Using bandwidth sharing to fairly overcome channel asymmetry

Sachin K Agarwal
Deutsche Telekom AG, Laboratories
Berlin, Germany-10587
Email: sachin.agarwal@telekom.de

Moshe Laifenfeld, Ari Trachtenberg and Murat Alanyali
Boston University
Boston, USA-02215
Email: moshel, trachten, alanyali@bu.edu

Abstract— **We propose a peer-to-peer architecture designed to overcome asymmetries in upload/download speeds that are typical in end-user dialup, broadband and cellular wireless internet connections. Our approach allows users at remote locations to access information stored on their home computers at rates often exceeding their home connection's upload capacity. The key to this approach is to share file data when communications are idle using random linear coding, so that, when needed, an end-user can download a file from several sources at a higher data rate than his home computer's upload capacity. We prove that our proposed system is asymptotically fair, in that (even malicious) users are proportionally assigned idle bandwidth depending on how much bandwidth they contribute, and that there is a natural incentive to join and cooperate fairly in the system. In addition, our approach provides cryptographic security and geographic data robustness to the participating peers.**

I. INTRODUCTION AND MOTIVATION

Many users connect to the internet through asymmetric links, in which the upload capacities are much smaller than download capacities. Internet service Providers (ISPs) employ this asymmetric design based on the premise that casual internet use mostly involves downloading from a relatively small number of *content providers*.

Recently the 'mostly download' profile of users has started to change. Users now commonly have access to devices like digital video cameras, high resolution scanners, and high capacity sound recorders that capture large volumes of digital data. This change in users' access profile makes upload speed a bottleneck for typical remote access. Thus, if a user remotely accesses data stored on a home computer, such as a song or video, his access rate is limited by *both* his home's upload capacity and his remote location's download capacity.

This work attempts to correct for such channel asymmetries by filling a high bandwidth download pipe through the aggregation of multiple idle lower bandwidth upload pipes. Our proposed approach has the following features:

- **Fairness** Unallocated bandwidth is re-distributed in proportion to the bandwidth contributed by system peers.
- **Incentive** There is a natural incentive for peers to participate and cooperate with others in the system.
- **Distributed operation** Only local information is needed (i.e. no control information needs to be exchanged).
- **Robustness** Data is available from many sources.

In addition, it is not too difficult to add security to the model.

In effect, our approach permits users to bypass the 'bandwidth: use it or lose it' service model offered by commercial ISPs, and instead maintain 'credit' for their contributions within the system as a whole. With hard-disk storage costing under a dollar per gigabyte, the benefits enumerated above quickly surpass the cost of caching other users' data.

The rest of this paper is organized as follows. In Section II we mention some of the related work in the field and contrast it with our approach. Thereafter, we formally introduce the details of our proposed bandwidth sharing method in Section III. In Section IV we analytically prove the fairness of our system and show that it provides a natural incentive for peer contributions. We simulate various aspects of our system in Section V to demonstrate its features, including specific cases where malicious peers attempt to take unfair advantage of the system. We also demonstrate the real-time efficiency of random linear coding for this application. Finally Section VI concludes our results and suggests directions for future work.

II. RELATED WORK

Peer-to-peer (P2P) systems are typically used to distribute content on the Internet, and it is estimated that a major portion of the bandwidth available on consumer ISP networks carries P2P content [13]. P2P services make scalable content distribution possible by utilizing peers' upload bandwidth to service other peers' download requests. It has been shown through analysis [7, 14–16], simulations and measurements [4, 8, 17] that the P2P content delivery model scales gracefully with user demands for heterogeneous P2P networks. In the remainder of this section we describe some of the literature that is relevant to the different aspects of our proposed system, ending with a brief explanation of the novelty of our approach.

a) Content distribution: Services like BitTorrent [11] assume some out-of-band mechanisms to locate content for users to download. In particular, BitTorrent content location information is published through web sites. Various distributed hash table (DHT) based mechanisms such as Tapestry [9] have been developed to provide the important functionality of locating shared content on P2P networks. Much recent work in P2P networks concentrates on mitigating non-cooperative behavior of peers by adding incentive schemes. Due to the scalability issues, most of these schemes are distributed, and require only local information readily available to each peer. As with our scheme, the most common schemes are based on *Barter economy* where peers offer their bandwidth to others according to the amount of bandwidth allocated to them [18, 19]. Although adding incentive scheme may increase the cooperation among reasonable users, namely users which try to optimize their resources, it usually has no guarantee against malicious peers, and additional measures are required to reduce their effect on the network.

