A Parallelized File-Transfer-Protocol for On-Board IP Networks

Shaleeza Sohail*, Salil S. Kanhere*, Sanjay Jha*,†, Adeel Baig*,†, Muhammad Ali Malik*,†
*University of New South Wales, 2052, Sydney, Australia
†National ICT Australia, 1430, Sydney, Australia
Email:- (sohails,salilk,sjha,abaig,mamalik)@cse.unsw.edu.au

***Abstract*— On board IP networks are an effective mean to provide uninterrupted Internet access to public transport passengers. Such networks require optimal resource allocation due to the scarcity and high cost of network resources. The parallelized file transfer protocol (P-FTP) is a novel network resource aware parallel technique for improving file transfer performance for the clients connected with wired networks. An extension of P-FTP approach, Mobile P-FTP is an efficient file transfer technique for the users of on-board networks. The simulation study shows that by virtue of its self-tuning capability, such technique exhibit improved performance for on-board IP network users as compared to a tradition file transfer approach.**

I. Introduction

Due to the recent advancements in the paradigm of pervasive connectivity, there has been a growing interest in introducing broadband Internet services to public transport passengers [2], [3]. The aim is to provide communication and information access anywhere and anytime. To this end, it is envisaged that high speed Local Area Networks (LANs) will be deployed on-board Public Transport Vehicles (PTVs). These on-board LANs will be connected to the global Internet via on board Mobile Routers (MRs) and wide area wireless access systems (e.g.: GPRS, UMTS , WLAN or Satellite) as shown in Figure 1. Mobility of the entire on-board network is managed transparently by the MR, using Network Mobility (NEMO) protocol [1], one such network is shown in Figure 2.

The routes of PTVs are repetitive and known apriori. The MR is capable of: 1) admission control, 2) performing active resource reservation from its current location, and 3) advance resource reservation for future locations [4]. All of these features can be leveraged to predict the available access link bandwidth at future locations of the PTV. Our focus is not the specific techniques for prediction and forecasting of available resources for on-board LANs, which remains a subject of ongoing research. We assume that the MR is capable of calculating the available access link bandwidth in the near future with a medium degree of accuracy. In this paper, our particular focus is to study how this information can be exploited to improve protocols and applications for better resource utilization.

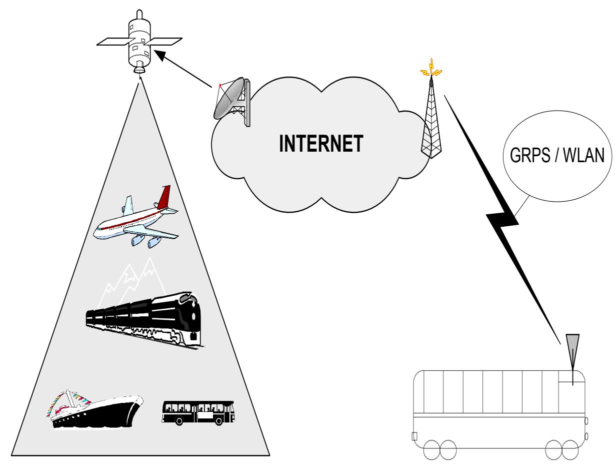

Fig. 1. Providing Internet Services to users of PTVs

Paraload [8] is a technique that proposes the aggregation of bandwidth from different interfaces in devices that are equiped with multiple interfaces for Internet access. Mobile Access Router (MAR) [9] is an application based on paraload approach which provides a bandwidth aggregation on the Internet connection for the users of PTVs and seamless handoffs between different wireless access technologies. In this paper we propose an application for effective file transfer based on MAR.

File transfer is one of the most commonly used application over the Internet today. We would also expect users of on-board communication systems to frequently download files from the Internet. Even on fixed communication links, a large file download can take considerable time due to the changes in traffic conditions in the network. Since wireless links generally exhibit higher loss rates, the download duration for on-board users can be considerably longer.

