Mapping of File-Sharing onto Mobile Environments: Feasibility and Performance of eDonkey with GPRS

Tobias Hoßfeld, Kurt Tutschku
Department of Distributed Systems
University of Würzburg, Germany
[hossfeld|tutschku]@informatik.uni-wuerzburg.de

Frank-Uwe Andersen
Siemens ICM N PG SP RC PN,
Siemensdamm 62, Berlin, Germany,
frank-uwe.andersen@siemens.com

***Abstract*— Peer-to-Peer (P2P) file-sharing has become a major application in the Internet with respect to traffic volume which is even surpassing web usage. This characteristic makes P2P commercially attractive to network operators interested in increased traffic. In parallel, the demand for wireless services has caused wireless networks to grow enormously. We assume that P2P file-sharing will be mapped onto mobile environments by its users. This results in a mobile P2P file-sharing service, which we denote as mobile P2P. In this paper, we examine the feasibility of the eDonkey file-sharing service in GPRS networks, detect problems of the interaction between P2P and the mobile network, and find solutions to overcome them. Furthermore, this paper measures and analyzes the characteristics of mobile P2P and gives first empirical performance values. Summarizing, the goal is the analysis of feasibility for an Internet-based file-sharing application in a mobile network and to provide first measurements from two real-world networks.**

I. Introduction

In recent years the internet traffic has revealed a significant trend. The use of P2P file-sharing services on the Internet has grown far more rapidly than browsing in the WWW. Internet service providers (ISPs) experience an amount of P2P traffic which dominates the Internet access [1] and is significant in WAN [2]. P2P has become the killer application which is attractive for the ISPs with regard to commercial aspects, as P2P users utilize the ISP's infrastructure and exchange a considerable amount of data. P2P traffic is on best effort basis and can easily be preempted. However, P2P is trading its decentralized nature by increased communication traffic. In particular, the peers generate a huge amount of signaling traffic for coordinating with each other [3] [4]. High application signaling traffic is considered to be too expensive in mobile networks. This shows the importance of P2P measurements in order to optimize the Internet access, e.g. caching strategies to reduce bandwidth or signaling traffic.

Since the demand for wireless services has caused wireless networks to extend at an enormous rate, the gap between these two important trends is narrowed by the convergence of both services: a P2P file-sharing service over a mobile telecommunication system which is referred to as mobile P2P in this paper. At the time of this writing, there is no operator supported mobile P2P service available. To get an impression of the behavior of P2P in mobile networks, we perform case-by-case measurements.

The aim of this paper is to examine the feasibility of mobile P2P and to give an insight how a general P2P architecture works in a mobile cellular environment. We detect problems of the interaction between P2P and mobile networks, e.g. restrictions because of the air interface, and describe how occurring obstacles, like network address translation (NAT) or firewalls, can be overcome. Finally, this paper measures and analyzes the characteristics of mobile P2P using GPRS transmission technology and gives first empirical performance values.

The broader scope of work is to use the experience gained from our measurements in order to identify mobile P2P specific problems. These are addressed by an architecture proposal [5] to be published recently. Furthermore, the measurements are helpful to create suitable parameter sets for simulative performance analysis. The simulation which we are currently implementing will further validate and verify our architecture proposal for operator supported mobile P2P.

Currently a number of P2P file-sharing applications are available. Due to its current popularity among users, the eDonkey 2000 system[1] is a candidate for mobile P2P. The important characteristics of eDonkey are the distribution of user created content, the open and well-known protocol [6], and the high popularity [7]. The latter one can be derived from the traffic volume of the most popular P2P file-sharing services in fixed access networks [1].

II. P2P Architecture

The eDonkey file-sharing service belongs to the class of hybrid P2P architectures comprising two applications: the eDonkey client and the eDonkey server [2]. The eDonkey client is used to share and download files. The eDonkey server operates as an index server for file locations and distributes addresses of other servers to clients.

The consuming client may operate in a *multiple source download* mode, i.e. it issues two or more requests in parallel to different providing clients. The uploading client keeps the outstanding requests in a list of current downloading requests. Then, the user data is transmitted in several parallel TCP connections from the uploading peers to the requesting peer.

[1]In this paper, we subsume eDonkey 2000 and its derivatives, e.g. eMule, mlDonkey, by the single term "eDonkey".

