University of Würzburg
Institute of Computer Science
Research Report Series

Impact of Vertical Handovers on Cooperative Content Distribution Systems

Michael Duelli, Tobias Hoßfeld, and Dirk Staehle

Report No. 428 June 2007

University of Wuerzburg, Germany
Institute of Computer Science
Department of Distributed Systems
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-888-6640, fax: (+49) 931-888-6632
`{duelli,hossfeld,dstaehle}@informatik.uni-wuerzburg.de`

Impact of Vertical Handovers on Cooperative Content Distribution Systems

Michael Duelli, Tobias Hoßfeld, and Dirk Staehle
University of Wuerzburg, Germany
Institute of Computer Science
Department of Distributed Systems
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-888-6640, fax: (+49) 931-888-6632
`{duelli,hossfeld,dstaehle}@informatik.uni-wuerzburg.de`

Abstract

We evaluate the performance of a P2P-based content distribution system in heterogeneous, wireless networks. The mobile users coordinate each other with cooperation strategies enabled by the multi-source download mechanism, as in eDonkey or BitTorrent. Due to the mobility, vertical handovers between different wireless access technologies are required which may result in transmission delays and IP address changes of the switching peer. Hence, connections among users have to be reestablished and downloading users are requeued at a providing peer's waiting queue. In detail, we investigate the impact of requeueing with each VHO as well as the use of mechanisms that preserve the IP address and connections beyond VHOs, like MobileIP. Another important phenomenon occurring with VHOs is the abrupt change of the available bandwidth, e.g., from a fast WLAN connection to a rather slow UMTS connection. We evaluate the download times for files by means of simulation while considering different load scenarios in today's and future network layouts of the B3G network. As a result of the performance evaluation, we derive a new time-based cooperation strategy that counters the impact of mobility. Instead of downloading individual blocks of a file, a user gets a time slot at a providing peer. We show that this leads to a significant performance gain.

Keywords: P2P, eDonkey, Mobility, B3G, VHO, Modeling, Analysis, Optimization

1 Introduction

Current telecommunication systems reveal two major trends: heterogeneous wireless networks and *peer-to-peer* (P2P) file sharing systems. The latter is a common application nowadays and makes most of current Internet traffic, as several studies, e.g., [1] and [2], have shown. P2P file sharing is based on cooperation among the users in the system, called *peers*, to enable an efficient content distribution. This requires mechanisms to coordinate and control the access to resources of the peers, i.e., their upload bandwidths and the desired contents. The fundamental *multi-source download* (MSD) mechanism, as used in *eDonkey* or *BitTorrent*, allows requesting users to order and download the desired data from several providing peers in parallel. Therefore, a file is typically split into chunks of fixed size which are exchanged among the peers. The providing peer then schedules the download requests and serves the user according to some cooperation strategy.

In prior work [3, 4], we investigated by means of measurement and simulation whether P2P file sharing is feasible in wireless networks. We found that UMTS as radio access technology already allows a content distribution service for mobile-related contents, like ring tones or small video files in the order of several megabytes. However, we consider a heterogeneous wireless network with different infrastructure-based radio access technologies, in particular UMTS and WLAN. This is referred to as *beyond third generation* (B3G) network.

A mobile user moving through this landscape needs to perform *vertical handovers* (VHO), i.e., pass the ongoing connections from one access system to another, as well as from one operator to another. A VHO implies some delay to reestablish the connections. During this period of time, no application data is transferred. In addition, registering to a new access technology might also change the peer's IP address which leads to the loss of all TCP connections currently opened for file transfer. But even worse, on application layer, when contacting a providing peer with new IP address, the peer does not keep its old position in the providing peer's waiting queue but reenters at the end of the queue and waits to be served.

Another important phenomenon in B3G networks, the switching between radio access technologies, results in an abrupt and dramatic change of the mobile peer's uplink and downlink capacity. Within milliseconds, a previously highly attractive peer with good Internet connection can become a very slow peer that may slow down the performance of the whole content distribution process. In this paper, we focus on the qualitative effect that three VHO phenomena have at application layer: abrupt bandwidth change, transmission delay, and change of IP address. Investigating these effects requires simulation runs at rather long time scales which makes it impossible to simulate background traffic. Consequently, we focused on simulating the P2P mechanism and the mobility of the users while simply assuming fixed transmission rates for WLAN and UMTS cells. As a result, we are not able to make quantitative statements on the downloading times for certain files in a certain environment but qualitative statements that are relevant when designing a content distribution service in a heterogeneous wireless environment with mobile users.

The goal of this paper is to evaluate this mobile P2P system and mitigate the impact of the users' mobility on the performance of distributing content. The P2P file sharing system is oriented towards *eMule*, a popular P2P client based on the eDonkey protocol. In particular, we investigate the impact of requeueing with each VHO as well as the use of mechanisms that preserve the IP address and connections beyond VHOs, like *MobileIP*, at the cost of additional transmission delays. We evaluate the download times for files among the users by means of simulation while considering different load scenarios in today's and future network layouts of the B3G network. In the future network layout, we assume a better WLAN coverage than in today's network layout. The question arises whether the increased capacity due to the higher WLAN density dominates the drawbacks of VHOs on P2P file sharing systems. We only want to make a qualitative and not a quantitative statement on these results focusing on the visible effects of mobility.

As a result of the performance evaluation, we derive a new *time-based cooperation* (TBC) strategy that counters the impact of mobility, in particular, abrupt changes in the available uplink capacity. Instead of downloading individual blocks of a file, a *volume-based cooperation* (VBC) as is common, a user gets a time slot at a providing peer. We show that this leads to a significant performance gain w.r.t. download efficiency and also regard fairness.

The remainder of this paper is organized as follows. In Section 2, we present related work on mobile P2P w.r.t. file sharing in cellular environments. Section 3 introduces the simulation model and formulates the objective of this paper more clearly. In Section 4, we analyse the impact of mobility in different network layouts and load situations, and emphasize the effects of mobility and VHO on the system's performance. In Section 5, we introduce the new time-based cooperation strategy to mitigate the impact of mobility on P2P networks that respects mobility in heterogeneous environments. In Section 6, we summarize our main results.

2 Background and Related Work

The research on P2P has received a lot of attention in the last years. A fundamental overview on P2P applications and systems is given in [5] discussing among others P2P file sharing networks as well as P2P techniques in mobile and ubiquitous environments. However, the combination of both, i.e., P2P file sharing in a mobile environment especially in infrastructure-based wireless networks, and its performance analysis is missing.

In general, the term mobile P2P (MP2P) extends the P2P paradigm to the domain of mobile, wireless networks. MP2P applications and protocols cover a broad range of use. Recently, MP2P research projects have received high attraction which is reflected by the popularity of latest IEEE workshops [6, 7]. However, most of the work addresses structured P2P networks based on distributed hash tables as lookup-service or considers mobile ad hoc networks [8, 9, 10]. Epidemic content distribution is also a typical application for infrastructure-less wireless networks.

In the context of infrastructure-based mobile networks, some investigations on P2P-based content distribution exist. The authors of [11] propose a JXTA solution to create a mobile file sharing environment in 3G environment. In previous work [3, 4], the feasibility of P2P file sharing in UMTS networks was shown. An architecture concept [12] was proposed to improve the system performance using caching peers for popular contents. In contrast to previous work, this paper addresses a heterogeneous wireless network with different infrastructure-based radio access technologies, e.g., UMTS and WLAN. To the best of our knowledge, this is the first paper which describes the impact of VHOs on P2P file sharing applications based on an MSD mechanism.