In [5], we have proposed a parallel downloading technique, called Parallelized File Transfer Protocol (P-

0-7803-8887-9/05/$20.00 (c)2005 IEEE

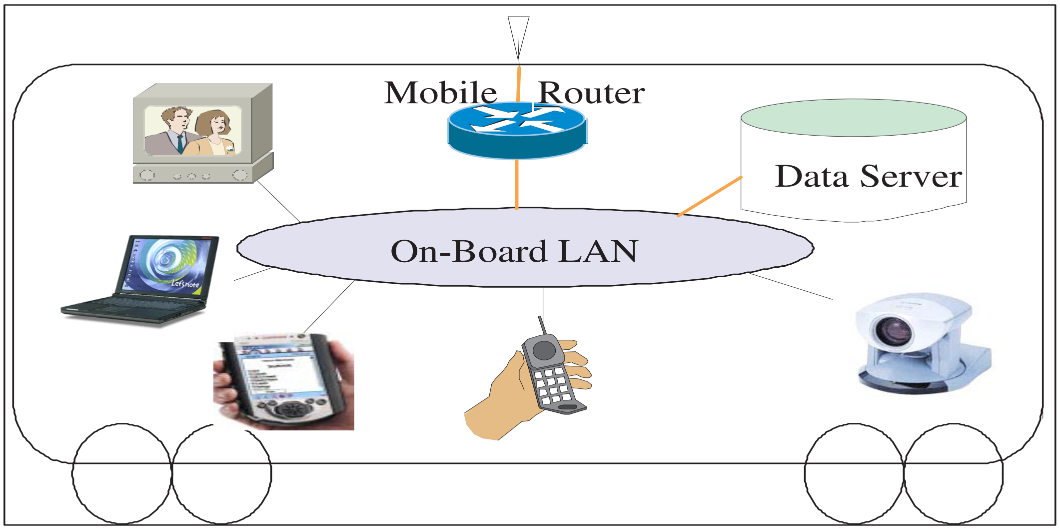

Fig. 2. On-Board IP Network

FTP) for reducing the download time of large files on wired networks. Our technique proposes the use of multiple download sessions with different FTP servers to download disjoint portions of a file in parallel. The P-FTP approach is based on the idea that the parallel use of available resources may enhance performance for resource intensive applications such as large file transfers. However, an important distinction between P-FTP and similar parallel download approaches proposed elsewhere, is that P-FTP considers resource availability in the network as well as at the mirror servers, before starting the file sessions. Moreover, our approach can dynamically adapt to the changing conditions of the servers and the network during the file transfers.

In this paper we discuss a novel extension of the P-FTP approach called Mobile P-FTP, which can effectively reduce the transfer delays of large files for the users of on-board LANs. By configuring the P-FTP server at the MR, the P-FTP clients of on-board networks can be informed about the increase in resource availability. During those intervals the clients can start additional file transfer sessions to accelerate file transfer process and download file in reduced time. We have carried out a simulation based study to show that the mobile P-FTP clients perceive better performance than a tradition file transfer clients when connected to an on-board IP network.

The rest of the paper is organized as follows: Section II summarizes the P-FTP approach in a nutshell. Section III provides details of the P-FTP extension for on board networks. Section IV describes the simulations carried out to study the performance of mobile P-FTP clients. Section V concludes the paper and provides some futire work options.

II. Parallelized File Transfer Protocol

The details of the P-FTP approach are published in [5], [6]. In this section, we present a brief overview of this approach. Interested readers can refer to [5], [6] for further details.

The P-FTP approach involves having a special P-FTP server running within the client's Autonomous System (AS). This server is capable of communicating with the file servers located anywhere in the Internet and of measuring network parameters. As opposed to traditional FTP, P-FTP client first contacts the P-FTP server when a particular file is to be downloaded. The P-FTP server then selects a set of suitable file servers from which the file can be downloaded in parallel. This information is sent back to the client which then initiates parallel file transfer sessions. In addition to this, the P-FTP server also sends a list of *backup* file servers that will be contacted by the client if there is congestion in the network or some particular file server slows down. This makes our technique capable of dynamically adapting to the changing network conditions.

