Optimal File Distribution in Peer-to-Peer Networks

Kai-Simon Goetzmann[1,⋆], Tobias Harks[1], Max Klimm[1,⋆], and Konstantin Miller[2]

[1] Technische Universität Berlin, Institut für Mathematik
{goetzmann,harks,klimm}@math.tu-berlin.de
[2] Technische Universität Berlin, Telecommunication Networks Group
miller@tkn.tu-berlin.de

Abstract. We study the problem of distributing a file initially located at a server among a set of peers. Peers who downloaded the file can upload it to other peers. The server and the peers are connected to each other via a core network. The upload and download rates to and from the core are constrained by user and server specific upload and download capacities. Our objective is to minimize the makespan. We derive exact polynomial time algorithms for the case when upload and download capacities per peer and among peers are equal. We show that the problem becomes strongly NP-hard for equal upload and download capacities per peer that may differ among peers. For this case we devise a polynomial time $(1 + 2\sqrt{2})$-approximation algorithm. To the best of our knowledge, neither NP-hardness nor approximation algorithms were known before for this problem.

1 Introduction

In the past decade, the concept of data distribution based on peer-to-peer overlay networks has become increasingly popular. A significant fraction of Internet traffic is nowadays generated by peer-to-peer file sharing applications [3]. The key principle underlying peer-to-peer file sharing is that peers who downloaded parts of the file (chunks), start to assist the server in uploading them to other peers. A chunk is the smallest indivisible unit w.r.t. the download source, that is, a peer cannot download parts of a chunk from different sources. It is assumed that the server and the peers are connected via a core network (Internet). The upload and download rates to and from the core are constrained by peer and server specific upload and download capacities. The core itself is usually overprovisioned, thus, there are no capacity constraints present.

While in the past many algorithms and protocols have been studied in the literature for the important problem of optimizing the download process [1,9,10], a systematic performance evaluation of the proposed solutions with respect to *optimal* solutions has not received similar attention. In this paper, we address the fundamental problem of minimizing the total completion time (makespan) of distributing a file among a set of peers in a peer-to-peer network. Due to the intrinsic difficulty of the problem, in this work we restrict ourselves to the case of a single chunk only. To the best of our knowledge, even for this restricted case there are no algorithms with provable performance

⋆ Supported by the Deutsche Forschungsgemeinschaft within the research training group 'Methods for Discrete Structures' (GRK 1408).

T. Asano et al. (Eds.): ISAAC 2011, LNCS 7074, pp. 210–219, 2011.

guarantee known. While the case of multiple chunks still eludes us, we see our study as an important first step towards understanding the general peer-to-peer file distribution problem.

The Model. An instance of this problem is described by a tuple $I = (N, c^d, c^u)$, where $N = \{0, \dots, n\}$ is the set of peers, $c^d = (c^d_0, \dots, c^d_n) \in \mathbb{Q}^{n+1}_{\geq 0}$ is the vector of download capacities from the core network and $c^u = (c^u_0, \dots, c^u_n) \in \mathbb{Q}^{n+1}_{\geq 0}$ is the vector of upload capacities to the core network. We will identify the server with peer 0 and assume that only the server initially owns a file of unit size, which is not divided into several chunks. See Figure 1 for an illustration. A feasible solution $S = (s_{i,j})_{i,j \in N}$ is a family of integrable functions $s_{i,j} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, $i, j \in N$, where $s_{i,j}(t)$ denotes the sending rate of peer i to peer j at time t. We require that every peer $i \in N$ receives its data from a unique peer, denoted by $p(i) \neq i$: $s_{k,i}(t) = 0$ for all $k \neq p(i)$ and all $t \in \mathbb{R}_{\geq 0}$. In addition, only peers that possess the file can send with a positive rate: $s_{i,j}(t) = 0$ for all $i, j \in N \setminus \{0\}$, $t \in \mathbb{R}_{\geq 0}$ with $\int_0^t \sum_{k \in N} s_{k,i}(\tau)\, d\tau < 1$. Finally, the sending rates have to obey download and upload capacity constraints: $\sum_{j \in N} s_{i,j}(t) \leq c^u_i$ for all $i \in N$, $t \in \mathbb{R}_{\geq 0}$ and $s_{p(j),j}(t) \leq c^d_j$ for all $j \in N$, $t \in \mathbb{R}_{\geq 0}$. We denote by $x_i(t) = \int_0^t s_{p(i),i}(\tau)\, d\tau$ the proportion of the file owned by peer $i \in N \setminus \{0\}$ at time t. For notational convenience, we set $x_0(t) = 1$ for all $t \in \mathbb{R}_{\geq 0}$. We denote by $C_i = \inf\{t \in \mathbb{R}_{\geq 0} : x_i(t) = 1\}$ the *completion time* of peer i. The makespan of a solution S is then defined as $M = \max_{i \in N \setminus \{0\}} C_i$. If an instance satisfies $c^u_i = c^d_i = c^u_j = c^d_j$ for all $i, j \in N \setminus \{0\}$ we speak of an instance with *homogeneous symmetric capacities*. If an instance satisfies $c^u_i = c^d_i$ for all $i \in N$ (but possibly $c^u_i \neq c^u_j$ for some $i, j \in N \setminus \{0\}$) we speak of *heterogeneous symmetric capacities*. In both cases, we only write $I = (N, c)$.

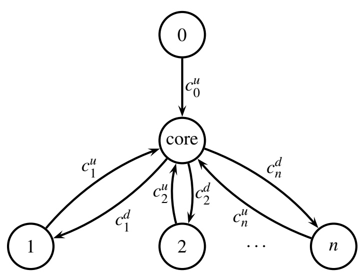

Fig. 1. Graphical representation of the file distribution problem we consider

Previous Work. There is an enormous body of work on the (minimum time) broadcasting problem, multicast problem, and gossiping problem, see [6] for a survey. In the broadcasting problem, the task is to disseminate a message from a source node to the rest of the nodes in a given communication network as fast as possible, see Ravi [15]. When the message needs to be disseminated only to a subset of the nodes, this task is referred to as multicasting, see Bar-Noy et al. [2] for approximation algorithms. In the gossiping problem, several nodes possess different messages and the goal is to transmit every message to every node.

The usual underlying communication model for these problems is known as the telephone model: a node may send a message to at most one other node in each round, and