[2]The terms "client" and "peer" are exchangeable in the context of eDonkey.

The upload management of a peer maintains an upload queue which consists of two lists, the waiting list and the uploading list. The uploading list holds the exchange requests which are currently served. Each served request gets typically an equal share of the upload capacity which may be restricted to a given limit. A download request is served as soon as it obtains an upload slot, i.e. it moves from the waiting list to the uploading list. The complex scoring mechanism of eDonkey decides which request is served next. One important factor of the scoring system is the "high ID/low ID"[3] mechanism to ensure fairness for peers before or behind a NAT or a firewall. A high ID increases the score whereas a low ID reduces it. A peer gets a low ID if the server cannot establish a new connection to the peer. This means that the peer is located behind a firewall or a NAT. The firewall rejects incoming connections and the IP address of the peer is unknown, respectively. This results in an unfair behavior as the peer does not answer file requests.

Further details on the eDonkey architecture and the download mechanisms are given in [3], [6], [8], and [9].

III. Mobile Network Characteristics

General Packet Radio Service (GPRS) is the current, GSM based infrastructure and the confluence of mobile telecommunications and IP data networking. GPRS brings IP-based services to the mobile mass market and has paved the way for 3G networks.

GPRS data rates depend on the overall number and ratio of voice and data users in a cell and the supported data rates of the mobile station (MS). In a GPRS network, the circuit-switched and packet-switched users exist in parallel. Therefore, they compete for the existing resources on the air interface. The dynamic allocation of bandwidth is mainly based upon granting circuit-switched voice traffic first priority, including the option to stop data communications in favor of voice calls. The combination of uplink/downlink channels depends on the mobile terminal and is referred to as the multislot-class.

Currently, the typical multislot-class of mobile terminals is class 8 and limited to 1 uplink and 4 downlink channels with coding scheme CS2. Theoretically, the data rates are 13.4 kbps and 53.6 kbps, as CS2 supports a data rate of 13.4 for one packet data channel.

In the GPRS network, the air interface is the lossy part of the link. The eDonkey (v. 0.40f) application uses TCP for transmitting user data. As a result of the mobile environment, TCP suffers from packet retransmissions due to packet losses [10]. All IP traffic is centrally directed through the GGSN network element. Even in the case of two MS exchanging IP data and attached to the same GGSN, the path between the MS and the GGSN is always traversed twice. This results in high delay times. Figure 1 roughly shows the increasing transfer delay of an IP packet on the data path between two terminals,

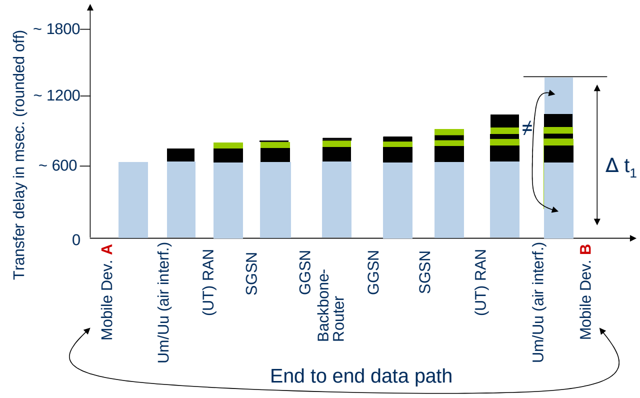


Fig. 1. Delay of an IP packet on the path between two terminals

identifying the crucial parts of the overall path (based on measurements performed by Siemens). All values depicted in Figure 1 relate to an unloaded network.

Furthermore, it has to be considered that the first packet in a packet stream between two mobiles experiences a significantly higher delay than the following ones because of temporary block flow (TBF) setup times. Many papers exist, describing the GPRS system, e.g. [11],[12], present an overview of GPRS, the architecture, the protocols and the air interface.

IV. Problems of Mobile Peers Using eDonkey

Some mobile German operators assign private IP addresses to peers and shield the peers by firewalls and NAT. As mentioned above, these peers would be assigned low IDs resulting in a discrimination in the upload queue. In order to avoid getting a low ID, a consistent address space is required to handle firewalls and NAT which can be realized by a virtual private network (VPN). These operators use firewalls that block mobile-terminating TCP connections. This means that a direct mobile-to-mobile connection is not possible without a VPN. Other German operators do not use firewalling and NAT, thus, a VPN is not required.[4]

A VPN is an extension of a private intranet across a public network, such as the Internet. Only authorized users can access the network which enables the separation of public and private communities and forming of user groups. On the other hand, the VPN concept leads to an increased transfer volume due to additional protocol overhead, i.e. the load on the air interface increases. An additional network element, the VPN gateway, is required that has to be administrated.