The effect of heterogeneous, but fixed link capacities in BitTorrent-like file sharing systems was analytically evaluated with a simple fluid model in [13]. It is shown that bandwidth heterogeneity can have a positive effect on content propagation among peers. [14] identifies the principal design choices of content distribution that draw the behavior of the system. Among others, the structure of the P2P overlay and the cooperation strategy are emphasized. According to [14], a cooperation strategy is the result of three factors coupled together, the peer selection strategy, the chunk selection strategy, and the network degree. In [15], a robust cooperation strategy is presented to overcome the problem of leeching peers and starving chunks in the system, while the cooperation concept in [16] makes peers help each other in download-

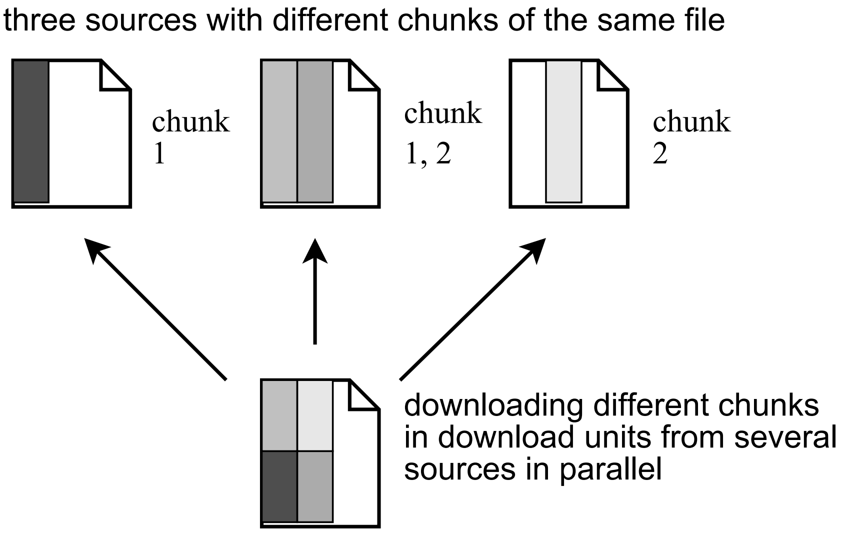


Figure 1: Basic principle of multi-source download.

ing data. However, these strategies do not take into account changing uplink and downlink capacities of peers, as caused by VHOs in a B3G network.

3 Modeling and Simulation Description

We consider a content distribution system in a heterogeneous wireless environment. In particular, we focus on a *multi-source download* (MSD) mechanism which is based on the *eDonkey* protocol as implemented in the eMule application. The investigated radio access technologies comprise an area-wide UMTS network and WLAN hotspots which may overlap. The mobile users move in the landscape and perform VHOs between both technologies or between different WLAN cells. In this context, the switch from one WLAN cell to another is also denoted as VHO, as it might cause an additional delay and the re-assignment of IP addresses. In the following, the different models and their implementation in the simulation are explained in more detail.

3.1 Multi-Source Download Mechanism

MSD is enabled by dividing a file into smaller chunks and blocks which are subparts of chunks. According to eMule, the chunks have a size of 9500 kB and the block size is 180 kB. A downloading peer requests blocks from multiple serving peers, i.e., from sources of that file, and might obtain them from multiple sources in parallel. As soon as a peer has downloaded a complete chunk, it becomes a source for the file, and can redistribute the already received chunks and their blocks. This basic principle of MSD is illustrated in Fig. 1. The benefit of MSD lies in the speed-up via the parallel download of data and the faster creation of additional sources for chunks. As a result, MSD does not rely on a single source and can therefore avoid bottlenecks and overcome churn.

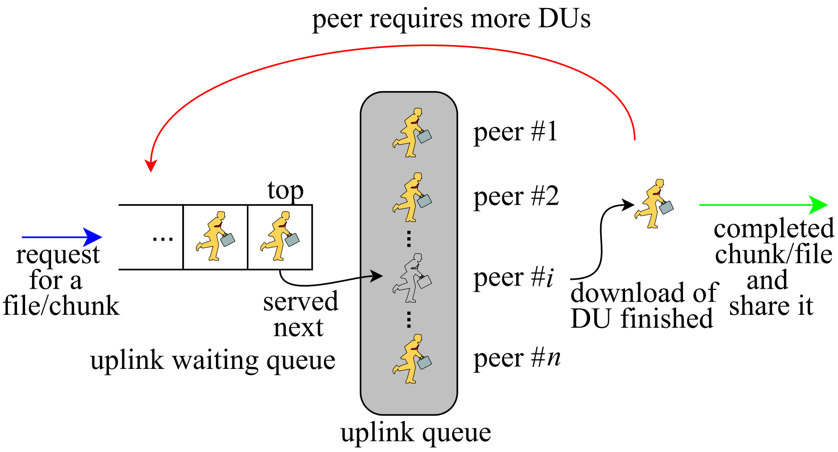


Figure 2: Upload queue of a providing peer.

In order to cooperatively share and exchange the files in such a content distribution network, resource mediation and resource access control functions are required. Resource mediation functions are used to locate the resources in the overlay. They vary from centralized concepts such as index servers, as in *eDonkey*, to highly decentralized approaches such as flooding protocols, as in the *Gnutella* network, or distributed hash tables (DHT), as used in the *Chord* protocol. In this work, we assume the existence of global knowledge of providing peers for all files, which may be achieved by index servers as in eDonkey. The focus is on the P2P resource access control, i.e., mechanisms to permit, prioritize, and schedule the access to shared resources. This is implemented via cooperation strategies in the P2P system.

A peer can download from an arbitrary number of sources in parallel. While the number of parallel download connections is not limited, the number of parallel upload connections at a peer is restricted to a maximum of n in order to guarantee a certain minimal bandwidth. In [15], we have shown that the restriction to a single upload connection is appropriate in a cooperative environment with peers willing to contribute by uploading data. Leeching peers and free-riders are out of the scope of this work. In the simulations, we use $n = 1$, i.e., a peer provides only a single upload connection independent of its upload capacity.

A peer sends a download request to a peer providing the desired file. If the provider already serves n peers, it pushes the request into its uplink waiting queue. As soon as an upload connection becomes available, the first peer in the uplink waiting queue is served. For short, the uplink waiting queue is a first-in-first-out (FIFO) queue. While being served, each peer downloads a specific amount of data in a row. In the current eMule application, these are three blocks resulting in a so called *download unit* (DU) of size 540 kB. After completing a DU, a peer will either reenter the waiting queue at the end or leave this peer, if it has already finished downloading the desired data. The upload queue model is demonstrated in Fig. 2. We also consider churn, i.e., peers alternate between online and offline phases with randomly distributed durations. If a peer goes offline, the existing data connections are dropped, but the already downloaded bytes of a DU are stored and do not get lost.

3.2 Mobile Users in a B3G Network

The mobile peers of the content distribution system are connected via UMTS or WLAN to the Internet. We assume an area-wide coverage of UMTS with a fixed transmission rate of 384 kbps in downlink and 64 kbps in uplink direction. For the WLAN technology, we assign a fixed symmetric bandwidth of 1 Mbps for up- and downlink each. Note, that we do not consider radio resource management mechanisms of the wireless network, like admission, power, or rate control, as we aim at the qualitatively evaluation of the effect of VHO on the P2P system. In addition, we do neither consider background traffic in the wireless network nor the case that multiple peers share the capacity of one cell. Including these effects into the simulation would on the one hand lead to unbearable simulation times and on the other hand blur the clear impact of the VHO only.

The WLAN cells are randomly uniformly distributed within the considered area. We use the disc model with a radius of 50 m to describe the coverage area of a single WLAN cell. In our simulations, we consider a typical city center which is modeled as a square of length 2400 m. According to the investigated scenario, we distinguish between a today's and a future network layout which only differ in the WLAN coverage. In *today's network layout*, we assume 19 WLAN cells according to the current number of public WLAN cells in Würzburg's city center of a German operator providing UMTS as well as WLAN [17]. In the *future network layout*, we assume a much better WLAN coverage with 200 WLAN access points.

For the users' mobility, we did not use a classic mobility model to simulate every movement of the user like *random direction* [18], but a more abstract way of modeling mobility which we introduce in the following sections.

3.2.1 Abstraction from Mobility

By using a specific network layout and conventional, detailed modeling of user mobility, we are only able to find results for these applied models. Since we want to find results for any scenario, we introduce a new framework to describe mobility in the context of cellular wireless networks. This framework is called *Abstract Mobility Model* (AMM). The AMM subsumes the network layout and the user mobility by a semi-Markov model. Thereby, the positioning of transceivers on the simulation plane and the explicit selection of the transceivers' technologies get superfluous. The AMM is more powerful than an explicit modeling of mobility and network layout, as an arbitrary mobility model and network layout can be mapped by AMM.