The idea of sharing disk-space for data backup and downloads is not new. For example, the Folder-share system [10] lets users share their documents with other users. In this system, however, users only download a file from one peer, thus limiting their download speed to the upload speed of the peer. In addition the system assumes that peers do not cheat others when it comes to offering bandwidth.

b) Coding: The Oceanstore project [12], provides a large data storage solution using erasure codes. Erasure code type approaches such as digital fountain [3] have been proposed for large scale content distribution. Random linear coding [2, 5] has been used for achieving a network coding [1] min-cut bound on multicast in networks. The authors in [5] proposed random linear coding as a way to avoid the “coupon collector’s problem” in a P2P storage system. Their application considers the case when *parts* of the same file are encoded and kept on separate hosts and rebuilt. While our system can also operate in this fragmented storage mode, the emphasis is on fairness and the ability to beat the upload link bottleneck.

c) Analysis: Much of the work on P2P systems characterizes fairness and incentives for peers to cooperate by simulations, measurements and experiments of P2P systems rather than actual analysis, probably because of the complexities arising from the size, chaotic nature and heterogeneous conditions that characterize real systems (e.g. [18, 19]). Nevertheless, there has been some recent queuing theoretic analytical work [16] that uses a queuing based approach to study the scalability and resilience to freeloaders in P2P systems. A game theoretic approach was also applied to a related problem, the problem of parallel downloading, *i.e.,* downloading a large file from several servers in parallel. In [20] this problem is analyzed using non-cooperative game theoretic tools. However, this approach cannot capture the effects of malicious users.

A. Our approach

Unlike many existing P2P systems that are built for discovering and disseminating popular content, our system attempts to share *unused bandwidth* among system subscribers. Users who contribute bandwidth to the system are rewarded with higher instantaneous bandwidth availability when they need it. In all cases, users are (asymptotically) assured that bandwidth they share with the network will be returned to them.

Our system also differs from typical P2P systems in that it is used by remote users, thus differentiating between users and network peers. More precisely, when a user u wishes to access her content (which has been distributed among the network peers off-line), she downloads content from *multiple* peers in the network, (possibly) including her own home computer. This subtle difference means that our system no longer needs the ‘non-dominant’ condition in [14, 15], *i.e.,* that the upload capacity of one peer is necessarily smaller than the sum of upload capacities of all other peers. This, in turn, means that our system does not require a symmetric instantaneous ‘tit-for-tat’ requirement to guarantee fairness (*i.e.,* the system intrinsically evens out contributions asymptotically).

III. ALGORITHMIC DETAILS

We now describe some algorithmic details of our system. Throughout this description we assume that each user corresponds to one peer on the network (*e.g.,* his home computer).

A. Initialization

Our system is initialized with each peer disseminating its data among other peers using a random linear coding approach motivated by the work in [5] (which applies coding to P2P storage applications). For the purposes of our analysis, we assume that each peer has an infinite amount of disk space so that there is no utility cost for caching another peer’s data.

More precisely, consider a long file X consisting of b bits to be disseminated in an n-peer network.In the standard random linear coding approach, X is split into k chunks $\{X_1, X_2, \ldots, X_k\}$ with each chunk mathematically represented as an m-element vector with components in a finite field $\mathbb{F}_q$ of size $q = 2^p$ for some p, *i.e.,* $X_j \in \mathbb{F}_{2^p}^m$ with $mpk = b$. This formulation effectively translates file X into k vector chunks. These vectors are coded into nk message vectors $\{Y_1, Y_2, \ldots, Y_{nk}\}$ whose i-th component Y_i is

$$Y_i = \sum_{j=1}^{k} \beta_{ij} \cdot X_j, \; i = 1 \ldots nk, \qquad (1)$$

where each $\beta_{ij} \in \mathbb{F}_q$ is randomly chosen. By choosing β’s randomly, we insure that $\beta_i = [\beta_{ij}, j = 1 \ldots k]$ are almost surely linearly independent [6].

