfied. A natural optimization problem is, therefore, to select the peers in order to minimize the total streaming cost subject to continuous playback. To simplify the analysis and to see the forest through the trees, throughout we assume that there is no initial client buffering before rendering, that is, the client begins playback as soon as it begins to receive bytes from any server. Note that for streaming, it is highly desirable that the playback can continue even if some of the server peers fail to provide their services. (In contrast, for downloading a server failure will merely result in a delay in the total download time.) Thus, it is important to explicitly account for failure in the optimal peer selection problem.

As in the previous section, denote $\{1, \ldots, I\}$ for the set of server peers that have a copy of the desired object, and denote. $c_i(b)$ for the cost per unit time when peer i transfers at rate b. To simplify the discussion, we remove the restriction $b_i \leq u_i$; thus, we allow b_i to take any value in $[0, r]$ for all $i = 1, \ldots, I$.

Note that in general, the client must not only select a subset of peer servers, but it must also determine and schedule the specific portions of the file that are downloaded from each selected peer, as well as the download rate from each selected peer. There are two broad approaches that can be taken to solve this problem: **time segmentation** and **rate segmentation**. In time segmentation, the video is partitioned along the time axis in distinct segments, and each server is responsible for streaming only one of the segments in the partition. Typically in the optimal solution for time segmentation, the client will begin downloading segments from various servers before the scheduled playout times of the first bytes of those segments. Thus, client buffering is required. Furthermore, in the optimal solution, the client will typically receive segments from all the selected servers at the beginning of the video and from only one of the selected servers at the end of the video. This means that the client must be able to download (at the beginning of the video) at a rate that is equal to the sum of the server download rates, which will exceed the playback rate. In the rate segmentation approach, each of the selected servers contributes bytes for each of the frames in the video, and at any instant of time the client downloads at the playback rate. In this paper we focus on rate segmentation.

To justify focusing on rate segmentation, we now demonstrate that for convex cost functions, time segmentation is at least as expensive as rate segmentation in terms of download cost. Since time segmentation has the additional drawbacks of requiring both client buffering and higher client download rates, rate segmentation for such cost functions will usually be the better strategy.

Theorem 3: Suppose $c_i(b)$ is convex with respect to b for all $i = 1, \ldots, I$. Then for any solution $\mathcal{S}$ that uses time segmentation, there is a solution $\mathcal{S}'$ using rate

segmentation that has no larger cost.

Proof: Recall that F is the size (in bytes) of the object being streamed, r is the rate of playback, and $T = F/r$ is the rendering time of the object. The convexity of the cost functions implies that for any rate b and any $\lambda \geq 1$, $c_i(\lambda b) \geq \lambda c_i(b)$. In solution $\mathcal{S}$, let x_i be the number of bytes of the object sent by server i and let t_i be the length of time during which server i sends these bytes. Since $t_i \leq T$ for all i,

$$\begin{aligned} \text{time segmentation cost} &= \sum_{i=1}^{I} c_i(b_i) t_i \\ &= \sum_{i=1}^{I} c_i\left(\frac{x_i}{T}\frac{T}{t_i}\right) t_i \\ &\geq \sum_{i=1}^{I} \frac{T}{t_i} c_i\left(\frac{x_i}{T}\right) t_i \\ &= \sum_{i=1}^{I} c_i\left(\frac{x_i}{T}\right) \cdot T. \end{aligned} \tag{27}$$

In solution $\mathcal{S}'$, server i still sends x_i bytes, but these are sent at a rate of $b_i = \frac{x_i r}{F} = \frac{x_i}{T}$ over the entire T seconds. Since $\sum_i x_i = F$, we see that $\sum_i b_i = r$ and thus the rate constraint is satisfied in solution $\mathcal{S}'$. Furthermore,

$$\begin{aligned} \text{rate segmentation cost} &= \sum_{i=1}^{I} c_i(b_i) t_i \\ &= \sum_{i=1}^{I} c_i\left(\frac{x_i}{T}\right) \cdot T. \end{aligned} \tag{28}$$

Comparing (27) and (28), we see that the cost of $\mathcal{S}$ is as least as great as that of $\mathcal{S}'$. ♣

A. Problem Formulation

In the rate-segmentation streaming problem, to ensure continuous playback the client must receive (at least) at rate r at all times. Thus the objective of the streaming problem is to choose the server rates $b_1, \ldots, b_I$ which minimize the total cost $c_1(b_1)T + \cdots + c_I(b_I)T$ subject to the constraint that the total received rate is at least r.