Additionally, the AMM brings a speed-up w.r.t. computational time. This speed-up is explained by the way we model mobility and calculate the mobility events. The conventional approach creates a lot of events. As illustrated in Fig. 3, there is an event for each waypoint, i.e., each separate move, and an event for each horizontal and vertical handover. In this figure, also "pseudo horizontal handovers" are depicted. Their creation originates from aspects of implementation. Instead of looking for the right UMTS NodeB after leaving a WLAN coverage area, we let a peer also notice handovers between UMTS coverage areas while it is within a WLAN coverage area. Thus, it can connect to the right UMTS coverage area after leaving a WLAN coverage area, without having to search therefore. These events have no effect on the P2P system, but are just technical. As illustrated in Fig. 4, the number of mobility events pro-

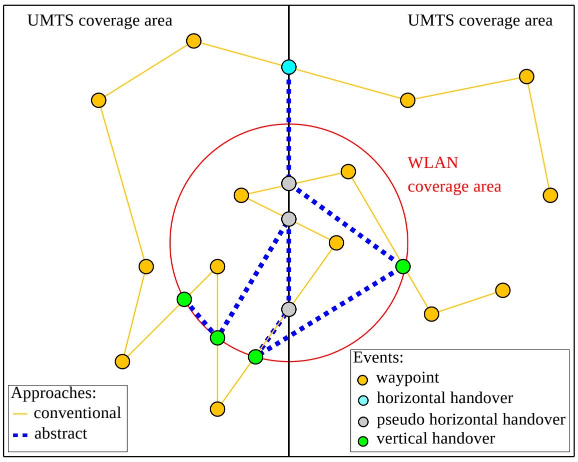


Figure 3: Example of mobility events created during several moves with the conventional and the abstract approach.

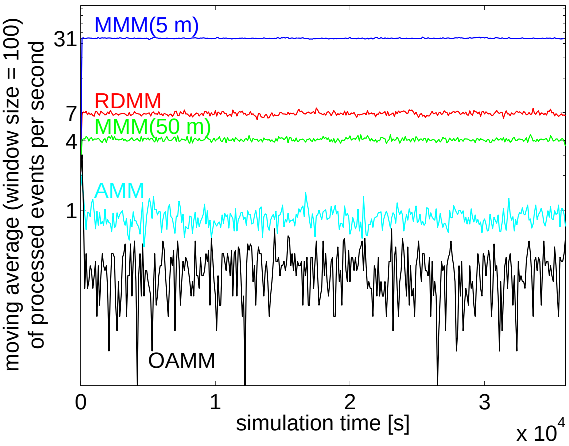


Figure 4: Number of mobility events in logarithmic scale over time based on the *Today's Network Layout*.

cessed per second for both conventional mobility models, *Random Direction Mobility Model* (RDMM) and *Manhattan Mobility Model* (MMM) [18], in combination with the *Today's Network Layout* is at least four times higher than for the AMM and the OAMM – an optimized variant – with corresponding data. We can further see, that the number of processed events is relatively constant in both approaches and all mobility models. The heavy fluctuation for the AMM/OAMM results from the logarithmic scale of the axis of ordinates which lets the standard deviation in the number of events look bigger. In this concrete scenario, the simulation runtime speed-up from 375 s with RDMM to 8 s with AMM. This speed-up is not only issued from the processing of events, but also the calculation of possible new events consumes a lot of computational time. In contrast, the AMM only creates the handover events in Fig. 3, as we describe in the next section. The waypoint events are not needed anymore. In the figurative sense, we move from handover to handover with the AMM.

Also the choice of the mobility model itself strongly influences the number of mobility events in the conventional approach. E.g., the MMM produces the more events per time unit the smaller the distance of two crossing d_{cross} is, as we can clearly see from Fig. 4 for $d_{\text{cross}} = 5\,\mathrm{m}$ and $d_{\text{cross}} = 50\,\mathrm{m}$. After motivating the AMM, we explain how the AMM works in the next section.

3.2.2 Functional Principle

The main ideas behind the AMM are easy to understand, as it basically consists of two parts. In the AMM, a mobile peer is location-unaware as there is no positioning on a simulation plane required. A peer can only perform a single action concerning mobility: switching between different technologies. In the remaining time, it stays within its current technology. Thus, the AMM can be described by a *finite state machine* (FSM). In Fig. 5, this is depicted by a simplified example for two different technologies, namely UMTS and WLAN, which are

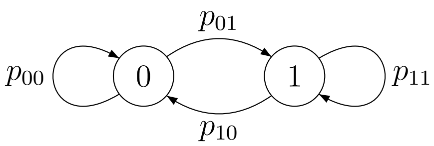


Figure 5: A simplified example of an FSM with state transitions and probabilities for two technologies.

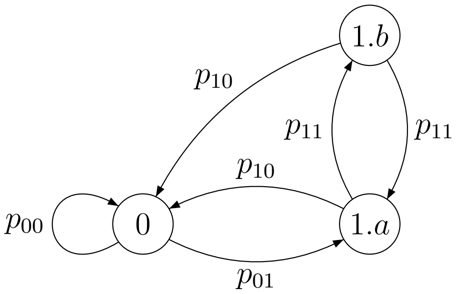


Figure 6: An FSM with state transitions and probabilities for two technologies as used in the simulation model.

abbreviated as 0 and 1, respectively.

For the first part, i.e., the definition of such an FSM, we need the probabilities of the transitions for each possible pair of technologies i and j, $p_{ij} \in [0, 1] \subset \mathbb{R}$, whereby p_{ij} is the probability to get from technology i to technology j. It is

$$\forall i : \sum_j p_{ij} = 1, \tag{1}$$

as migrating to any technology is the certain event. Thus, we can create a *transition matrix* $P = (p_{ij}) \in [0, 1]^{N \times N}$, if N is the number of technologies, e.g., $N = 2$ in Fig. 5. Entering a WLAN coverage area while already being inside another WLAN coverage area is also interpreted as a VHO, thus the FSM in Fig. 6 describes the model used for simulating handovers in our study. Therein, $1.a$ and $1.b$ are abbreviations for two different WLAN coverage areas, and instead of staying within the WLAN technology with probability p_{11}, we switch between these two states, to mark that switching between two WLANs is a VHO.

The second part of the AMM deals with the time a peer stays within a specific technology, which is a state regarding the FSM. This time is usually called the *sojourn time* or *dwell time*. It starts when a peer enters a technology, and it ends when it leaves this technology. The sojourn time can be regarded as the *service time* of a *Markovian* FSM, as described in [19]. Indeed, we can arbitrarily choose the distribution of the sojourn time, and thus it may not necessarily follow a negative exponential distribution. Thereby, the property of memorylessness of the stochastic process is lost. The concept that our FSM matches is known as a *semi-Markov process* as described in [20].

If $p_{00} > 0$, we have a more complex case. Then the sojourn time $\mathcal{S}_{\text{UMTS}}$ for staying within UMTS follows a *compound distribution*, cf. [19]. This compound distribution consists of an *inner* and an *outer distribution*, which here is a *geometrical distribution* for the number of events to switch from one UMTS coverage area to another UMTS coverage area, and the arbitrary distribution of the sojourn time S_{UMTS} within a single UMTS coverage area, respectively. In contrast, the sojourn time of WLAN coverage areas does not “add up”, as we have a VHO between two WLAN coverage areas. Using this compound distribution for UMTS in the AMM, results in the *Optimized AMM* (OAMM). With the OAMM, only events for VHOs

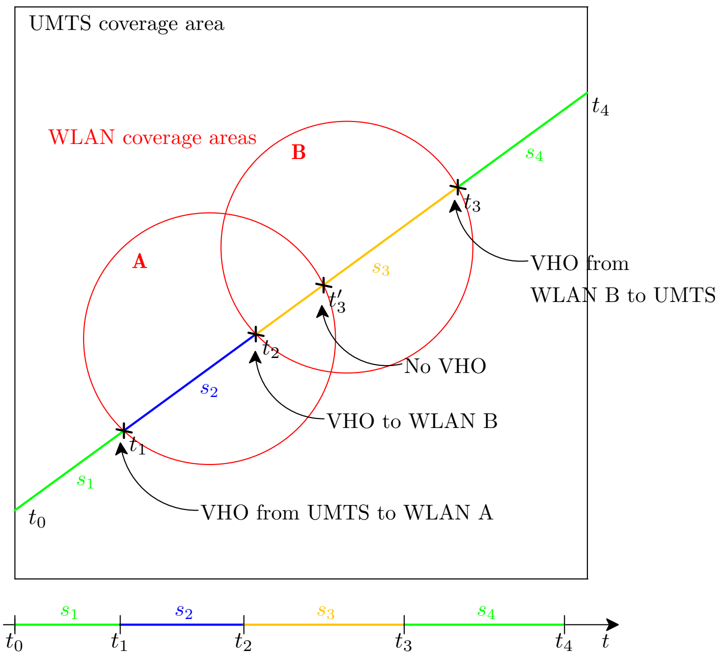


Figure 7: Extraction scheme of sojourn times with UMTS and WLAN coverage areas.

are created. Thus, the number of events with the OAMM is even smaller than for the normal AMM, as illustrated in Fig. 4.