Our encoding is similar to the encoding proposed in [2] for network coding-based multicast [1], with two important technical differences: (1) rather than transmitting β’s as message headers, we use them as a secret key; (2) rather than having peers transferring linear combinations of their information to others on the network, peers transmit exactly what was uploaded to their storage area. The first difference guarantees that no peer can decode a message stored on its system unless it correctly guesses the k-tuple β_i (and *knows* that the guess is correct). The second difference ensures that peers do not need to perform any computation when messages are requested from them; they simply forward what they have stored.

To complete the initialization phase, each plain text message-id i is appended to the Y_i’s of Equation (1) and these encoded messages are then uploaded to the n participating peers (up to k messages per peer) where they are stored.

B. Accessing data

To decode a file, a user requests a total of k messages (*i.e.,* Y_i's) from multiple peers, preferably in parallel, and multiplies this by the inverse of the appropriate square sub-matrix of the coefficient matrix $\beta = [\beta_{ij}]$. The rows of the coefficient sub-matrix are determined from the message-id's appended to each of the k received messages Y_i (as in Section III-A).

IV. FAIRNESS AND INCENTIVE TO COOPERATE

In this section we introduce a technical analysis of our system. In particular we focus on the *incentive* and *fairness* aspects for peers comprising the network. A user has an incentive to join the system since he is guaranteed to receive on the average at least the amount of bandwidth he would get if he had operated with his corresponding peer in isolation, potentially reaching higher average and instantaneous bandwidths. A user has incentive to cooperate since the amount of additional allocated bandwidth it receives from the system is proportional to the amount of bandwidth it allocates to the network. We will further argue that the system is resilient to adversarial users, that is, the incentive remains for any strategy adopted by one or several users. We will also show that the system is *fair* in the sense that the users are allocated with *free* bandwidth according to how much bandwidth they share with the network (relative to others).

A. Technical definitions and motivation

Our formal model considers n peers sharing their upload bandwidths in a time-slotted fashion. As indicated in previous sections, we will refer to the remote owner of a peer, say peer i, as user i. We denote the individual upload bandwidth of peer i with μ_i, and assume a random demand pattern for each user. Specifically, we assume that user i requests bandwidth for download at time-slot t with probability γ_i, independently of other users and of the history of the system at slot t. Let $I_i(t)$ be the binary random variable that is 1 if and only if user i requests bandwidth at time t. The independence assumption above implies that the variables $I_i = (I_i(t) : t \geq 0)$ are i.i.d. with $Pr(I_i(t) = 1) = \gamma_i$, and that the sequences $(I_i : i = 1, 2, \cdots, n)$ are independent.

Note that if user i operates *in isolation* and downloads only from its own peer, then its download speed is limited by μ_i per request, which corresponds to an average capacity utilization of $\mu_i\gamma_i$ per time slot in the long term. We next introduce our bandwidth allocation scheme and study its properties in the next subsections. Towards this end, let $\mu_{ij}(t)$ denote the upload bandwidth that peer i devotes to user j at slot t. This non-negative quantity is non-zero only if $I_j(t) = 1$, that is, user j has a request at that time. The proposed bandwidth allocation scheme is given by

$$\mu_{ij}(t) = \frac{\mu_i}{\sum_{l=1}^{n} I_l(t) \sum_{k=0}^{t-1} \mu_{li}(k)} I_j(t) \sum_{k=0}^{t-1} \mu_{ji}(k), \tag{2}$$

with some arbitrary small positive initial values for $\mu_{ji}(0)$. Note that the proposed scheme relies only on local measurements at each peer, and it does not require any transfer of information among the peers or users, which is prone to attack.