Because the servers in a P2P system are inherently unreliable, we must ensure that the client continues to receive at rate r even when one or more of the selected servers fails. In the ensuing analysis, we allow for up to one server failure (in the next subsection we extend the analysis to multiple server failures). If server j fails during some period of the streaming, then the client receives at rate $\sum_{i \neq j} b_i$. Thus, to ensure that the client continues to receive the video at rate r even when there is one failure, the rates $b_1, \ldots, b_I$ must satisfy

$$\sum_{i \neq j} b_i \geq r, \quad j = 1, \ldots, I.$$

We therefore arrive at the following optimization problem:

$$\begin{aligned} \min \quad & c_1(b_1) + \cdots + c_I(b_I) \\ \text{s.t.} \quad & \sum_{i \neq j} b_i \geq r, \quad j = 1, \ldots, I. \\ & 0 \leq b_i \leq r \quad i = 1, \ldots, I. \end{aligned} \tag{29}$$

Without loss of generality we have included the constraints $b_i \leq r$ for all $i = 1, ..., I$. Indeed, if an optimal solution has $b_j > r$ for some j, we can always reduce b_j to r without violating feasibility and without having to increase the objective value (since $c_j(\cdot)$ is a non-decreasing function).

Before proceeding to solve this streaming problem, we briefly say a few words about implementation. The optimal solution to (29) typically has $b_1 + \cdots + b_K > r$, that is, the aggregate streaming rate (before failure) exceeds the encoded video rate r. In practice, the video would be erasure encoded in a manner that server i sends $x_i = b_i T$ bytes and that client can reconstruct the video if any $I - 1$ of the I streams are received. Although beyond the scope of this paper, it is indeed possible to devise such erasure encoding schemes.

The above problem can be solved by first solving the following problem: for any given y: $0 \leq y \leq r$,

$$\begin{aligned} \min \quad & c_1(b_1) + \cdots + c_I(b_I) \\ \text{s.t.} \quad & \sum_{i} b_i \geq r + y, \\ & 0 \leq b_i \leq y, \quad i = 1, \ldots, I. \end{aligned} \tag{30}$$

Denote $C(y)$ as the corresponding optimal value. Then, solve the problem $\min_{y \leq r} C(y)$.

To show that the two problems are equivalent, let Φ be the set of feasible solutions for (29) and, $\Phi(y)$ be the set of feasible solutions for (30). It is easily seen that if $(b_1, \ldots, b_I)$ belongs to $\Phi(y)$ for some $0 \leq y \leq r$, then $(b_1, \ldots, b_I)$ also belongs to Φ. Furthermore, it is seen that if $(b_1, \ldots, b_I)$ belongs to Φ then it also belongs to $\Phi(y)$, where $y = \max_i\{b_i\} \leq r$. Thus,

$$\Phi = \bigcup_{y \leq r} \Phi(y)$$

and hence minimizing $c_1(b_1) + \cdots + c_I(b_I)$ over Φ can be solved by minimizing $c_1(b_1) + \cdots + c_I(b_I)$ over each $\Phi(y)$ and then taking the minimimum over all $0 \leq y \leq r$.

B. Convex Costs

Suppose $c_i(.)$ is a convex function, for all $i = 1, \ldots, I$. For a given y, the subproblem (30) can be solved in a variety of different ways. For example, if the cost functions are also differentiable, then (30) can be solved by solving $c_i'(b_i^*) = \alpha$ for $i = 1, \ldots, I$, and then

searching through α so that $b_1^* + \cdots + b_I^* = r + y$. We now provide a marginal allocation algorithm, which does not require differentiability:

Marginal Allocation:

- Start with $\mathcal{S} := \{1, ..., I\}$ and $b_i = 0$ for all $i \in \mathcal{S}$.
- In each step identify
$$i^* = \arg\min_{i \in \mathcal{S}} \{c_i(b_i + \Delta) - c_i(b_i)\},$$
where $\Delta > 0$ is a pre-specified small increment (depending on required precision), and reset $b_{i^*} \leftarrow b_{i^*} + \Delta$. Whenever $b_i > y - \Delta$, reset $\mathcal{S} \leftarrow \mathcal{S} - \{i\}$.
- Continue until the constraint $\sum_j b_j \geq r + y$ is satisfied.