Using a VPN means that the index server and all peers must be connected to the VPN, i. e. a peer sends data to another peer via the VPN gateway, as the peers communicate by using internal addressing schemes. In our measurements, we used a Point-to-Point-Tunneling Protocol (PPTP) based VPN. In this context, the VPN gateway is also denoted as PPTP server.

How expensive is the application of a VPN?
Figure 2 shows the protocol stack of the different network

[3]The eDonkey *ID* identifies peers and is assigned upon registration of a client at an index server.

[4]In our measurements, we denote the operator without using firewalling and NAT as *A*, and the other as *B*.

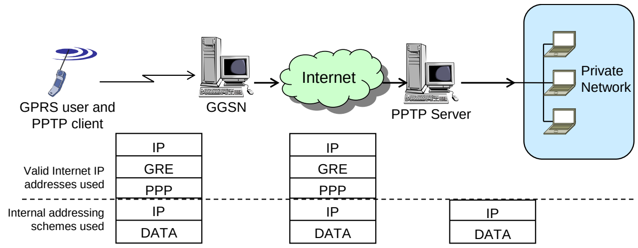


Fig. 2. Connecting a PPTP client to the private network

elements. A peer uses PPTP in order to connect to the private network which is formed by several (mobile or fixed) P2P clients and at least one (internal) P2P server. It is interesting to examine how expensive a VPN is as a solution to get over firewalls and NAT. The costs are expressed by the protocol overhead, the download time, and the received bandwidth on application layer, as the bandwidth of a mobile cellular system is expensive and limited. The overhead of an encapsulated data packet via VPN is at least 28 Byte [13]. Header compression schemes may be applied, but are not widely deployed. Since the user is mainly interested in the resulting download time which directly reflects the introduced overhead as well, we make download time our adequate criterion for both VPN and non-VPN environments.

Is a multiple source download possible in a mobile network?
It may be conceivable that the performance of a multiple source download with many sources does not significantly differ from a single source download with a small (or zero) number of users in the waiting list of the upload queue.

Is the performance influenced by the content type?
Regarding the mobile equipment of a peer, a set of content types seems to be typical for mobile P2P users. Nowadays, the mobile handsets support multimedia, e.g. polyphony ringing tones or even mp3-audio files and taking pictures with an integrated digital camera. These file types are reflected by different distributions of the file size. On the other hand, the memory capacities are limited up to several megabytes. We investigate the performance of a mobile P2P user with respect to the content type and the corresponding file size, e.g. to answer the question is it practical do download mp3-audio files.

To summarize, we first investigate the feasibility and performance of mobile P2P, compare it to a fixed network scenario, and answer the main questions discussed in this section.

V. Measurement Scenarios

The measurements took place in between December 2003 and February 2004 at the University of Würzburg in Germany. We selected two German GPRS operators, *A* and *B*, with differently profiled Internet access services. *A* assigns a global IP address to each mobile which enables a direct communication between a mobile and another mobile or another fixed peer and thus, a VPN is not required. On the other hand, operator *B* uses a firewall which denies mobile-terminating TCP connections, but allows connections to an external VPN gateway. This means that a VPN is required for direct mobile-to-mobile communication. Additionally, a VPN avoids low IDs for using eDonkey over the GPRS system of operator *B*.

Two alternatives have been considered during the measurements of a P2P file-sharing service with GPRS as wireless access system. First, the P2P file-sharing application uses GPRS as a bit pipe connection to the public Internet: the public Internet scenario, abbreviated as ”pub”. Secondly, the P2P file-sharing application resides in a VPN domain and uses virtual connections among the participants for exchanging information: closed network scenario, short denoted as ”vpn”.