As a motivation, we first consider a different allocation scheme, which is similar to the *global proportional fairness* scheme of [14]. In this scheme the bandwidth allocated to peer i is proportional to his contribution among all actively requesting peers at that time. Although it can be shown to be asymptotically pairwise fair in the number of peers, *i.e.,* $\mu_{ij}(t)\gamma_i \approx \mu_{ji}(t)\gamma_j$, it has major drawbacks, primarily because it gives incentive to peers to lie about their contribution. The problem could be mitigated through accurate measurement of peer contributions, but not without implementational difficulties and allowing secondary attacks. This leads to the proposed allocation rule of (2), which is studied hereafter.

B. Incentive and fairness

Let $\overline{\mu_{ij}}(t)$ denote the time-average of the bandwidth that user j receives from peer i by time t. That is, $\overline{\mu_{ij}}(t) = t^{-1} \sum_{k=0}^{t-1} \mu_{ij}(k)$. The random sequence $(\{\mu_{ij}(t), \overline{\mu_{ij}}(t),\ i, j = 1, 2, \cdots, n\} : t \geq 1)$ is Markovian. Let us assume without rigorous justification that this chain is ergodic; in particular that $\overline{\mu_{ij}}(t)$ converges to the equilibrium expectation of $\mu_{ij}(t)$. Let $\overline{\mu_{ij}}$ denote the limit and define $\overline{\mu_j} = \sum_{i=1}^{n} \overline{\mu_{ij}}$. Recall that an average upload bandwidth of peer i operating in isolation is $\gamma_i\mu_i$, and the *unallocated* or *free* average bandwidth is $(1 - \gamma_i)\mu_i$.

Theorem 1 *Allocation rule* (2) *guarantees that, asymptotically, the average download bandwidth of user i is not only its average bandwidth in isolation, but also fractional portions of the free bandwidth of other users in the network. That is,*

$$\overline{\mu_i} \geq \gamma_i\mu_i + \gamma_i \sum_{l \neq i} \alpha_{il}(1 - \gamma_l)\mu_l.$$

where the fractional portions are proportional to the amount of the bandwidth user i shares with the network, i.e.,

$$\alpha_{il} = \frac{\overline{\mu_{il}}}{\overline{\mu_{il}} + \sum_{j \neq i} \gamma_j \overline{\mu_{jl}}}.$$

Proof sketch: Multiplying and dividing by t in (2) gives:

$$\mu_{ij}(t) = \mu_i \frac{I_j(t)\overline{\mu_{ji}}(t)}{\overline{\mu_{ji}}(t) + \sum_{l \neq j} I_l(t)\overline{\mu_{li}}(t)}. \tag{3}$$

For large t, the allocation rule is well approximated by

$$\mu_{ij}(t) = \mu_i \frac{I_j(t)\overline{\mu_{ji}}}{\overline{\mu_{ji}} + \sum_{l \neq j} I_l(t)\overline{\mu_{li}}}. \tag{4}$$

In this form the denominator and numerator are independent; taking the expectation and applying Jensen's inequality gives

$$\begin{aligned} \overline{\mu_{ij}} &= \mu_i\gamma_j\overline{\mu_{ji}}\mathbf{E}\left(\frac{1}{\overline{\mu_{ji}} + \sum_{l \neq j} I_l(t)\overline{\mu_{li}}}\right) \\ &\geq \frac{\mu_i\gamma_j\overline{\mu_{ji}}}{\overline{\mu_{ji}} + \sum_{l \neq j} \gamma_l\overline{\mu_{li}}}. \end{aligned} \tag{5}$$

Taking (5) when $i = j$, and after some manipulations we get

$$\overline{\mu_{ii}} \geq \gamma_i \mu_i - \sum_{l \neq i} \gamma_l \overline{\mu_{li}} \tag{6}$$

By adding $\sum_{l \neq i} \overline{\mu_{li}}$ to both sides,

$$\overline{\mu_i} = \overline{\mu_{ii}} + \sum_{l \neq i} \overline{\mu_{li}} \geq \gamma_i \mu_i + \sum_{l \neq i} (1 - \gamma_l) \overline{\mu_{li}}. \tag{7}$$

To complete the proof use (5) to substitute for $\overline{\mu_{li}}$ ∎

Theorem 1 provides two important features of the network. First it shows that each user has an incentive to share its corresponding peer's bandwidth. The larger the bandwidth he shares relative to the amount others share - the larger the portion of free bandwidth that he receives. This can also be interpreted in terms of *fairness, i.e.,* . someone who has a larger portion in the upload bandwidth of user i will benefit from a larger portion of the free bandwidth of this user.