Note that the complexity of this algorithm is proportional to $n(r+y)/\Delta$. To determine the best y, we can do a line search on $C'(y) = 0$, for $y \in [\frac{r}{I-1}, r]$. (If $y < \frac{r}{I-1}$, then (30) is infeasible.)

If $C(y)$ is convex in y, then $\min_y C(y)$ is itself greedily solvable: We can start with $y = \frac{r}{n-1}$, increase y by a small increment each time, solve the problem in (30), and stop when $C(y)$ ceases to decrease or $y = r$ is reached.

In this algorithm, when we go from one y value to the next, say, $y + \delta$, we do not have to do the marginal allocation that generates $C(y+\delta)$ from scratch (i.e., starting from all x_j values being zero and $\mathcal{S} := \{1, ..., I\}$). We can start from where the previous round of marginal allocation — the one that generates $C(y)$ — first hits a boundary, i.e., $b_j = y$ for some j, and continue from there. Or, if no b_j has hit the boundary in the previous round, then simply start from where the previous round ends (i.e., continue with the solution generated by the previous round). [Recall, $y \in [\frac{r}{I-1}, r]$. As y increases, the number of b_j values that can hit the boundary in the marginal allocation will decrease. Specifically, when $y \in [\frac{r}{k-1}, \frac{r}{k-2}]$, for $k = 3, ..., I$, the number of b_j values that can hit the boundary cannot exceed k, since we have $ky \geq r + y$.]

The convexity of $C(y)$, in turn, is guaranteed if the $c_i(\cdot)$ are convex functions. To see this, let $(b_i(y))_{i=1}^I$ denote the optimal solution to the problem in (30), and consider two such problems, corresponding to $y = y_1$ and $y = y_2$, respectively. For any $\alpha \in (0, 1)$, we have

$$\begin{aligned}
& \alpha C(y_1) + (1-\alpha) C(y_2) \\
= \quad & \alpha \sum_j c_j(b_j(y_1)) + (1-\alpha) \sum_j c_j(b_j(y_2)) \\
\geq \quad & \sum_j c_j(\alpha b_j(y_1) + (1-\alpha) b_j(y_2)),
\end{aligned}$$

where the inequality follows from the convexity of the c_j. Next, consider a third version of (30), with $y = \alpha y_1 + (1-\alpha) y_2$. It is straightforward to verify that $\alpha b_j(y_1) + (1-\alpha) b_j(y_2)$, $j = 1, ..., I$, is a feasible solution to this problem. Therefore, we have

$$\sum_j c_j(\alpha b_j(y_1) + (1-\alpha) b_j(y_2)) \geq C(y),$$

and hence

$$\alpha C(y_1) + (1-\alpha) C(y_2) \geq C(y) = C(\alpha y_1 + (1-\alpha) y_2).$$

That is, $C(y)$ is a convex function. To summarize, we have:

Theorem 4: Suppose for each $i = 1, ..., I$, $c_i(\cdot)$ is a convex function. Then, the optimal value in (30), $C(y)$, is convex in y. In this case, the streaming problem in (29) is greedily solvable: In each step increase y by a small increment (starting from $y = \frac{r}{I-1}$), apply the marginal allocation algorithm to generate $C(y)$, and stop when $C(y)$ ceases to decrease or $y = r$ is reached.

C. Concave Costs

Now, suppose the costs $c_i(\cdot)$, $i = 1, \ldots, I$, are *concave* (instead of convex) functions. The equivalence of (29) and (30) continues to hold. However, there are two changes:

(i) The marginal allocation will not generate the optimal solution to (30).
(ii) $C(y)$ is no longer a convex function. (Neither is it a concave function, for that matter.)

The reason for (ii) is evident from examining the earlier argument that established the convexity of $C(y)$. The reason for (i) is that the marginal allocation typically generates a solution that is at the *interior* of the feasible region of the problem in (30), which is a polytope. As such the solution can be expressed as a convex combination of the extreme points (vertices) of the feasible region. (This follows from the well-known Carathéodory Theorem.) Due to the concavity, the objective value corresponding to this solution will dominate (i.e., be no smaller than) the convex combination of the objective values corresponding to the extreme points, and hence dominate the smallest of these values.