The physical access of a peer can be either Ethernet for fixed network access (max. 100 Mbps) or GPRS for mobile network access (max. 53.6 kbps). The fixed peers, the internal eDonkey server, and the VPN gateway are located within the LAN of the Department of Distributed Systems at the University of Würzburg. The LAN is connected to the university's campus LAN by a half-duplex FastEthernet link. The campus LAN is linked to the public Internet via the German Research Network (DFN) using a Gigabit-Ethernet line shaped to 155 Mbps.

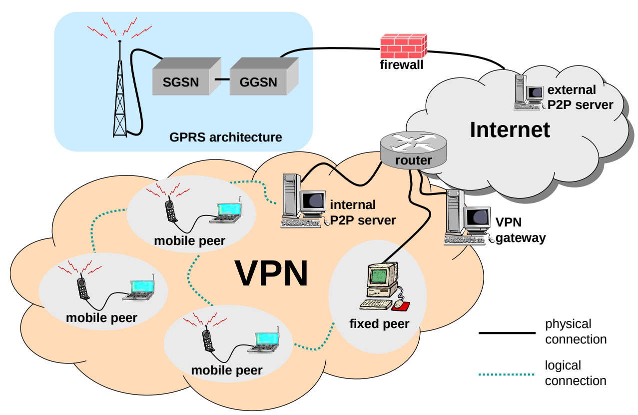


Fig. 3. Network architecture for a mobile P2P file-sharing service

Figure 3 shows the closed network architecture for a mobile P2P file-sharing service over GPRS. This architecture differs from the public Internet scenario by the application of a VPN. Therefore, each host initiates a Point-to-Point Tunneling Protocol (PPTP) connection to the VPN gateway. Furthermore, it is not possible in the closed network scenario to communicate with an external server that is not connected to the VPN gateway.

The firewall which separates the GPRS domain from the Internet prevents that a host outside the GPRS domain can initiate a connection to a mobile peer. It is only used by operator *B*. The logical connection between the peers indicates the flow of information independent of the physical connection.

The mobile P2P clients consist of a mobile phone which is used as a modem and a laptop running Windows2000. The mobile phone is connected to the laptop via a RS232 serial interface. The used mobile phones are Siemens S45, Siemens

TABLE I

PARAMETERS OF THE MEASUREMENT SCENARIOS

P2P application	eMule 0.40f
mobile phone	Siemens S45, S55, ME45 of multislot-class 8 as modem via RS232 serial interface
operating system of the peers	Windows 2000 (SP4)
packet capture software	WinDump 3.6.2 (using PCap 2.3) installed on peers
VPN gateway	PoPToP v1.1.3 - a freeware PPTP server running under SuSE Linux 8.2, kernel version 2.4.20
internal eDonkey server (non-public)	eserver 16.43-i686 (Lugdunum)
external eDonkey server (public)	207.44.200.40:4242 with more than 50,000 users and 2,900,00 files

S55, and Siemens ME45 which support the GPRS multislot-class 8, i.e. one uplink slot and maximal four downlink slots, resulting in a maximal uplink and downlink bandwidth of 13.4 kbps and 53.6 kbps on the MAC layer. A complete packet trace was captured for every mobile or fixed peer during the measurement campaign.

The external P2P server is a well-known eDonkey server with a fixed IP address. In contrary, the internal P2P server is part of the LAN at the Department of Distributed Systems which is non-public. Therefore, it is only accessible for peers within the VPN. The used software and hardware for the measurements are summarized in Table I.

In order to investigate a single source download, a peer provides a unique file that is yet unknown to the eDonkey network. That way, it can be assured that the number of sources to download from is one. On the other hand, a multiple source download is realized by downloading a popular file which is shared by at least two peers. In both cases, we download files of the same content type and the same size.

TABLE II

FILE TYPES AND SIZES FOR THE MEASUREMENT SCENARIOS

content type	**mean size**	**format**
ring tone	6,830 Byte	mid, mmf
game	39,114 Byte	jar
image	483,525 Byte	jpg
song	4,726,618 Byte	mp3

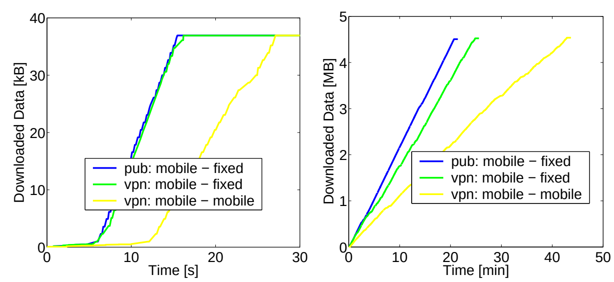
The investigated file types represent a typical set of files that seems to be interesting for mobile users: ring tones, games for java-capable mobiles, digital camera images, and mp3-audio files. The first two file types are already popular in today's mobile generation, whereas the latter become more and more popular because of the improved mobile equipment with integrated digital cameras and an increasing memory capacity. We measured the file sizes for the corresponding content types and used the mean file size as reference value in our measurement scenarios, summarized in Table II.