The second feature of the network, is the incentive to join the network. Theorem 1 guarantees that all users receive at least the amount of bandwidth that they would have received if they hadn't joint the network. Note that this guarantee for a specific user, say user i, holds under the mere assumption that this user requests downloads independently of the remaining users. Consider, for example, the case where other users form a coalition in order to manipulate the bandwidth of peer i. No matter what strategy they use, the guarantee of Theorem 1 implies that another user can only gain by joining the network.

It can be further shown that in a stationary regime when all peers are saturated with upload requests, a strong *pair-wise fairness* holds.

Corollary 1 *In the saturated regime, when* $\gamma_i \rightarrow 1 \quad \forall i$, *the network of allocation* (2) *guarantees pair-wise fairness, i.e*

$$\overline{\mu_{ji}} = \overline{\mu_{ij}} \quad \forall i, j.$$

A similar result in [15] requires the non-dominant condition that we do not. Note also that the peer-wise fairness property does not hold in general. Individual users can enjoy other peer's free upload bandwidth to increase their total average upload bandwidth even beyond their own single peer-user isolated bandwidth.

V. SIMULATIONS

In Section V-A we demonstrate the fairness and incentive results of Section IV. We implemented a discrete time simulator of the p2p system described in Section III for the purposes of confirming the claims made in this paper. Each peer reallocated their upload bandwidths once per second, and our graphs were smoothed with a running average of 10 seconds. In another set of experiments (Section V-B), we show the efficiency of random linear coding.

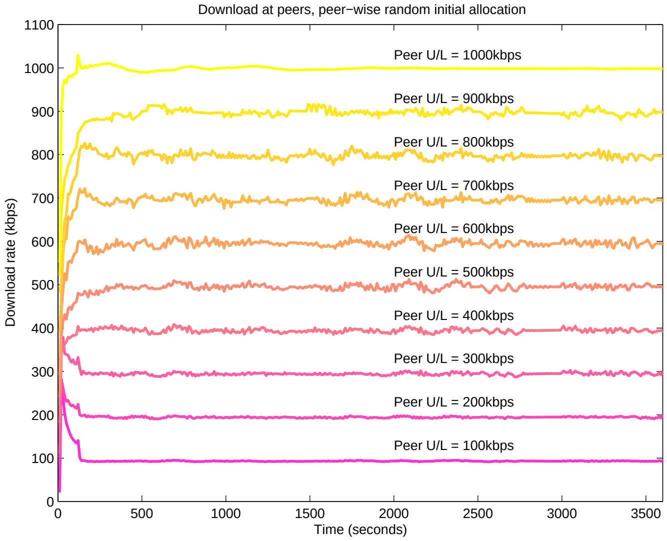

Fig. 1. Ten users request a large file from the system. Their download rate converges to the upload rate (U/L) of their corresponding peers.

A. Fairness and benefit

In our first experiment, ten users request large files at the same time, but their corresponding peers have different upload capacities ranging from 100kbps to 1000kbps. The download rate provided by the system is plotted in Figure 1 as a function of time. We can see that system initially goes through a period where bandwidth allocation among peers looks random, but that it quickly converges to rates that are commensurate with the upload capacities of the various peers.

Our next graph (Figure 2) demonstrates the benefits of the proposed system to all collaborating peers commensurate with Theorem 1. For these graphs, we simulated a three peer network comprising users that have encoded and distributed home videos to all three peers. The users stream their videos to a remote computer for 12 randomly chosen hours in a day (meaning they request bandwidth from the shared pool of upload bandwidth during these intervals).