3.2.3 Proof of Concept

We want to show that the AMM returns similar results as revealed with common mobility models, if using appropriate parameters which are derived from the detailed simulation. Therefore, we developed a way to extract the probabilities of the transition matrix and the distributions of the sojourn time per technology from simulation runs with the conventional modeling of mobility.

The extraction and calculation of the transition matrix is as follows. We create a matrix $Q = (q_{ij}) \in \mathbb{N}^{N \times N}$ with $q_{ij} = 0$, $i, j \in \{1, \ldots, N\}$ on initialization. The counter q_{ij} increases by 1, whenever a peer switches from technology i to technology j. This also includes switching between the same technology, i.e. $i = j$. After accumulating all these values in Q, we can compute the transition matrix P by

$$P := \begin{pmatrix} \left(\sum_{j=1}^{n} q_{1j}\right)^{-1} & 0 & \cdots & 0 \\ 0 & \left(\sum_{j=1}^{n} q_{2j}\right)^{-1} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \left(\sum_{j=1}^{N} q_{Nj}\right)^{-1} \end{pmatrix} \cdot Q, \qquad (2)$$

which simply does a row-wise division of Q by the sum of the entries in a row. Albeit, we now have probabilities for switching between technologies, we are still lacking probabilities to decide in which technology a peer is on start-up. Thus, we defined the initialization probability $T_{\text{init}}(i)$ for each technology i to be

$$T_{\text{init}}(i) := \sum_{j=1}^{N} p_{ij} \cdot \left(\sum_{k,l=1}^{N} p_{kl} \right)^{-1} = \frac{1}{N} \cdot \sum_{j=1}^{N} p_{ij}, \tag{3}$$

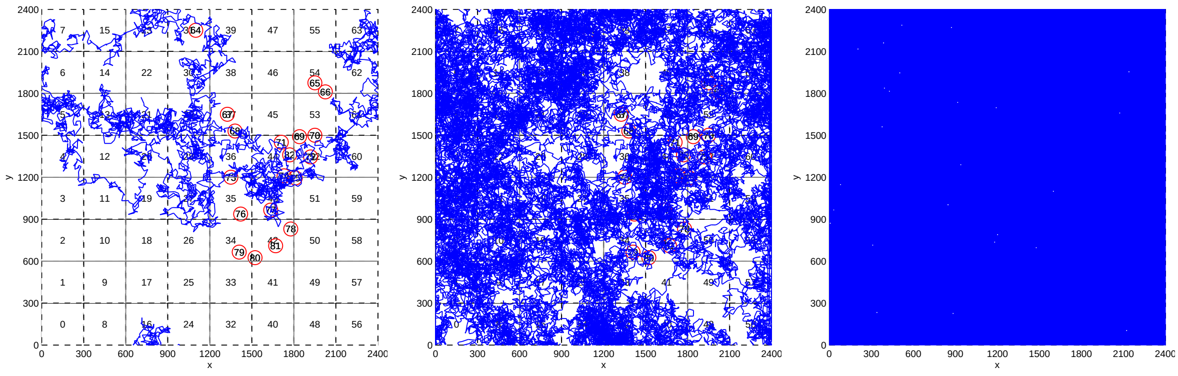
i.e., $T_{\text{init}}(i)$ is the sum of column i, which is the probability to get from any technology to technology i, divided by N, which is the overall sum of all entries of matrix P as of Eq. (1).

The principle of the extraction and calculation of the sojourn times for each coverage area is illustrated in Fig. 7. In this example, for a span of time s_1, the peer is only connected to a UMTS coverage area. Then it enters WLAN coverage area A at t_1 and stays therein for time span s_2. As it enters across WLAN coverage area B at t_2, that intersects with the previous WLAN coverage area, it switches to the new WLAN coverage area, and stays therein for a span of time s_3. Finally, it leaves all WLANs at t_3 and re-enters the UMTS coverage area for s_4, which is the remaining time until t_4, and so on. Note, that at time t'_3 no VHO event occurs. By applying this extraction principle to a simulation run of sufficient simulation time, we get an adequate amount of values for sojourn times for each technology. By sorting these sojourn time values, we have an *empirical CDF* F_i for each technology i, which is implemented as described in [21]. Using these empirical CDFs F_i, we can get new values for the distribution described by the extracted values. Therefore, we use a uniformly distributed random variable to get values y in the interval $[0, 1]$, which are used as arguments for F_i^{-1} to return the quantile of F_i as sojourn time in a coverage area of technology i.

Now, we have all means to extract the probabilities of the transition matrix as well as the distributions for the sojourn times per technology from a simulation run using conventional mobility modeling. To obtain statistically sufficient data, we let a single peer move around for a simulation time of 100 days. Tab. 1 contains the results for the extraction of the transition matrix for the *Random Direction Mobility Model* (RDMM) in combination with the *Today's Network Layout*. The mobility trace of the peer over time is depicted in Fig. 8 for the RDMM. After this rather long time for a simulation run, the peer was at least once almost everywhere in the unit square. To verify this, we discretized the simulation plane into square fields of size 10 m and counted how often the peer moved to each field. Since the mean of these field counters is 8.3 and the corresponding 5 %-quantile is 3.0, we have statistically reliable data. In Fig. 9, the CDFs for the sojourn times of the different technologies are plotted in a logarithmic scale. Thereby, the radius of the circle shaped WLAN coverage areas is 50 m, and the unit length of the square shaped UMTS coverage areas is 300 m. Thus, WLAN coverage areas are significantly smaller than the UMTS coverage areas, with about $7854\,\mathrm{m}^2$ and $90000\,\mathrm{m}^2$,

Table 1: Extracted transition matrix for the *Today's Network Layout* and RDMM.

RDMM 5 - 50 m		to	
		UMTS	WLAN
from	UMTS	$p_{00} = 0.926$	$p_{01} = 0.074$
	WLAN	$p_{10} = 0.961$	$p_{11} = 0.039$



(a) 1 day (b) 10 days (c) 100 days

Figure 8: Trace of a single moving peer after certain simulation times.

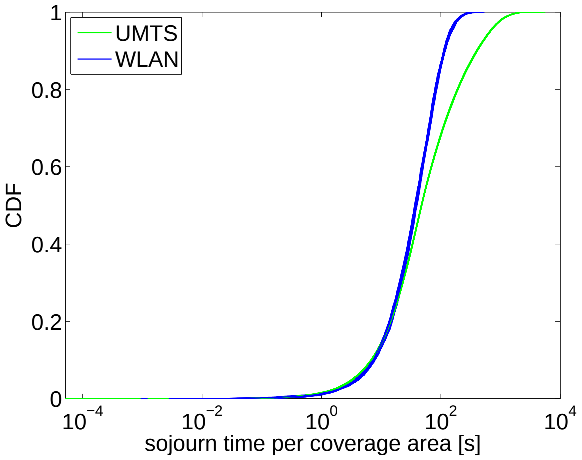


Figure 9: Sojourn time CDFs extracted from the *Today's Network Layout*.