VI. RESULTS

The measurements provided in this work are carried out in real-world networks for two different operators. This section is structured as follows. First, the feasibility of mobile P2P is investigated by means of the download time, the TCP packet error rate, the download abort rate, and the P2P setup, download, and idle time. Secondly, the overhead due to VPN is considered, i.e. the transmitted volume and the number of packets on the IP layer. Then, we take a look at the feasibility of multiple source downloads as one of the eDonkey main features. Finally, we show the influence of the downloaded file content type and file size on the performance of the P2P file-sharing services.

A. Feasibility of Mobile P2P

We consider a single source download where the two peers are connected to the internal eDonkey server. The downloading peer has a mobile access via the operator *B*. The physical access of the sharing peer[5] is chosen to be fixed via Ethernet and mobile via operator *B*, respectively. The considered scenarios are the closed network scenario using a VPN and the public Internet scenario. As a direct mobile-to-mobile connection is not possible for the operator *B* without using a VPN, this leads to three different scenarios.



(a) Game (b) Song

Fig. 4. Transmitted data volume over time for downloading a file

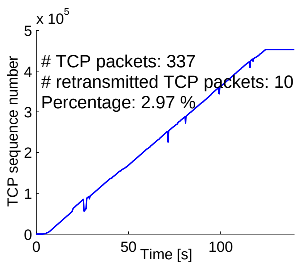
Figure 4(a) illustrates the amount of downloaded data in [kB] over time in [s] for downloading the game; the legend describes the scenarios by "network scenario: access type of downloading peer - access type of sharing peer". The download behavior seems similar for the scenario with the mobile downloading peer and the fixed sharing peer regardless of using VPN or not because of the small file size. In contrast, Figure4(b) shows that it takes indeed more time to transmit the song with the VPN. Certainly, the download time is increased for direct downloading a file from a mobile peer, as the uplink bandwidth of the sharing mobile peer determines the download bandwidth of the downloading peer.

The same measurements were also performed with an external index server, but there are no remarkable differences for internal and external index servers what we expected, because the index server is only responsible for exchanging information on the files shared by the peers and does not influence the download mechanism between the peers. For this

[5]The terms "sharing peer" and "serving peer" are exchangeable in this work.

reason, the location of the server is not noted in the following sections.

1) TCP Packet Loss Rate: Packet loss may occur in GPRS even though the radio link protocol retransmits corrupted data [14]. A TCP packet loss can be detected by retransmissions of TCP segments. A loss free transmission results in a strictly monotonic increase of the sequence number. Figure 5 shows multiple retransmissions visible as sharp spikes for an upload from a fixed peer to a mobile peer. No loss was observed for downloading the ring tone. It has to be noted that the complete ring tone file is transmitted within one TCP segment. The observed packet loss rate for larger files is approximately 3%, which is consistent with previous measurements [15].



file type	error rate
ring tone	0.00 %
game	2.94 %
image	2.97 %
song	2.71 %

Fig. 5. Packet loss rate as ratio between retransmitted and totally transmitted packets on TCP layer

2) Aborted Downloads: A download is detected to be aborted if no more data is send from the sharing peer to the downloading peer for at least 10 min or if the GPRS connection hangs up. It should be noted that in case of an aborted download in eDonkey, the data is not completely discarded. The download can be resumed later and only the missing data has to be transmitted.

If the sharing peer has fixed access, no aborts were observed, i.e. the success rate is 100%. However, if all involved peers use mobile access, we noticed a significant abortion rate of downloads. It should be noted that only single source download is used, here.

We first measured the set of possible scenarios with the operator *B* and compared the results to the same scenarios for operator *A*. The results are the same with respect to download time, transmission rate, and packet loss rate. Therefore, the results in the following sections are only given for operator *B*. However, the number of aborted downloads differs significantly between operator *B* and *A*.