Peers 2 and 3 are available to upload to other peers all through a 24 hour period but peer 1 only starts to contribute to the system after the first three hours. Two interesting artifacts occur: first, we notice that user 1 is still able to get some service from the network in the first hour because user 2 has not yet requested anything from the network and so peer 2 is splitting its bandwidth between users 0 and 1, being oblivious to peer 1 not contributing (this is corrected in the 2-3 hour time slot). In the 3-4-hour time slot, user 1 is penalized for his non-contribution to the system, though this penalty decays by time $t = 4$ hours as all users start benefiting from the contributed bandwidth.

B. Performance of random linear codes

We next demonstrate the efficiency of the random linear coding component of our system (and the corresponding decoding complexity). In order to establish the speed of random

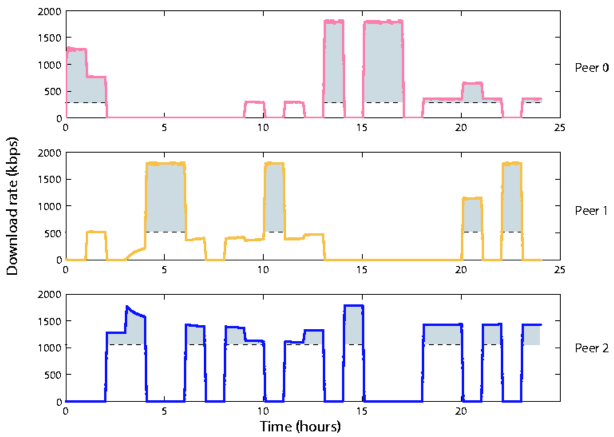

Fig. 2. Three peer network, $\mu_0 = 256kbps$, $\mu_1 = 512kbps$, $\mu_2 = 1024kbps$. Peer 1 starts contributing after the first 3 hours. The shaded regions show the gains of using the proposed approach; duty cycles indicate when the users are requesting bandwidth.

linear coding and infer the maximum throughput when the bottleneck is the decoding computation on the user's computer, we have built a simple encoder/decoder and tested it on 1MB of data for various values of message size m, finite field size q and corresponding number of messages k into which the 1MB of data is split and encoded according to Equation (1). We believe that this study will also aid in making design decisions in related applications such as network coding based multicast.

Our experiments were performed on a Pentium 4 dual processor workstation running the Linux operating system, although the code only ran on one processor. The encoding and decoding operations are essentially the same, the latter using the inverse of the coefficient matrix in Equation (1), so we only consider decoding times in our results.

From Table I it is apparent that lower values of k (*i.e.*, the number of chunks into which the file is split) yield faster decoding. The data further confirms that it makes sense to use larger field sizes to further reduce k, even with the additional overhead of more expensive field operations, although reducing k indiscriminately would be problematic in maintaining fairness due to quantization errors. As a specific example, a 1MB file chunks represented as $m = 32,768$-element vectors over $q = GF(2^{32})$ can be decoded in one second (*i.e.*, a decoding rate of 1MB/s). At this speed, the ISP-offered download rate is more likely to be a bottleneck than the computations or upload capacities of the various peers.

VI. Conclusions

In this paper we have proposed a new peer-to-peer application that enables users to overcome slow upload bandwidth bottlenecks when remotely accessing data on their home computers. The key to our approach is the use of random linear coding to disseminate the desired content over several peers, thus multiplexing their upload bandwidths in order to fill up the downloading user's data pipe. This model fits very well with into the typical user pattern of short periods of heavy link usage interspersed with long idle times.

$q \downarrow, m \rightarrow$	2^{13}	2^{14}	2^{15}	2^{16}	2^{17}	2^{18}
GF(2^4)	117.28	58.8	30.05	14.99	7.57	3.9
GF(2^8)	34.78	17.52	8.85	4.46	2.29	1.18
GF(2^{16})	10.97	5.53	2.81	1.42	0.72	0.4
GF(2^{32})	3.9	1.96	1	0.51	0.26	0.15

TABLE I

Decoding (encoding) times in seconds for files split into chunks represented as m-element vectors over $\mathbb{F}_q$.