The P-FTP system selects the file severs based on their utilization and the network path quality. A server's utilization depends on the server load, which includes its CPU and memory utilization. The path quality is determined by such parameters as available bandwidth, round trip time and packet dropping probability. However any other network-related characteristic can be used. A database of these parameters is maintained and regularly updated in order to improve the P-FTP server's response time. All available servers are ranked based on these parameters. The P-FTP server selects a subset of these file servers for the file transfer. The selection limitation is the available bandwidth between the client and its Internet gateway. A number of file servers are selected in such a way that the aggregate P-FTP flow does not produce congestion at the clients access link. Given this information, the P-FTP server determines the respective file portions that can be loaded from the selected file servers. This information is then sent to the client.

The client is able to dynamically adapt to changing network conditions. We focus on two such conditions: congested common links and the file server's throughput. Our client can detect servers that are sharing a common congested link on their path to the client. The connections to such servers are terminated upon successful detection and new connections with the *backup* servers nominated by the P-FTP server are initialized. Moreover, the throughput of the file servers, relative to themselves, is regularly monitored to detect low-performing file servers. The file portions to be downloaded from slow servers is reduced. The client starts new sessions with the *backup* servers to download the remaining file portions. This approach has two additional advantages:

- It distributes the load by downloading large files in portions from different servers. Due to the dynamic adaptation ability it reduces the congestion in the Internet.
- The download time does not vary significantly when the network conditions change.

The P-FTP approach for wired networks has been tested on the Internet using the Planet-Lab infrastructure [7] and the results are presented in [6]. The empirical results show at least 50% improvement in download time when compared to a traditional file transfer approach. Moreover, download time measured with the P-FTP approach shows significantly less standard deviation than download time measured with a traditional single server file transfer approach. The small standard deviation of download time for the P-FTP approach is another advantage of the approach which enables files to be downloaded in a predictable range of time.

III. Mobile P-FTP

Mobile P-FTP is an extension of P-FTP approach for wireless network users. In this paper due to space limitations, we are only discussing mobile P-FTP in the context of on-board IP networks. In that case, the functionality of the P-FTP server is implemented in the MR. As mentioned earlier, we assume that the MR can predict with reasonable accuracy the change in the available access link bandwidth as the PTV travels along its route. This information is passed on to the P-FTP clients by the MR through periodic broadcast messages on the on-board LAN. During the periods of large bandwidth availability, the P-FTP clients start additional file transfer sessions to accelerate the download speed. In the same manner, the number of file transfer sessions are reduced when the available bandwidth at the access link is less. In addition, the P-FTP clients can detect congestion at the access link and accordingly reduce the number of parallel file transfer sessions to ease the congestion [6]. It is interesting to note that, this behavior can be achieved without any feedback from the MR. This can considerably reduce the response time for congestion events and also allow recovery from prediction errors. Due to space limitations, further details have been omitted but interested readers can consult [6].

In Mobile P-FTP, the P-FTP clients start multiple additional file transfer sessions when more bandwidth is available. This may seem unfair to users of other applications (such as web browsing). To resolve this issue, the MR/P-FTP server should be designed in such a manner that other applications can get a fair share of the available bandwidth. Based on its admission control policy and prediction mechanisms, the MR can accurately estimate how much of the excess bandwidth needs to be apportioned for other applications. The MR will thus advertise only a part of the bandwidth increase (say 50% for example) to the P-FTP clients. Moreover, for better resource utilization, the P-FTP server can readjust update frequency depending upon the traffic volume at the MR.

IV. Analysis and Simulation Results

In this section, we study the performance of the mobile P-FTP approach and compare it with the traditional file transfer approach using NS-2 simulations [10]. We assume that the change in the access link bandwidth on all the interfaces as a function of the PTV's route is available to the MR in the form of a pre-generated file. The MR/P-FTP server sends this information to the P-FTP client along with the information about the file-servers and file portion sizes. We have run the simulations for a variety of scenarios. However, due to space limitations, in this section we only present two representative cases. Both the scenarios involve same topology of 100 routers generated with Boston University Representative Internet Topology Generator (BRITE) [11]. There are 10 mirror file servers, each of which was attached to one randomly chosen router.