Therefore, to solve the optimization problem in (30) in this case, we only need to consider the extreme points of its feasible region.

First, observe that the constraint $\sum_j b_j \geq r+y$ must be binding at optimality; otherwise, we could always reduce some of the b_j values and thereby improve the objective value while maintaining feasibility.

Second, we divide the range $y \in [\frac{r}{I-1}, r]$ into segments $[\frac{r}{k-1}, \frac{r}{k-2}]$, for $k = 3, ..., I$. Then, with the constraint $\sum_j b_j \geq r + y$ binding, it is clear that the number of b_j values at the boundary, $b_j = y$, cannot

exceed k when $y \in [\frac{r}{k-1}, \frac{r}{k-2}]$. (Here we assume $I \geq 3$; the case of $I = 2$ is trivial: the optimal solution is $b_1 = b_2 = y = r$.)

Consequently, when $y \in [\frac{r}{k-1}, \frac{r}{k-2}]$, we only need to consider extreme points that take the following form: $b_j = y$ for $k-1$ distinct indices j, $b_\ell = r-(k-2)y$ for another distinct index ℓ, and $b_i = 0$ for all the remaining i's. For each such extreme point, the b_i's sum up to

$$(k-1)y + r - (k-2)y = r + y,$$

making the constraint $\sum_j b_j \geq r + y$ binding.

Specifically, with y a given value in the interval $[\frac{r}{k-1}, \frac{r}{k-2}]$, without loss of generality, suppose

$$c_1(y) \leq c_2(y) \leq \cdots \leq c_I(y), \tag{31}$$

$$c_{i_1}(r-(k-2)y) \leq \cdots \leq c_{i_I}(r-(k-2)y), \tag{32}$$

where $(i_1, \ldots, i_I)$ is a permutation of $(1, \ldots, I)$. Denote

$$\alpha_i := c_i(y), \qquad \beta_i := c_i(r-(k-2)y).$$

Clearly, we only need to consider no more than k such extreme points, which we shall refer to as *non-dominant*. Each of the other extreme points is *dominant*, in the sense that it's objective value $C(y)$ will dominate (i.e., be at least as large as) one of the non-dominant points.

Let α_{-i} denote the vector $(\alpha_1, \alpha_2, \ldots, \alpha_k)$ without the component α_i for some $i = 1, ..., k$. Then, specifically, these (possibly) non-dominant points are

$$(\alpha_{-1};\beta_1),\ (\alpha_{-2};\beta_2), \ldots, (\alpha_{-k}, \beta_k) \tag{33}$$

$$(\alpha_{-k};\beta_{k+1}),\ (\alpha_{-2};\beta_{k+2}), \ldots, (\alpha_{-k}, \beta_I). \tag{34}$$

As before, let $C(y)$ denote the optimal objective value of (30). Then the $C(y)$ value corresponding to an extreme point is the sum of $k-1$ values of α_i (for $k-1$ distinct i's) and one value of β_j for a j that is distinct from all the i's.

Theorem 5: Suppose that for all $i = 1, ..., I$, $c_i(\cdot)$ is a concave function. Then, the optimal solution to (30), for $y \in [\frac{r}{k-1}, \frac{r}{k-2}]$, $k = 3, ..., I$, is generated by taking the minimum of the objective values, $C(y)$, corresponding to the I points in (33) and (34). The solution to the streaming problem in (29) is then obtained by applying a line search to $\min_{y \in [r/(I-1), r]} C(y)$.

We may be able to further eliminate some of the non-dominant points. Let us illustrate this through an example. Consider $n = 5$. Suppose the permutation in (31) is $(i_1, \ldots, i_5) = (2,4,1,3,5)$. Consider $k = 4$. Then, the following four points correspond to the ones in (33):

$$(1,3,4;2),\ (1,2,3;4),\ (2,3,4;1),\ (1,2,4;3). \tag{35}$$

A closer examination tells us, however, that the last two of the four points in (35) are, in fact, dominant: they dominate, respectively, $(1,3,4;2)$ and $(1,2,3;4)$. Hence, in this case, there are only 2 non-dominant points. Specifically, any point that involves a β_{i_ℓ} such that i_ℓ violates the *increasing* order in the permutation $(i_1, i_2, \ldots, i_n)$ cannot be a dominant point. This is the case for 1 and 3 in the permutation $(2,4,1,3,5)$ in the above example.