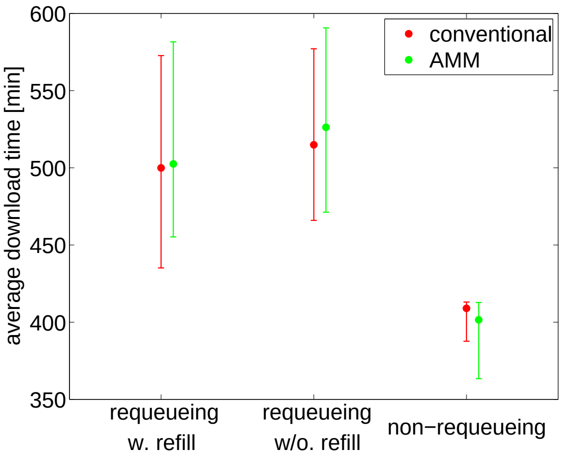


Figure 10: Comparison of conventional and abstract mobility modeling.

respectively. Furthermore, there are some overlapping areas for WLANs which truncate the sojourn times of WLAN coverage areas. Hence, we expected that the sojourn times of UMTS, the green curve, are longer as those of WLAN, the blue curve. To increase the statistical reliability, we performed 20 repetitions of each run with different initial random seeds. Tab. 1 shows the mean of 20 runs. Also in Fig. 9, we plotted 20 curves for each technology.

With these extracted data, we did simulation runs with the AMM, to compare the AMM with the conventional modeling of mobility. Since a mobile P2P system is a highly dynamical, we exemplary plotted the intervals, that contain the means of all 20 runs for a single seed, FRIAT of 80 s, and three VHO methods for the AMM in Fig. 10. The dot plotted within each interval represents the mean of all 20 runs. We see that the results are similar, i.e. the AMM generates similar results compared to conventional mobility modeling. This is substantiated by the size of the intervals that is caused by the random fluctuations of a mobile P2P system.

3.3 Effect of VHOs on P2P File Sharing

In this section, the effect of VHOs on a P2P file sharing application is explained. When a moving user performs a VHO, the transmission of application data is stopped for a certain delay Δt_{VHO}. Registering to a new access technology may lead to a change of the current IP address and break the peer's ongoing upload and download connections. On application layer, the dropped connection of a downloading peer to some providing peers re-schedules the request of the download, i.e., the downloading peer P is requeued in a providing peer's waiting queue.

In addition, a peer P performing a VHO might serve as a providing peer. The IP address change results in lost connections and the peers served by P need to rediscover P by asking the index server for new sources of a file. In standard eMule implementation, this is done periodically every ten minutes. In the following, we will refer to this technique as *requeueing w/o refill*.

An alternative method is called *requeueing with refill*. It introduces a minor modification of the peer's cooperation strategy to improve the system performance and utilizes the fact that a providing peer knows all peers in its uplink waiting queue before and after the VHO. Thus, the providing peer simply reidentifies itself at the served peers with its new IP address and invites them to continue the download. Thus, it can speed up the recovering after a VHO.

Previously, we focused on the situation that a VHO implies an IP address change. However, approaches like MobileIP preserve the peer's IP address and allow TCP connections to continue after the VHO. These mechanisms lead to an additional delay Δt_{MIP} which we assume to be static. On application layer, a peer keeps its current connections running which means that it also maintains the position in the uplink waiting queue or is still served. However, the total transmission delay during which no application data is exchanged is now $\Delta t_{\text{VHO}} + \Delta t_{\text{MIP}}$. Such a mechanism is denoted as *non-requeueing technique*. Since the VHO delay Δt_{VHO} can be assumed to be rather small, especially compared to Δt_{MIP}, we fix $\Delta t_{\text{VHO}} = 100\,\text{ms}$ from now on.

In this paper, we want to answer the question if P2P file sharing applications in B3G networks require non-requeueing techniques like MobileIP or not. In detail, we compare a) requeueing and b) non-requeueing with refill, and c) non-requeueing w/o refill for different additional delays for preserving the IP address. The investigated scenarios comprise a today's and a future network layout in different load situations. In addition, we derive a new cooperation strategy in order to counter the impact of mobility. Its performance is demonstrated for the future layout using a non-requeueing VHO technique.

4 Analysis of Current P2P in B3G Networks

We want to analyse the impact of VHOs on a P2P network of these days. The simulation scenario for our investigations in this section is as follows. There is a single file of size 9500 kB which is to be shared by the mobile peers. There are 100 mobile peers that want to download this file and altruistically share this file after download. Every 120 s, a random peer sends a request to the sources currently available for this file until all peers have placed their request. At the beginning, the P2P network consists of a number of *Internet peers* with a

Table 2: Average download times [min].

average download time [min]		today		future	
		high	low	high	low
non-req. Δt_{MIP}	0 s	175.5	3.5	80.6	2.3
	1 s	175.6	3.5	82.7	2.3
	5 s	176.7	3.6	102.9	2.5
	10 s	181.7	3.6	145.3	2.8
	100 s	–	–	1074.5	14.5
req.	with refl.	246.7	4.1	1206.8	6.5
	w/o refl.	266.7	4.1	1486.2	6.6

constant uplink capacity of 768 kbps that serve as initial sources, and keep serving throughout the simulation. This ensures that the mobile peers always find equal conditions on simulation start-up. The number of these initial seeds controls the load of the P2P system. Few Internet peers lead to a high load since at the beginning there are only few sources available, the first downloads may take a long time, and the file propagates only slowly. All stochastic influences except for the mobility pattern are avoided so the impact of VHOs is not tampered by stochastical fluctuations not caused by mobility itself. Especially, there is no churn in this scenario. For the same reason, we kept to a single set of parameters defining the network and traffic. We performed 20 repetitions with different seeds for the random number generator in every simulation run.

We measure the impact of VHOs on the P2P system by the download times of all mobile peers individually. The download time of a single peer is defined by the period of time between sending the file request and receiving the last data belonging to the file. We illustrate the impact of VHOs by plotting the cumulative distribution function (CDF) of the download time of a random peer. We obtain the CDF from the download times of all peers in all simulation runs with different seeds. For our analysis, we consider four scenarios: today's network with a low load, today's network with a high load, a future network with a low load, and a future network with a high load. A high load corresponds to a single Internet peer and a low load to ten Internet peers.

In Fig. 11, we see different CDFs of the download times for today's networks in the high load situation. There are four CDFs for non-requeueing with delays Δt_{MIP} of 0 s, 1 s, 5 s, and 10 s, and two CDFs for requeueing, one with refill and one w/o refill. Preserving the IP address outperforms loosing the IP address in this high load situation. A peer that looses its IP address is forced to reenter the uplink waiting queues of its sources and therefore has to wait much longer until it is allowed to download for the next time. There is no clear impact of the non-requeueing delay even if the non-requeueing delay is extremely high, since there simply are too few VHOs in today's network layout. The low load scenario in today's network nullifies the impact of the different IP address handling mechanisms, since even less VHOs occur during the shorter download time in this scenario, and the waiting queues are almost empty. Thus, the average download times are nearly the same, cf. Tab. 2. Tab. 2 shows also the average download time of today's and future network layout in the low and high load

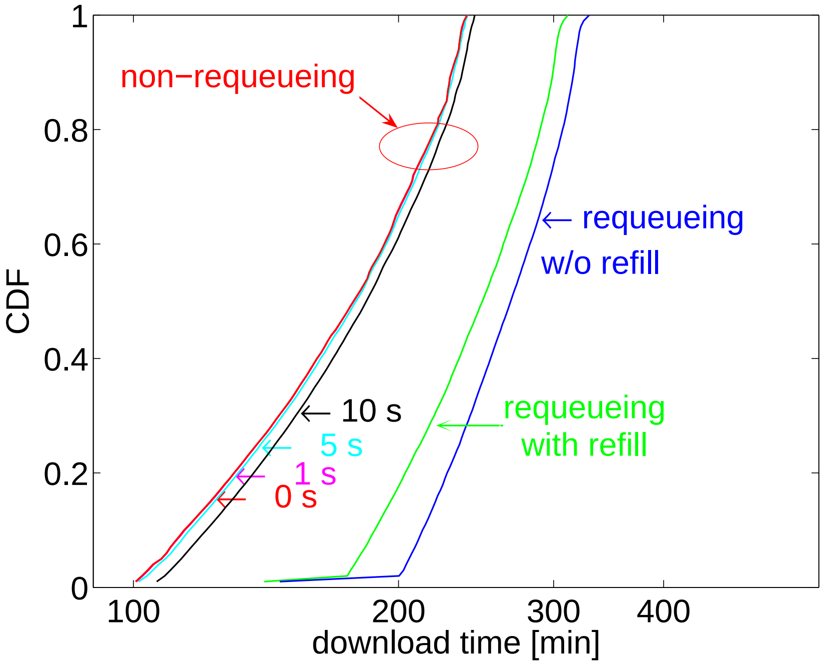


Figure 11: Comparison of requeueing and non-requeueing with varying delay in today's networks with a high load.

scenario for non-requeueing and requeueing.