Table III shows the success rate for downloading the song between two mobile stations for both operators. In order to explain the aborted downloads, we investigated the exchange of the same file by FTP between the mobiles. Again, we noted a higher success rate for operator *A*. A reason for this observation cannot be derived directly by our measurements. The most likely explanations are errors in early software implementations of mobile handsets and network infrastructure.

Aborted downloads require the user to re-initiate the data transfer. While in some cases even the entire PDP context and thus the GPRS connection itself were lost, resulting in a new IP address for the peer, the peer does not loose its credit

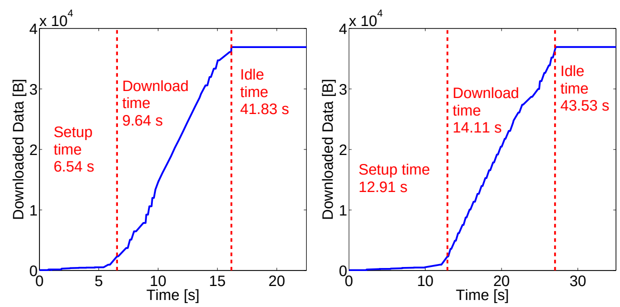
TABLE III

SUCCESS RATE FOR DOWNLOADING THE SONG VIA P2P AND FTP

appl.	downl.	sharing	operator	success
eDonkey	mobile	mobile	*B*	$0 = 0:5$
FTP	mobile	mobile	*B*	$0.5 = 4:8$
eDonkey	mobile	mobile	*A*	$0.6 = 3:5$
FTP	mobile	mobile	*A*	$0.75 = 6:8$

points which are based on a user hash, independent of the IP address. The peer will also soon obtain the old queue positions for download.

3) P2P Setup Time, Download Time, Idle Time: The *P2P setup time* is defined as the time period from the observation of the first TCP SYN packet to the first TCP packet containing user content. The *download time* is the time interval from the observation of the first TCP to the last TCP packet containing user content. The *idle time* is considered as the time from the last TCP packet containing user content until the observation of a TCP FIN or TCP RST packet for this connection.



(a) Fixed to mobile

(b) Mobile to mobile

Fig. 6. Transmitted data volume over time for downloading the game

Figure 6 illustrates the downloaded data volume over time and the above introduced time intervals. Figure 6(a) depicts the single download of the game from a fixed peer to a mobile peer, and Figure 6(b) from a mobile peer to a mobile peer. Since in the mobile to mobile case, the air interface has to be passed twice, the setup time is twice as much as in the mobile to fixed case, see also Figure 1. The download time is determined by the minimum of the download bandwidth of the requesting peer and the upload bandwidth of the sharing peer. Therefore, the download time is significant larger in the mobile-to-mobile case. The idle time is independent of the connection type and dominated by a timeout mechanism of the eDonkey application.

B. Overhead due to VPN

The overhead introduced by using a VPN is characterized by the increased data transmission volume due to additional header information required by PPTP and by the number of transmitted packets on TCP layer. Figure 7 shows the number of transmitted TCP packets for two scenarios using VPN and one without the application of VPN. Using a VPN leads to

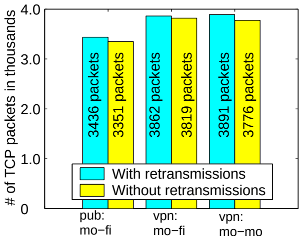


Fig. 7. Number of transmitted packets on TCP layer

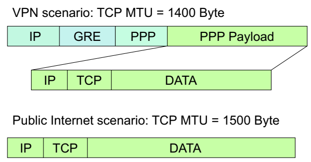


Fig. 8. MTU for the different scenarios

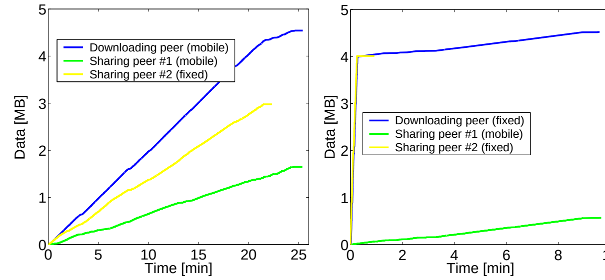
a higher number of transmitted packets because of different segmentations of user data according to TCP's maximum transmission unit (MTU), cf. Figure 8.

