Algorithms for Storage Allocation Based on Client Preferences

Tami Tamir [*] Benny Vaksendiser [†]

Abstract

We consider a variant of the knapsack problem arising in storage management of *Video on Demand* (VoD) systems. Formally, a VoD system services n clients, that are interested in watching movies from a collection of M movies. The system has limited resources: it consists of N disks, each having a limited storage capacity, C, and a limited bandwidth (load capacity), L. Each transmission requires a dedicated stream of one bandwidth (load) unit. This implies that each of the N disks can store movies of total size C and can transmit broadcasts to at most L clients simultaneously.

The problem is therefore reduced to a *class-constrained packing* problem, in which the items to be packed (streams) are drawn from M classes (movies) and have the same unit size. The bins (disks) have a limited capacity, L, and can pack items from at most C classes. This storage management problem motivated the study of class-constrained packing in recent years. In all previous work it is assumed that each client specifies a single movie he wishes to watch and the goal is to allocate storage to movies and transmissions to clients in a way that maximizes the number of clients whose request is granted. In this work we define and study a more generalized setting: For each client j and movie i, let $b_{i,j} \geq 0$ denote the payment that client j is willing to pay for watching movie i. That is, each client provides his complete preference over the whole collection of movies. The objective is to allocate storage to movies and transmissions to clients in a way that maximizes the system's profit given by $P = \sum_{j=1}^{n}\{b_{i,j} |$ movie i is transmitted to client $j\}$.

Our Results: For a VoD system with a single disk we present a hardness proof and a $(1-1/e)$-approximation algorithm. We then extend the approximation algorithm for systems with variable file sizes and/or storage costs, and for the k-round problem, in which there are k synchronized broadcasting rounds.

For multiple disks we first present a $(C-1)(e-1)/Ce$-approximation algorithm. Next, we propose algorithms for solving the problem in two stages. In the first stage an allocation of movies to the disks is determined. In the second stage, given the storage allocation, the bandwidth allocation problem is to decide which of the clients will be serviced by which disk. We present two heuristics for the first task, and an optimal algorithm for the second task. The bandwidth-allocation problem can be reduced to a special case of a min-cost max-flow problem. For this flow problem we present and implement an algorithm based on dynamic programming for efficient detection of negative-cost cycles.

In order to better evaluate the performance of our algorithms we simulated a VoD system, and compared their performances. In our simulated system, as in the real-world, client preferences and payment vectors are power law distributed. We used the Zipf distribution to determine the popularity and payment-readiness of clients. As a result, a few movies are very popular and clients are willing to pay significantly more for watching them.

Our results can be applied to other packing and subset selection problems (e.g. auctions) in which clients provide preferences over the elements.

[*]School of Computer Science, The Interdisciplinary Center, Herzliya, Israel. tami@idc.ac.il
[†]School of Computer Science, The Interdisciplinary Center, Herzliya, Israel. vaksendise.benny@idc.ac.il