The full details of this example can be worked out as follows:

- $y \in [\frac{r}{4}, \frac{r}{3})$, $k = 5$: the non-dominant points are:

$$(1,3,4,5;2),\ (1,2,3,5;4),\ (1,2,3,4;5).$$

- $y \in [\frac{r}{3}, \frac{r}{2})$, $k = 4$: the non-dominant points are:

$$(1,3,4;2),\ (1,2,3;4).$$

- $y \in [\frac{r}{2}, r]$, $k = 3$: the non-dominant points are:

$$(1,3;2),\ (1,2;4).$$

In each case, the optimal solution (to (30)) is obtained by comparing the $C(y)$ values of the non-dominant points and picking the one corresponding to the smallest $C(y)$ value.

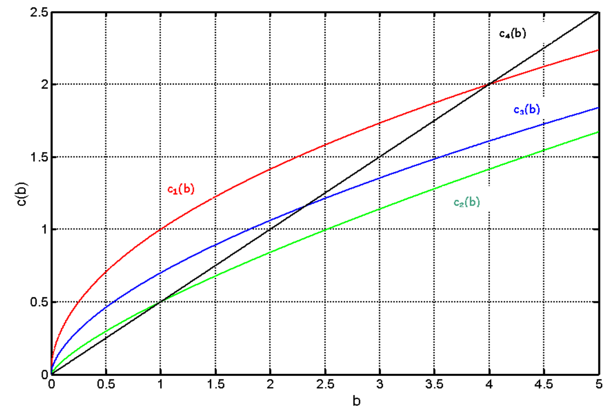
Finally, a comment on the line search mentioned in the above proposition. Suppose we divide the interval $[\frac{r}{I-1}, r]$ into equal segments, each of length Δ. Let $N := \frac{r(I-2)}{(I-1)\Delta}$ denote the number of such segments. When Δ is sufficiently small, we can safely assume that the ordering in (31) does not change over any given segment. This means that for any y that belongs to a given segment, the optimal value $C(y)$ is determined by a single non-dominant point $(\alpha_{-i_\ell}; \beta_{i_\ell})$. That is,

$$C(y) = \sum_{i \leq k, i \neq i_\ell} c_i(y) + c_{i_\ell}(r-(k-2)y).$$

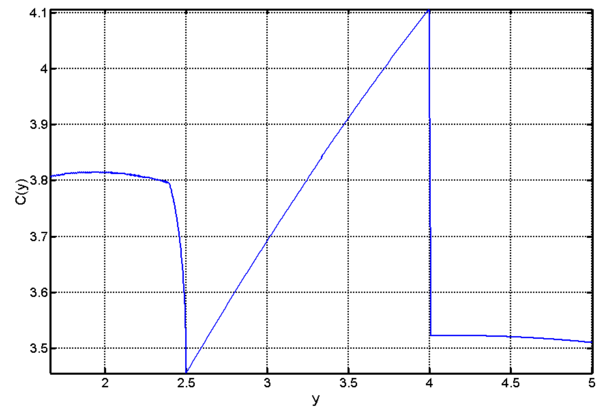
Hence, $C(y)$ is a concave function over this segment, since the c_i and c_{i_ℓ} are all concave functions. Consequently, the minimum of $C(y)$ can only be attained at the two end points of the segment. Therefore, the line search to minimize $C(y)$ amounts to evaluating N values of $C(y)$ and picking the smallest one. This way, the streaming problem is solved by an algorithm of $O(NI)$ time.

Example: We now completely work out the optimal bandwidth profile for the problem in (29) in the case of concave cost functions. Let $r = 5$ and

$$\begin{aligned} c_1(b) &= \sqrt{b} \\ c_2(b) &= 0.5b^{\frac{3}{4}} \\ c_3(b) &= 0.7b^{\frac{3}{5}} \\ c_4(b) &= 0.5b \\ r &= 5 \end{aligned}$$



(a) Concave $c_i(b)$ functions



(b) Objective function C(y)

Fig. 1. (a) shows the various concave cost functions for $0 \leq b \leq 5$. (b) is the corresponding plot for the objective function $C(y)$ for $5/3 \leq y \leq 5$.