Figure 4(b) reveals that the same amount of user data is transmitted, however, the download takes longer due to the transmission of additional header information.

C. Multiple Source Download

In this Section, we investigate the feasibility and performance of multiple source downloads (MSD). First, we consider two peers sharing the complete file and a single peer requesting the download. Figure 9 shows the uploaded data volume of the two sharing peers with fixed access and the total downloaded data volume of the requesting mobile peer. MSD does not become effective for small files, like the game (cf. Figure 9(a)). In this case, the requesting peer receives in one download connection almost all of the requested data. Contrary to this, we observe for large files an efficient MSD, cf. Figure 9(b). The requested data volume is equally split between the two sharing peers.

In the second scenario, we investigate the influence of the access type of the requesting peers on the MSD mechanism, while one sharing peer has mobile access and the other one has fixed access. Figure 10(a) shows the MSD for a downloading mobile peer. This case reveals the asymmetry of the mobile equipment, see Section III. The mobile downloading peer has four slots for downloading data. The mobile sharing peer can only use one slot due to his uplink restrictions. The remaining



(a) Mobile (b) Fixed

Fig. 10. Multiple source download for a large file by peers with different access types

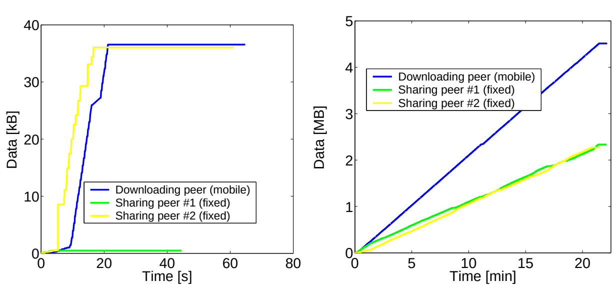
downlink capacity of the mobile downloading peer is utilized by the fixed sharing peer.

Figure 10(b) depicts the MSD behavior for the same scenario with a fixed downloading peer issuing requests to all sharing peers. The fixed sharing peer serves this request with high throughput. The mobile sharing peer is also serving the file request, immediately. However, he provides the minimal amount of data eDonkey transmits for request (which is in eDonkey three blocks, each of 180 kB [6]). The downloading peer completes the file after receiving the data from the mobile sharing peer. A redirection of the download request to another peer which can serve the request faster would reduce the download time. So far, this is not possible in eDonkey.

D. Influence of Content Type and File Sizes

The eDonkey application compresses the user data before transmitting. This will result in shorter download times for compressible content types like ring tones, midi-files, or text files. In Table II, the mean values of the file size for the considered content types are given. As seen in the previous section, the file size influences the multiple source download behavior.

In our measurements, eDonkey transmits the content via TCP. TCP's bandwidth-delay-product (BDP) indicates the minimum amount of data that shall be outstanding in order to fill the link capacity. A high round trip time (RTT), which is typically in a mobile environment, leads to a high BDP



(a) Game (b) Song

Fig. 9. Multiple source download from two peers with fixed access

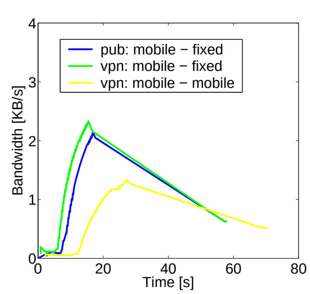


Fig. 11. Throughput for game

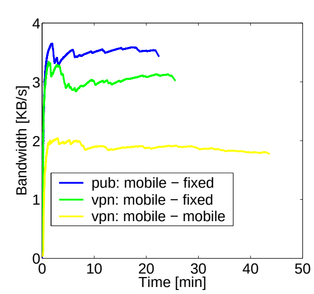


Fig. 12. Throughput for song

which requires a high minimum amount of data to utilize the link completely. Small files are already transmitted before the link is fully utilized. The RTT for a mobile-to-mobile connection is typically 1300 ms from our experience. With a bit rate of 53.6 kbps, we obtain a BDP of 75.04 kb. For a DSL connection with a bit rate of 768 kbps and a RTT of 30 ms, the BDP is 23.04 kb, i.e. three times lower. Therefore, the maximal download throughput cannot be reached for small files, since the BDP is too high. The maximum throughput for the downloading the game, cf. Figure 11, is in all scenarios significantly smaller than for downloading the larger song, cf. Figure 12. The clear decrease of the throughput at the end of the download of the game is caused by the idle time (see Section VI-A.3) after the complete transmission.