We have also shown that our proposed system provides a natural incentive for peers to voluntarily join and cooperate within our framework. Moreover, our system is asymptotically fair in the sense that each user benefits from unallocated network bandwidth in proportion to its contribution to the system. As such, our system is also resilient to adversarial or malicious collusion, guaranteeing that fairness even when some peers do not use the prescribed bandwidth allocation rule or attempt to interfere with others' access. These analytic conclusions have been confirmed through simulation. We have also demonstrated the real-time applicability of random linear coding for the proposed application.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li and R. W. Yeung, "Network information flow", *IEEE Trans. on Information Theory, vol. 46, pp. 1204-1216*, 2000.
[2] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding", *In Proc. Allerton Conf. on Comm., Control, and Computing*, October 2003.
[3] J. Byers, M. Luby, M: Mitzenmacher and A. Rege, "A Digital Fountain Approach to Reliable Distribution of Bulk Data", *SIGCOMM*, Sept. 1998.
[4] D. Qiu, and R. Srikant, "Modelling and performance analysis of bittorrent-like peer-to-peer networks", *SIGCOMM*, August 2004.
[5] S. Acedanski, S. Deb, M. Medard and R. Koetter, "How Good is Random Linear Coding Based Distributed Networked Storage?", Invited talk, *http://www.netcod.org/, NetCod*, 2005.
[6] E. Calabi and H. S. Wilf. "On the sequential and random selection of subspaces over a finite field", *Journal of Combinatorial Theory (A)*, 22: 107-109, 1977.
[7] R. Ma, S. Lee, D. Yau, "A game theoretic approach to provide incentive and service differentiation in P2P networks", *in Proc. of ACE SIGMETRICS/Performance Evaluation Review*, 2004.
[8] T.S.E Ng, Y. Chu, S.G. Rao, K. Sripanidkulchai, and H. Zhang, " Measurement-based optimization techniques for bandwidth-demanding peer-to-peer systems", *INFOCOM*, April 2003.
[9] Ben Y. Zhao, Ling Huang, Jeremy Stribling, Sean C. Rhea, Anthony D. Joseph, and John Kubiatowicz, "Tapestry: A resilient global-scale overlay for service deployment", *IEEE Journal on Selected Areas in Communications*, 2003.
[10] Foldershare.com, Bytetaxi, Inc.
[11] B. Cohen, "BitTorrent", *http://bitconjurer.org/BitTorrent*.
[12] S. Rhea, P. Eaton, D. Geels, H. Weatherspoon, B. Zhao and J. Kubiatowicz, "Pond: The Oceanstore Prototype", *in Proc. of the USENIX Conference on File and Storage Technologies*, 2004.
[13] Joanna Glasner, "P2P Fuels Global Bandwidth Binge", *Wired*, 2005.
[14] X. Yang and G. de Veciana, "Service Capacity of Peer-to-Peer Networks", *INFOCOM*, 2004.
[15] X. Yang and G. de Veciana, "Performance of Peer-to-Peer Networks: Service Capacity and Role of Resource Sharing Policies", *(to appear) in Performance Evaluation - Special Issue on Performance Modeling and Evaluation of P2P Computing Systems*, 2005.
[16] Z. Ge, D. R. Figueiredo, S. Jaiswal, J. Kurose and Don Towsley, "Modeling Peer-Peer File Sharing Systems", *INFOCOM*, April 2003.
[17] C. Gkantsidis, P. Rodriguez, "Network Coding for Large Scale Content Distribution", *in Proc. of IEEE Infocom* March 2005.
[18] K. G. Anagnostakis, M. B. Greenwald, "Exchange-Based Incentive Mechanisms for Peer-to-Peer File Sharing", *24th ICDCS*, 2004.
[19] B. Yang, T. Condie, S. Kamvar, H. Garcia-Molina, "Non-cooperation in competitve P2P networks", *25th ICDCS*, 2005.
[20] J. Song, C. Sha, H. Zhu, "Nash Equilibria in Parallel Downloading with Multiple Clients", *24th ICDCS*, 2004.