In Figure 1(a) we plot the four concave cost functions $c_i(b)$ for $0 \leq b \leq r$.

Let $C(y)$ be defined as in (30). Then the minimal cost (29) is given by

$$C_{opt} = \min_{y \in [\frac{r}{I-1}, r]} C(y).$$

We use the solution procedure described in Section IV-C to evaluate $C(y)$ for $y \in [\frac{r}{I-1}, r]$. This result is plotted in Figure 1(b). As can be seen from Figure 1(b), $C(y)$ is neither a concave nor a convex function, implying that a line search has to be done for finding C_{opt}. It can also be seen that C_{opt} is achieved at $y = 2.5$. The corresponding optimal bandwidth profile is given by $b_1 = 0$, $b_2 = 2.5$, $b_3 = 2.5$, $b_4 = 2.5$. The corresponding cost of downloading is $C_{opt} = C(2.5) = 3.4557$.

D. Multiple Unavailable Servers

The above approach extends readily to the general case when multiple servers can become unavailable. Let f be the maximum number of servers that can be unavailable, where $1 \leq f \leq I - 1$. In this case, the problem formulation in (29) becomes,

$$\begin{aligned} \min \quad & c_1(b_1) + \cdots + c_I(b_I) \qquad (36) \\ \text{s.t.} \quad & \sum_{j \neq i_1, \ldots, i_f} b_j \geq r, \quad i_1, \ldots, i_f = 1, \ldots, I. \\ & 0 \leq b_i \leq r, \quad i = 1, \ldots, I. \end{aligned}$$

In the above optimization problem, the notation $i_1, \ldots, i_f = 1, \ldots, I$ means one such constraint for every subset of f elements from $\{1, \ldots, I\}$.

We claim that the equivalent problem, for $0 \leq y \leq r$, becomes

$$\begin{aligned} \min \quad & c_1(b_1) + \cdots + c_I(b_I) \qquad (37) \\ \text{s.t.} \quad & \sum_j b_j \geq r + fy, \\ & 0 \leq b_i \leq y \leq r, \quad i = 1, \ldots, I. \end{aligned}$$

The key observation here is that the optimal solution to (36) must satisfy the property that the largest f values of b_i are all equal. Specifically, without loss of generality, suppose

$$b_1 \geq b_2 \cdots \geq b_f \geq b_{f+1} \geq \cdots \geq b_I \qquad (38)$$

is an optimal solution to (36). Then, we must have $b_1 = b_2 = \cdots = b_f$. Consider any $e < f$, and hence $b_e \geq b_f$. We can reduce b_e to b_f and still do no worse on the objective value (as the c_i's are non-decreasing functions), while maintaining feasibility. To see this, consider the constraint

$$b_e + b_{f+2} + \cdots + b_I \geq r. \qquad (39)$$

Reducing b_e to b_f turns the above into

$$b_f + b_{f+2} + \cdots + b_I \geq r, \qquad (40)$$

which certainly holds as it is one of the constraints involving b_f. Furthermore, any other constraint that involves b_e has a left hand side that is at least as large as the left hand side of (39) – due to the ordering in (38). Hence, it will also remain feasible when b_e is reduced to b_f, since its left hand side, after the reduction, will still dominate the left hand side of (40).

Therefore, we can solve the following equivalent problem:

$$\begin{aligned} \min \quad & c_1(b_1) + \cdots + c_I(b_I) \qquad (41) \\ \text{s.t.} \quad & \sum_j b_j \geq r + fy, \\ & 0 \leq b_i \leq y \leq r, \quad i = 1, \ldots, I. \end{aligned}$$

This equivalence is similarly argued as before. First, any feasible solution to (36) is a feasible to (41) with y set at the largest b_i value. (Note, as before, the optimal solution to (36) must satisfy $b_i \le r$ for all i.) Second, given a feasible solution to (41), we must have

$$\begin{aligned}\sum_{j \neq i_1,\ldots,i_f} b_j &\geq r + fy - b_{i_1} - \cdots - b_{i_f} \\ &\geq r + fy - fy = r,\end{aligned}$$

i.e., it satisfies the constraint in (36) as well.