VII. Conclusions and Outlook

In this work, we provided first measurements on the performance of a mobile P2P file-sharing service. The measurements were carried out in real-world networks for two different operators. We demonstrated that mobile P2P is feasible already with today's technologies. However compared to fixed P2P, throughput and efficiency are lower, as expected. Particularly, the direct exchange of large parts of files between two mobile peers is not practical. Furthermore, traversing of the air interface has to be minimized in order to reduce the transmission delay. This could be achieved by the application of a cache, which has also the advantage of overcoming the asymmetric access bandwidths of mobile stations.

Multiple source download is not required for small files. As mentioned above, large parts of files should also not be transmitted. This characteristic indicates that there seems to be an optimal segment size for MSD which depends on the total file size and the capacity of the access of the sharing peers. In addition, sharing peers should be selected with respect to their responsiveness.

In future studies, we will perform additional measurements in order to obtain more comprehensive statistical characterizations of mobile P2P file-sharing and to investigate P2P for the upcoming UMTS radio bearer types. Further research should be devoted to how to optimize P2P file-sharing services in mobile environments.

Acknowledgment

The authors would like to thank Uwe Seiler for performing the measurements, Dr. Andras Balazs for providing the data in Figure 1, Dr. Cornelia Kappler and Prof. Phuoc Tran-Gia for fruitful discussions, and Nicoletta Bieber for helps in hardware.

References

[1] N. Azzouna and F. Guillemin, "Analysis of ADSL traffic on an IP backbone link," in *GLOBECOM 2003*, San Francisco, California, Dec. 2003.

[2] S. Sen and J. Wang, "Analyzing peer-to-peer traffic across large networks," in *2nd Internet Measurement Workshop*, Marseille, France, Nov. 2002.

[3] K. Tutschku, "A measurement-based traffic profile of the edonkey filesharing service," in *5th Passive and Active Measurement Workshop (PAM2004)*, Antibes Juan-les-Pins, France, Apr. 2004, pp. 12–21.

[4] K. Tutschku and H. deMeer, "A measurement study on signaling on gnutella overlay networks," in *Fachtagung - Kommunikation in Verteilten Systemen (KiVS) 2003*, Leipzig, Germany, Feb. 2003.

[5] Frank-Uwe Andersen, Hermann de Meer, Ivan Dedinski, Cornelia Kappler, Jens Oberender, and Kurt Tutschku, "An architecture concept for mobile p2p file sharing services," in *Workshop at Informatik 2004 - Algorithms and Protocols for Efficient Peer-to-Peer Applications*, Ulm, Sep. 2004.

[6] "eMule Project Team," `http://www.emule-project.net`.

[7] G. Wearden, "eDonkey pulls ahead in europe p2p race," `http://business2-cnet.com.com/2100-1025_3-5091230.html`.

[8] F. Lohoff, "Lowlevel documentation of the eDonkey protocol," `http://silicon-verl.de/home/flo/software/donkey`.

[9] "eDonkey2000 Home Page," `http://www.eDonkey2000.com/`.

[10] K. Brown and S. Singh, "M-TCP: TCP for mobile cellular networks," in *ACM Computer Communications Review*, 1997.

[11] Gunnar Heine and Holger Sagkob, *GPRS-Gateway*, Franzis Verlag, 2001.

[12] C. Bettstetter, H.-J. Vogel, and J. Eberspacher, "GSM Phase 2+, general packet radio service GPRS: Architecture, protocols and air interface," in *IEEE Communications Surveys*, 1999, vol. 2.

[13] D. Farinacci et al., *RFC 2784 - Generic Routing Encapsulation (GRE)*, 2000.

[14] 3GPP TS 24.022, "Radio link protocol (rlp) for circuit switched bearer and teleservices," `http://www.cl.cam.ac.uk/users/rc277/linkchar.htm`.

[15] Joel Cartwright, "Gprs link characterization," `http://www.cl.cam.ac.uk/users/rc277/linkchar.htm`.