Let us next investigate the situation in future networks with higher WLAN hotspot density. Fig. 12 and Fig. 13 show CDFs for requeueing with and w/o refill as well as CDFs for non-requeueing with delays of 0 s, 1 s, 5 s, 10 s, and 100 s in the future network layout. Fig. 12 shows the results for the high load scenario. Analogous to the results from Fig. 11, non-requeueing is better than the two requeueing variants, but the difference between requeueing and non-requeueing increased from a factor of two in today's network layout to a factor of ten in the future network layout. The higher WLAN density in the future layout has two effects, a higher network capacity and more VHOs. The higher available amount of bandwidth leads to an average download time of 82.7 min in the future layout compared to 175.6 min in today's layout for the non-requeueing technique with $\Delta t_{\text{MIP}} = 1\,\text{s}$. However, the higher number of VHOs in the future layout increases the relative impact of the non-requeueing delay, compared to $\Delta t_{\text{MIP}} = 0\,\text{s}$, expressed by larger differences in download times, cf. Tab. 2.

Using the requeueing technique, the peer changes its IP address at every VHO. Thus, it is often loosing its connections, is removed from being served, and shifted back to the end of the waiting queue. Together with frequent VHOs, this technique has to be avoided for an efficient content distribution service in a future network layout. Only for unrealistic VHO delays of 100 s, the requeueing and the non-requeueing technique show the same download performance in a high load scenario as can be seen in Fig. 12.

In the following, we focus on the low load scenario in future networks for which Fig. 13 shows the equivalent CDFs as in Fig. 12. We can still see a difference in requeueing and non-

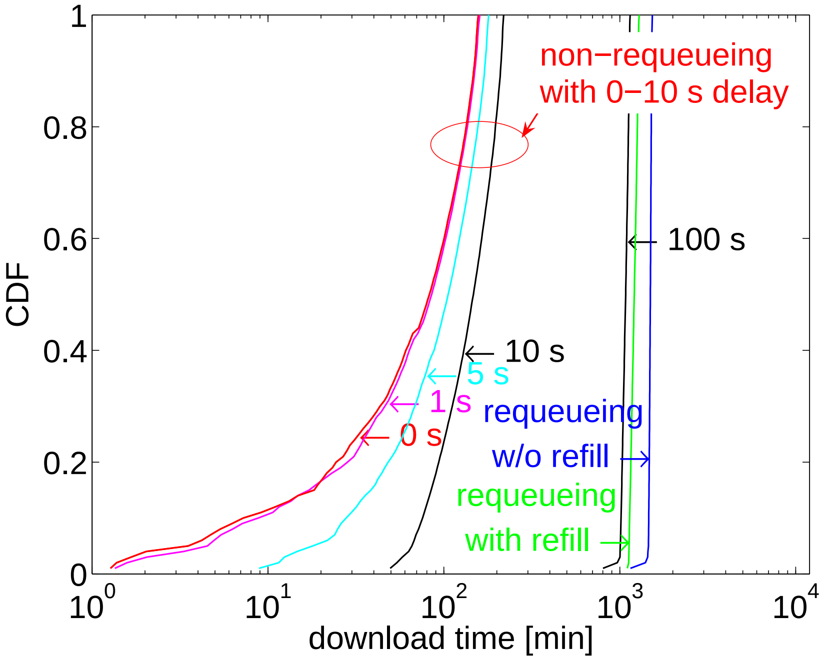


Figure 12: Comparison of requeueing and non-requeueing with varying delay in the future network with a high load.

requeueing as well as the non-requeueing delays even with a low load as opposed to today's network layout, cf. Tab. 2, since more VHOs occur even in the shorter download times. If the load in the P2P system is low, then downloads take less time which leads to less VHOs occurring during the downloading time. In general, the impact of mobility decreases with the load and vice versa.

In both load scenarios, preserving the IP address with non-requeueing outperforms requeueing techniques. Nevertheless, the performance gain of non-requeueing melts in the low load scenario, since the waiting queues at the providing peers are almost empty and hence the waiting times are almost negligible. In such a scenario, a delay Δt_{MIP} exists such that the download performance is even worse than with requeueing techniques. However, this only happens for unrealistic large delays above 10 s.

As a result of the performance evaluation, we see that non-requeueing techniques, like MobileIP, are recommended in mobile P2P file sharing systems w.r.t. download performance, if this technique only requires a small transmission delay below a few seconds. In future network layouts, the increased uplink capacity due to the higher WLAN density leads to smaller download times. In order to foster the download from such high-capacity peers, a new cooperation strategy is proposed in the next section which tries to smoothen changes in the available uplink capacity as a consequence of the user's mobility and the resulting VHOs.

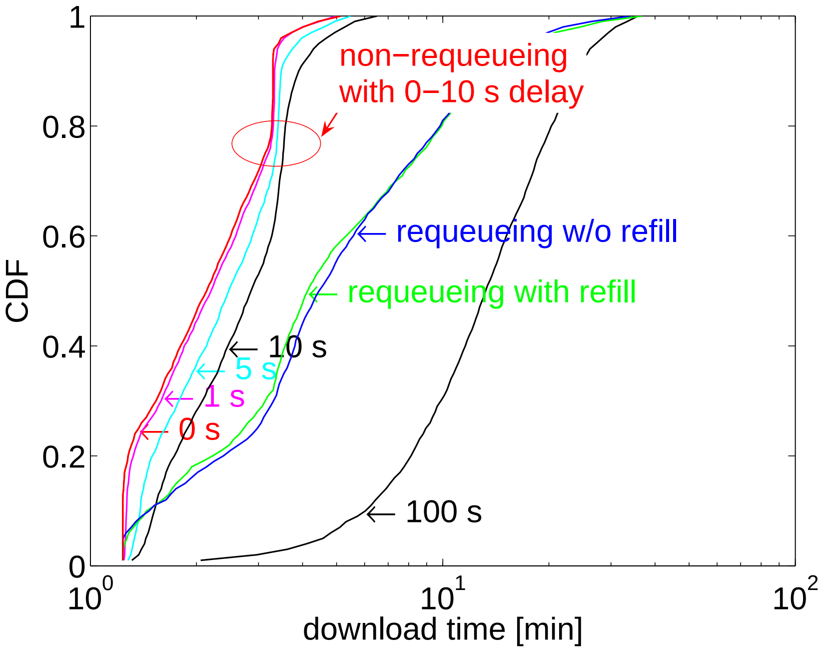


Figure 13: Comparison of requeueing and non-requeueing with varying delay in the future network with a low load.

5 Mitigating the Impact of Mobility

In this section, we introduce a new cooperation strategy that affects the duration a user is allowed to access the uplink capacity of a providing peer. In common P2P networks like *eDonkey*, the resource exchange is volume-based, i.e., each peer is allowed to download the same amount of data in a row, also called *download unit* (DU). In *eDonkey* P2P systems, the size of such a DU is 540 kB corresponding to three blocks. We will further speak of *volume-based cooperation* (VBC). The problem of VBC is that a peer with a high-capacity technology, like WLAN, is thwarted by peers with smaller bandwidths, like UMTS, if these peers wait to be served by the same source. Thus, a user in a high-capacity technology can not finish its download quickly and serve as a new seed for all other peers in the network.

As a small example, we imagine three peers P_0, P_1, P_2 with download capacities C_0, C_1, C_2. The ratio of the corresponding downlink capacities may be $C_0 : C_1 : C_2 = 3 : 2 : 1$. If the peer with the highest capacity, i.e., P_0, takes a download time Δt to download a DU, then it takes $2 \cdot \Delta t$ for P_1 and $3 \cdot \Delta t$ for P_2. If these three peers start downloading at the same time from the same source, then P_0 will have to wait for $5 \cdot \Delta t$, i.e., the time P_1 and P_2 are served until P_0 is served next. Thus, it is thwarted by these two peers and the P2P network can not fully profit by its higher capacities. As a consequence, the whole content distribution process is slowed down.