Hence, we can solve the equivalent problem in (41) as in the case of $f = 1$, for both convex and concave cost functions. It is easy to see that, for both types of cost functions, we must have $y \ge \frac{r}{I-f}$; otherwise, the problem is infeasible. For concave costs, we will consider the intervals $y \in [\frac{r}{k-f}, \frac{r}{k-f-1}]$, for $k = f + 2, \ldots, I$ (assuming $f \le I - 2$, the case of $f = I - 1$ being trivial). For the k-th interval, the non-dominant points are $k - 1$ distinct b_i values set at y, and another distinct b_j set at $r - (k - f - 1)y$, with the total being

$$(k-1)y + r - (k-f-1)y = r + fy;$$

and $0 \le r - (k - f - 1)y \le y$ (i.e., b_j is feasible), or $y \in [\frac{r}{k-f}, \frac{r}{k-f-1}]$.

V. Summary and Conclusion

We envision a free-market resource economy in which peers buy and sell resources directly from each other. In the context of a P2P resource market, we considered the problem of optimal peer and rate selection. To our knowledge, this is the first work that considers optimal peer selection in a P2P resource market.

Throughout this paper we allowed for a natural pricing function of the form $c_i(b_i) \cdot t_i$, where i indexes the peer server, b_i is the rate at which the server transmits bytes to the client, and t_i is the duration of the transfer. We considered optimal peer selection for two broad classes of problems: downloading and for streaming. For both classes of problems we considered both convex and concave cost functions.

For the downloading problem with concave cost functions, we provided an explicit solution to the problem, whereby all selected peers transmit at their maximum rate u_i. For convex cost functions we showed how the problem can be easily solved with marginal analysis, and that for many natural convex cost functions, all I servers are selected, with none of the servers transmitting at their maximal rates. We also found that in the Nash equilibrium, each server sets its cost to the price per byte that the client is willing to pay.

For the streaming problem, we showed that for problems of practical interest, rate segmentation can always do as well as time segmentation. We then focused on rate segmentation. We first considered the scenario in which at most one server peer can fail. We then extended the results to the scenario in which up to f server peers can fail, for any value of f. We again analyzed both convex and concave cases. We found that each case requires a different methodology, although both cases are quite tractable.

The contribution of our work is the development of theoretical methodologies for these types of peer selection problems. We have formulated and solved a rich array of optimal downloading and streaming problems. The techniques presented here should be helpful in solving alternative formulations of peer selection problems.

Acknowledgment: We want to thank David A. Turner of California State University San Bernardino for his insightful comments.

References

[1] E. Adar and B.A. Huberman, "*Free Riding on Gnutella*," First Monday, 5(10), October 2000. http://www.firstmonday.dk/issues/issue5_10/adar/

[2] D. Qiu and N.B. Shroff, "*A Predictive Flow Control Scheme for Efficient Network Utilization and QoS*," IEEE/ACM Transactions on Networking, February 2004.

[3] KaZaA Homepage, http://www.kazaa.com

[4] Overnet, http://www.edonkey2000.com

[5] J. Liang, R. Kumar, K.W. Ross, "*Understanding KaZaA*," submitted, 2004.

[6] S. Saroiu, P. K. Gummadi, and S. D. Gribble, "*A measurement study of peer-to-peer file sharing systems*," Proceedings of Multimedia Computing and Networking, Jan 2002.

[7] D.A. Turner and K.W. Ross, "*A LightWeight Currency Paradigm for the P2P Resource Market*" Seventh International Conference on Electronic Commerce Research, Dallas, TX, Jun 2004.

[8] D.A. Turner and K.W. Ross, "*The Lightweight Currency Protocol*," Internet Draft, September 2003.

[9] S. D. Kamvar, M. T. Schlosser, H. Garcia-Molina, "*The EigenTrust Algorithm for Reputation Management in P2P Networks*," In Proceedings of the Twelfth International World Wide Web Conference, 2003.

[10] R. Dingledine, N. Mathewson, P. Syverson, "*Reputation in P2P Anonymity Systems*," Workshop on Economics of Peer-to-Peer Systems, 2003.

[11] M.R. Garey and D.S. Johnson, "*Computers and Intractability: A guide to the Theory of NP-Completeness*," W.H. Freeman, 1979.

[12] R.E. Barlow and F. Proschan, "*Statistical Theory of Reliability and Life Testing*," Holt, Reinhart and Winston, 1975.