The P-FTP client was compared with the traditional file transfer client in terms of the throughput and delay incurred while downloading a 100 MB file. The

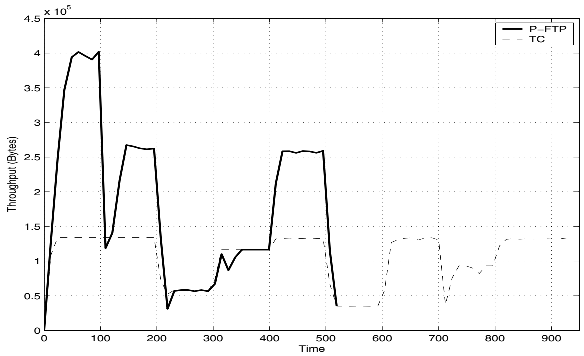

Fig. 3. Throughput of P-FTP and TC clients

TABLE I
AVAILABLE BANDWIDTH AND PARALLEL P-FTP SESSIONS

Time Interval (Sec)	Available Bandwidth(MByte)	Parallel Sessions (P-FTP)
0-100	8	3
100-200	3	2
200-300	0.5	1
300-400	1	1
400-500	5	2
500-600	0.3	1
600-700	10	–
700-800	0.8	–
800-900	2	–
900-1000	7	–

instantaneous throughput was measured and the average value was calculated for every 4 second interval. The traditional file transfer client downloads the file from a single closest server and is referred to as Traditional Client (TC) in the rest of this section. Both clients connect to the file servers using the Full-TCP model of NS-2. Background load was introduced into the network using on/off UDP flows at random paths of the network.

The first case was chosen when the MR was equipped with one wireless interface. The bandwidth capacity of the access link was chosen randomly between the range of 300 Kbps - 10 Mbps as shown in Table 1. The bandwidth was changed after every 100 seconds, which may seem too frequent as compared to the change in access link bandwidth for actual on-board LANs. The reason for choosing such a short interval was to show the effectiveness of the dynamic adaptation ability of the P-FTP client in response to the change in available bandwidth.

Figure 3 shows the average throughput of the P-FTP and TC clients during the transfer of a 100 MBytes file in first case. The throughput of both the clients is almost the same during the intervals of low available bandwidth (200-300 sec, 500-600 sec etc). However, during the high available bandwidth intervals (0-100 sec, 400-500 sec etc) the throughput of the P-FTP client is much higher than TC client since the P-FTP client starts multiple file transfer sessions during those intervals. The number of parallel file transfer sessions for P-FTP client is shown in Table 1. The download time of the 100 MByte file is 519 sec and that for TC client is 950 sec. This shows that the P-FTP client can effectively utilize available resources to reduce the download delays for large files.

For the second case, we assumed a multi home on-board network when a GPRS network of 144 Kbps was available throughout the route of the PTV and MR always connects to this network. In addition to that along the route of the PTV at every bus stop a wireless LAN of 10 Mbps capacity was provided. The bus stops were at an average distance of 2 Km from each other and the bus traveled at an average speed of 40 Km per hour so the bus reaches a bus stop after every 3 minutes. The range of WLAN was considered as 200 meters, so the bus remains within the WLAN range for approximately 20 seconds. In this case the MR/P-FTP server informed the user about the possible time intervals when the PTV may travel in the range of WLAN near the bus stops. The P-FTP client started additional file transfer sessions during the short period of those time intervals. During those times a considerable portion of the requested file was transfered due to the large wireless link capacity of WLAN available at that time.