Our new approach avoids this thwarting by not restricting the amount of data, but the time a peer is allowed to download in a row. This approach is called *time-based cooperation* (TBC). Thus, peers with a higher capacity will serve earlier as new sources, since they are able to

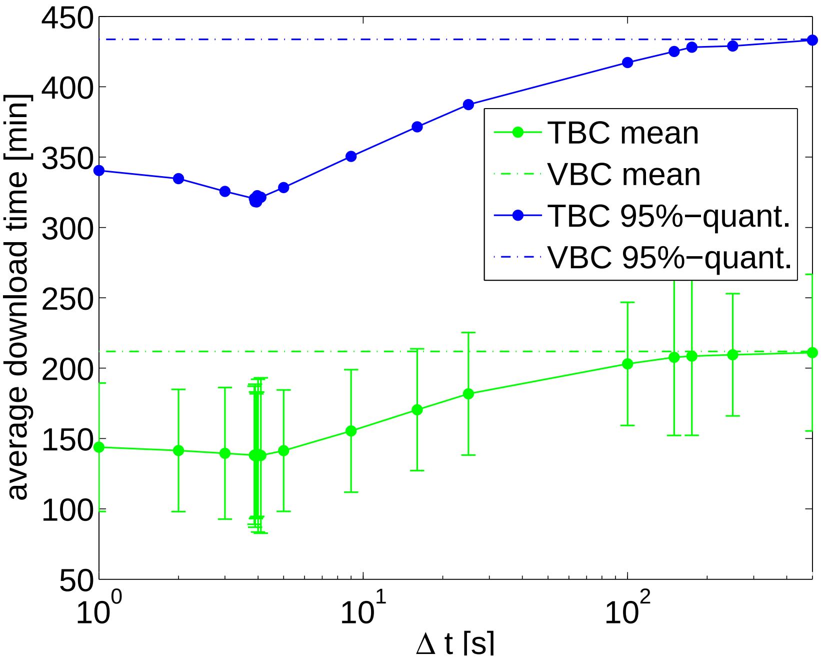


Figure 14: Download performance of a P2P system with VBC and TBC approach.

download more data in the same time. Alas, the effectiveness of this approach heavily relies on the peers' altruism assumed in this work. The basic principle of this TBC approach is a time-out Δt which is the maximum time a user is allowed to download from a providing peer. Additionally, we still need a limitation of transferred volume, since MSD needs a reservation mechanism for the data currently downloaded to prevent downloading data twice. We set this limit to be $V = 540\,\mathrm{kB}$. The providing peer stops serving the downloading peer if either the time Δt is spent or the volume V is uploaded. In particular, the downloading peer is interrupted after time $\Delta t' = \min\{\Delta t, \Delta t_V\}$ while Δt_V is the duration a peer needs to download V. Note, that Δt_V might vary due to VHOs of the downloading or uploading peers.

For the analysis of TBC, we consider the following scenario which makes greater demands on optimisation. There are 100 mobile peers which move around in the future network layout. There is a total of 20 different files, each of size 9500 kB. On average, each peer shares one file at the beginning. The peers want to download all remaining files they not already have, i.e., 19 files on average. The interarrival time between two file requests is exponentially distributed with a mean $\mu_F = 40\,\mathrm{s}$. Additionally, we consider churn here. The peers switch from online to offline with exponentially distributed lengths of the online and offline phases, each with a mean $\mu_C = 1\,\mathrm{h}$.

Fig. 14 shows the average download time and the 95% quantile of the download time of the VBC and TBC approach. The latter's performance depends on the choice of the time Δt, a peer is allowed to download. The figure illustrates that the performance of TBC is always at least as good as of VBC. We see that the larger Δt the smaller is the performance gain. This results from the peers with fast technologies having to wait the longer on peers in slower technologies the larger Δt. We can see that there is an upper bound for Δt beyond which

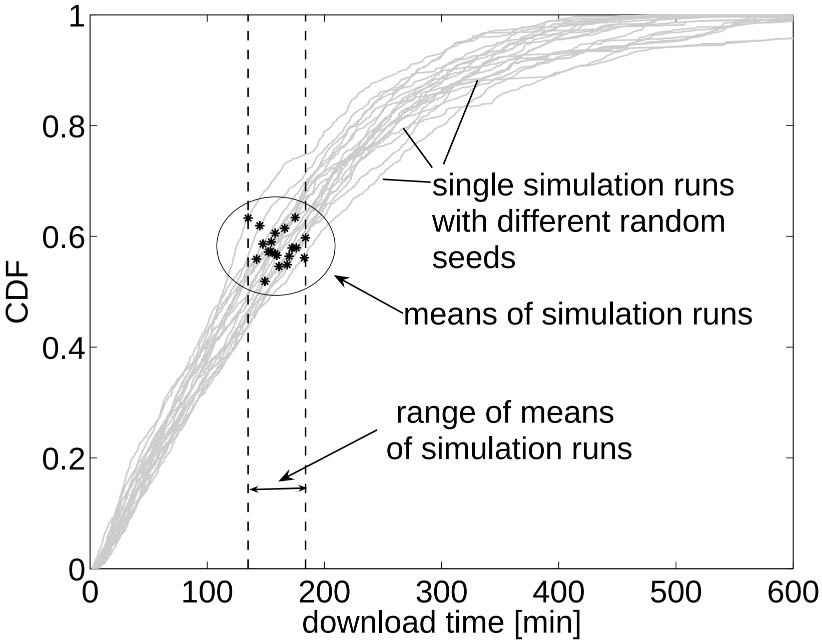


Figure 15: Fluctuations in an exemplary run with time-based approach for $\Delta t = 4\,\mathrm{s}$.

the two approaches give the same results since even a peer in the slower technology is able to finish its download before the time-limit is exceeded.

Fig. 14 suggests that there is an optimal value of the allowed download time, roughly at $\Delta t = 4\,\mathrm{s}$. However, the size of the 95% confidence intervals of the average download times, indicated by error bars in Fig. 14, is quite large. Hence, it's difficult to find an optimum. This results from the fact that we are investigating a highly dynamical and complex system. The behavior of such a system can vary largely depending on small changes in the overall situation as, e.g., a peer that stayed within WLAN for a longer time and/or became a new seed for a file faster. To emphasize this dynamic, we separately investigate the download performance for all 20 runs of the same scenario with $\Delta t = 4\,\mathrm{s}$, only varying the seeds of the random number generator. Fig. 15 depicts the CDF of the download time for each simulation run with a different random number generator seed. In addition, the black dots on each CDF indicate the average download time of this single simulation run on the x-axis and the corresponding quantile on the y-axis. The mean of these 20 average download times corresponds to a single point in Fig. 14, while the range of the average download times of the 20 simulation runs is responsible for the large confidence intervals in Fig. 14. All means lie within an interval of a length of around 70 min which manifests the dynamic character of this system.

A second relevant aspect of P2P systems is fairness, i.e., whether all peers are treated equally. We use the *fairness index* f_I introduced in [22], defined as $f_I := \frac{1}{1+c_x^2}$, where c_x is the coefficient of variation of the download time. The fairness index returns values between 0 and 1. A fairness index of 1 means all peers experience the same download time, while lower values indicate a more unfair system. Fig. 16 shows the fairness index of the download time

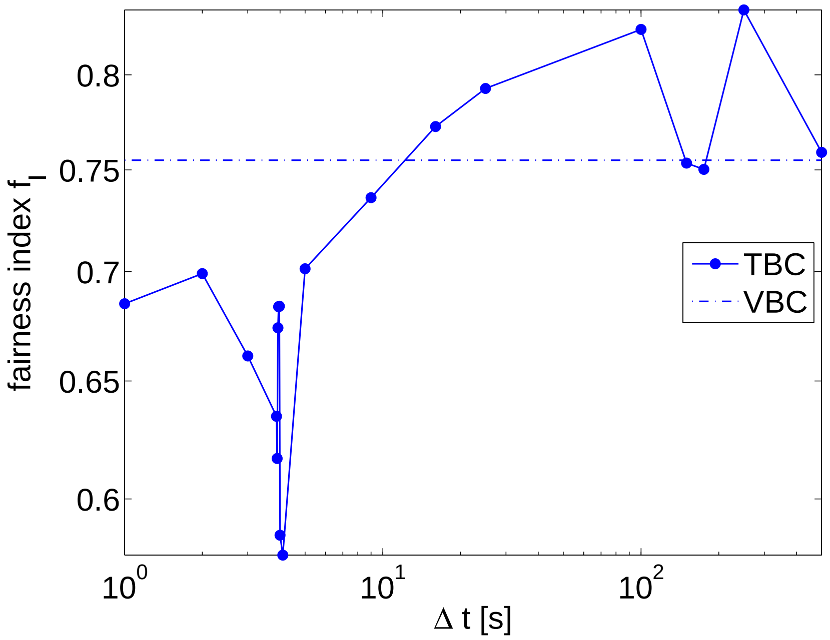


Figure 16: Fairness index f_I of a P2P system with VBC and TBC approach, respectively.

for VBC as well as TBC in dependence of Δt. The figure reveals that the fairness is lowest if the performance of TBC is best. This is due to high-capacity peers being preferred by TBC and being able to download more data in the same time.

6 Conclusion

We investigated the impact of VHOs on the download performance of a P2P-based content distribution system in different heterogeneous networks and load situations. We considered that VHOs can lead to a loss of the IP address and investigated the impact of this loss as well as the use of a mechanism to preserve the IP address beyond a VHO at the cost of additional transmission delays. Non-requeueing techniques, like MobileIP, are recommended in such mobile P2P file sharing systems w.r.t. download performance, if this technique only requires a small delay below a few seconds. When not using non-requeueing techniques, an IP address change implies being requeued in the uplink waiting queue of providing peers which increases the overall download time due to the higher waiting times. This harms the distribution process of contents in the whole network.

Nevertheless, an upper bound for the additional transmission delay of such IP preserving mechanisms exists. In a low load scenario, a peer will be served relatively fast after being requeued and the additional delay of non-requeueing might outweigh the waiting times of requeueing techniques.

In future network layouts, the increased uplink capacity, e.g., due to better WLAN coverage, leads to smaller download times. In order to foster the download from such high-capacity peers, a new cooperation strategy is proposed for a content distribution system based on multi-source download. Instead of downloading individual blocks of a file, a user gets a certain time slot at a providing peer. We have shown that this time-based cooperation strategy increases the download performance of the P2P system in the considered heterogeneous wireless environment. Of course, the fairness of the system is decreased as the high-capacity peers, like WLAN users, are preferred and are allowed to download more data in the same time than peers with small access bandwidths. The investigation of this approach revealed that the high complexity and the dynamic character of an MSD P2P-based content distribution system with mobile users make it hard to quantitatively describe this system. Hence, we focused on a qualitative description instead.

References

[1] ipoque GmbH, “P2P Survey 2006.” Whitepaper. Available at http://www.ipoque.com/pub/P2P-Survey-2006.pdf.

[2] A. Parker, “The true Picture of Peer-to-Peer File Sharing,” Aug. 2006. Online available at http://www.cachelogic.com/home/pages/research.

[3] T. Hoßfeld, K. Tutschku, and F.-U. Andersen, “Mapping of File-Sharing onto Mobile Environments: Enhancement by UMTS,” in *Mobile Peer-to-Peer Computing MP2P, in conjunction with the 3rd IEEE International Conference on Pervasive Computing and Communications (PerCom’05)*, (Kauai Island, Hawaii), Mar. 2005.

[4] T. Hoßfeld, K. Tutschku, F.-U. Andersen, H. de Meer, and J. Oberender, “Simulative Performance Evaluation of a Mobile Peer-to-Peer File-Sharing System,” in *Next Generation Internet Networks NGI2005*, (Rome, Italy), Apr. 2005.

[5] R. Steinmetz and K. Wehrle, eds., *Peer-to-Peer Systems and Applications*. LNCS 3485, Berlin: Springer, 2005.

[6] “1st International Workshop on Decentralized Resource Sharing in Mobile Computing and Networking.” Los Angeles, USA, Sep. 2006. Information available at http://www.mobishare.org.

[7] “3rd IEEE International Workshop on Mobile Peer-to-Peer Computing (MP2P’06).” Pisa, Italy, Mar. 2006. Information available at http://www3.informatik.uni-wuerzburg.de/mp2p2006.

[8] T. Heer, S. Gotz, S. Rieche, and K. Wehrle, “Adapting Distributed Hash Tables for Mobile Ad Hoc Networks,” in *PERCOMW ’06: Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops*, (Washington, DC, USA), 2006.

[9] O. Landsiedel, S. Gotz, and K. Wehrle, “Towards Scalable Mobility in Distributed Hash Tables,” in *P2P ’06: Proceedings of the Sixth IEEE International Conference on Peer-to-Peer Computing*, (Washington, DC, USA), pp. 203–209, IEEE Computer Society, 2006.

[10] S. Zöls, R. Schollmeier, W. Kellerer, and A. Tarlano, “The Hybrid Chord Protocol: A Peer-to-Peer Lookup Service for Context-Aware Mobile Applications.,” in *Networking - ICN 2005, 4th International Conference on Networking*, (Reunion Island, France), Apr. 2005.

[11] J. Biström and V. Partanen, “Mobile P2P - Creating a mobile file-sharing environment,” Tech. Rep. HUT T-111.590 Research Seminar on Digital Media, Telecommunications Software and Multimedia Laboratory, Helsinki University of Technology, 2004.

[12] J. O. Oberender, F.-U. Andersen, H. de Meer, I. Dedinski, T. Hoßfeld, C. Kappler, A. Mäder, and K. Tutschku, “Enabling Mobile Peer-to-Peer Networking,” In Mobile and Wireless Systems, LNCS 3472, (Dagstuhl, Germany), Springer, Jan. 2005.

[13] F. L. Piccolo, G. Neglia, and G. Bianchi, “The Effect of Heterogeneous Link Capacities in BitTorrent-Like File Sharing Systems,” *hot-p2p*, vol. 00, pp. 40–47, 2004.

[14] A. A. Hamra and P. A. Felber, “Design choices for content distribution in P2P networks,” *SIGCOMM Comput. Commun. Rev.*, vol. 35, pp. 29–40, Oct. 2005.

[15] D. Schlosser, T. Hoßfeld, and K. Tutschku, “Comparison of Robust Cooperation Strategies for P2P Content Distribution Networks with Multiple Source Download,” in *Sixth IEEE International Conference on Peer-to-Peer Computing (P2P2006)*, (Cambridge, UK), Sept. 2006.

[16] P. Garbacki, A. Iosup, D. H. J. Epema, and M. van Steen, “2Fast : Collaborative Downloads in P2P Networks.,” in *Peer-to-Peer Computing*, pp. 23–30, 2006.

[17] T-Mobile, “T-MAP HotSpot Locator,” Sept. 2006. Online available at http://www.t-mobile.de/hotspot/locator/.

[18] C. Bettstetter, “Mobility Modeling in Wireless Networks: Categorization, Smooth Movement, and Border Effects,” *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 5, pp. 55–66, July 2001.

[19] P. Tran-Gia, *Einführung in die Leistungsbewertung und Verkehrstheorie*. München, Deutschland: Oldenbourg, 2nd ed., July 2005.

[20] J.-K. Lee and J. C. Hou, “Modeling Steady-state and Transient Behaviors of User Mobility: Formulation, Analysis, and Application,” in *MobiHoc ’06: Proceedings of the seventh ACM international symposium on Mobile ad hoc networking and computing*, (New York, NY, USA), pp. 85–96, ACM Press, May 2006.

[21] A. M. Law and D. W. Kelton, *Simulation Modelling and Analysis*. McGraw-Hill Education, 3rd ed., April 2000.

[22] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, “A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System,” Tech. Rep. DEC-TR-301, Eastern Research Lab, Digital Equipment Corporation, 77 Reed Road, Hudson, MA 01749, Sept. 1984.