Figures 4 and 5 show average throughput of TC and mobile P-FTP client during download of an 100 MBytes file while the user was a part of on-board IP network. The complete figure is divided in two parts to show more details. The throughput of TC was almost constant and the file was downloaded in 5919 seconds. The TC client used the traditional approach for file download which utilized constantly available low capacity GPRS connection to transfer file. Due to the small access bandwidth of the GPRS network the large file was transferred in huge amount of time with one file transfer session. The P-FTP client also used the same low capacity GPRS connection for file transfer by a single file transfer connection due to small access bandwidth. During the intervals when the bus traveled near the bus stops where high capacity WLAN was available the mobile P-FTP client initiated additional file transfer connections. The throughput of the mobile P-FTP client during those intervals was much higher and the spikes in the figures show that interval of time. The P-FTP client received the 100 MBytes file in 3306 second only.

By using the prior knowledge of available resources in near future the file transfer time was considerably reduced in both cases. The MR/P-FTP server informed the client about the time intervals when large wireless

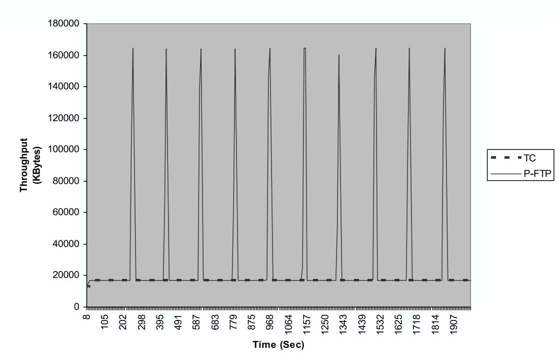

Fig. 4. Throughput of P-FTP client and TC (0-2000sec)

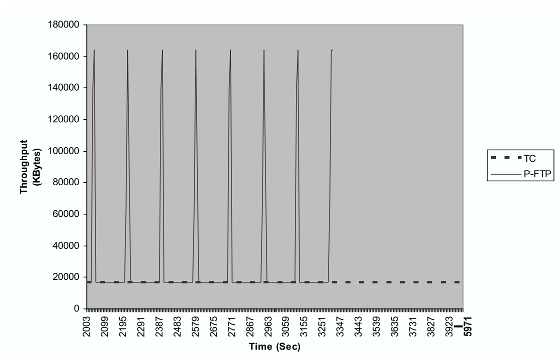

Fig. 5. Throughput of P-FTP client and TC (2000sec–)

resources would be available and the client used that information to accelerate the file transfer process.

V. Conclusion

In this paper, we have evaluated an extension of P-FTP, mobile P-FTP for on-board mobile networks. We have shown that, by exploiting the predefined and repetitive routes of public transport vehicles, Mobile P-FTP can perform file transfers effectively and in a reduced amount of time. Our simulations demonstrate that the performance of file transfers can be improved many folds for users of such on-board LANs. Even though we have focused on file-transfer, a similar approach can be adopted for improving performance of other applications such as streaming multimedia.

References

[1] V. Devarapalli, *Network Mobility (NEMO) basic support protocol*, Internet draft, June 2004.
[2] *Inmarsat's Swift64*,http://www.inmarsat.com/swift64.
[3] *European Commission IST OverDRiVE*, http://www. ist-overdrive.org, Aug 2004.
[4] M. A. Malik, S. S. Kanhere, M. Hassan and B. Benatallah *On-board RSVP: an extension of RSVP to support real time services in on-board IP networks*, In Proceedings of IWDC 2004.
[5] S. Sohail, S. Jha and H. Elgindy, *Parallelized File Transfer Protocol*, High Speed Local Networks , In Proceedings of IEEE Local Conference on Networks 2003.
[6] S. Sohail and S. Jha, *QoS Driven Parallelized File-Transfer-Protocol*, In proceedings of ATNAC 2004.
[7] Planet Lab, http://www.planet-lab.org/ Jun-2004
[8] Paraload Project, http://research.microsoft.com/ pablo/paraload.aspx.
[9] P. Rodriguez et al., *MAR: A Commuter Router Infrastructure for the Mobile Internet*, In Proceedings of the ACM Mobile Systems, Applications and Services Conference, 2004.
[10] *ns-2*, http://www.isi.edu/nsnam/ns, Aug 2004.
[11] *BRITE*, http://www.cs.bu.edu/brite, Aug 2004.