ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 7, January 2016

Performance Enhancement: P2P content sharing on Smart Phone Network by using Chunking algorithm

Vairagar Sagar , Vairagar Shubhangi, Shyam Sunder Gupta

Abstract*— *Mobile environment is very effective way of sharing contents between users. These file sharing using Peer to peer scheme is limited due to the lesser distance factor and continuous availability of transport medium. Whenever large file transfer is required, it requires more time and limitation on radius. Additionally content sharing is in plain format without any security considerations. Peer-to-peer networks offer several advantages over traditional client-server networking models like lack of connectivity to reliable hosts or servers and the use of inexpensive communication channel. Peer-to-peer network model has become popular in the wired broadband environment but not yet been effectively adapted to the mobile network environment. Many researchers are currently proposing and developing new P2P schemes for mobile environments as well. They are significantly adapted into applications such as the sharing of large files like multimedia and DB files between mobile devices. The peer-to-peer model based on mobile environment having several challenges like the limited on device processing power, limited memory, wireless data bandwidth, and available battery energy. With the proposed a peer-to-peer protocol, these specific constraints are addressed. We also investigate the feasibility of a practical implementation of a peer-to-peer file sharing model of android smartphones using Wi-Fi technology. This includes an analysis of impacted performance by various variables that can be dynamically controlled in the protocol. Proposed work is done on leading android platforms, we have found various optimal strategies which include minimizing the upload and download ratio to conserve battery life effectively, using flexible file segments to increase throughput, and decrease memory overhead.

***Index Terms*— Bluetooth, mobile computing, Parallel processing, Peer to peer computing, multimedia communication, wireless communication.**

I. INTRODUCTION

In the recent few years, communications based on Peer-to-Peer (P2P) protocol and their applications have become conventional architecture in the wired network environment. However, they have not been so much effectively adapted in mobile environment which is composed of various devices such as smartphone mobile devices, tablets, laptops, and devices with embedded software. Recently, Android-based, iOS-based, Windows-based Smartphones and tablets have been rapidly advancing.

Due to the popularity of advances of smartphones has been increasing since they are very powerful multimedia devices having built-in rapid processors, effective energy consumption, and good quality, high resolution touch screen with embedded cameras and sensors.

In Peer-to-Peer (P2P) systems, each node can act as a client and as a server at the same time and shares with others its own data. Smart phones have become highly-capable multimedia devices with built-in video cameras and many users enjoy sharing the multimedia content that they capture. Several users may be interested in automatically receiving multimedia from one particular source. There is cost problem associated when employing a 3G or 4G wireless data connection and uploading to a traditional. Users are typically subject to wireless data plans with finite usage limits and steep overage charges. The other alternative approach to upload content is through a wired USB connection is cumbersome and delays the user experience. A promising solution can be provided using peer-to-peer sharing among smartphones, in order to consume free peer-to-peer wireless links versus expensive packet data networks for file sharing purposes [1].

For instance, a camera-enabled device could publish new content and advertise it through a service that the other devices in the home would subscribe to. Through peer-to-peer network, the participating hardware devices could then share the multimedia resources back and forth without the use of any intermediary hosts, and not be subject to billing. This differs markedly from the traditional client-server model in which the parties have dedicated roles. The peer-to-peer model has found significant traction in the wireless mobile context, due to the many challenges of a resource-constrained environment. Here aim is to investigate the feasibility of implementing a peer-to-peer content sharing network on mobile phones to determine its feasibility. The performance of modern mobile devices such as smartphones has improved greatly, to the point that they are now able to run complex and bandwidth-intensive protocols that have traditionally been reserved for desktop systems only.

II. LITERATURE REVIEW

Following different approaches of peer to peer models:

A. Peer to Peer Content Sharing Application

Nokia Research Center and Helsinki University of Technology proposes a Mobile Peer to Peer Content Sharing Application. This method practices the SIP protocol as a foundation for the deployment of mobile P2P services. The service consists of a peer to peer client application in the mobile phone and an application server in the network. The mobile P2P client was fulfilled on the Nokia Series 60 (Symbian platform). Generic Engine for Collaborative

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 7, January 2016

Mobile Applications.

B. JMobipeer

This technique uses a framework designed to work on J2ME supported mobile devices on mobile ad hoc networks. This technique uses a reactive routing algorithm and modular architecture.

C. Proem

Proem is a technique for the expansion of P2P collaborative applications in mobile ad- hoc networking environments. Proem delivers a complete SDK which includes a group of Java interfaces and classes for speedy progress of mobile peer-to-peer applications named peerlets. Proem also delivers a runtime atmosphere for the implementation peerlets.

D. Peer2Me

Peer2Me is an open source project mounting a framework for mobile collaborative applications on cell phones. Peer2Me cares developers to make collaborative product for mobile phones using a network technology such as Bluetooth. The construction and concepts of structure are not dependent on the kind of PAN technology maintained in the mobile device.

E. Mobile Chedar

Mobile Chedar is an extension to the Chedar peer-to-peer network permitting mobile policies to access the Chedar network and also to transfer with other Mobile Chedar peers. Chedar (CHEap Distributed ARchitecture) is a peer-to-peer middleware designed for peer-to-peer applications.

F. Symella

Symbian smartphones uses Symella is a Gnutella file sharing client. It is capable of searching and transferring, but it not provides the facility for upload any data in its current release. It uses multi-threaded downloads it means that if multiple users have a particular file then Symella can download the file from several locations simultaneously. Gnutella is a flooded request model. Every request from a peer is flooded to straight linked peers. This solution chomps a lot of bandwidth, and therefore this mobile client does not funding data uploading [3].

G. SymTorrent

SymTorrent is a complete BitTorrent client for Symbian OS. It ropes downloading multiple torrent files at the same time it is also capable of both downloading and uploading of file contents. It can save the status of unfinished torrents, so that if application gets restarted downloading can be resumed from saved status. BitTorrent arranges peers sharing the same file into a P2P network and focuses on fast and efficient replication to distribute the file.

H. Bedd

Bedd is a commercial application which runs on Symbian Series 60 smart phones. Bedd, currently, uses GPRS and Bluetooth wireless technology. Bedd is an application which enables ad hoc mobile communication between mobile phones.

III. SYSTEM ARCHITECTURE

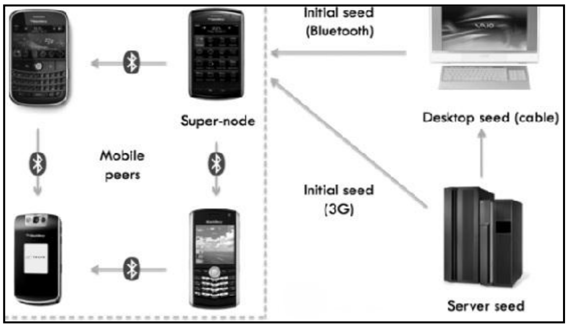

Fig 1

Peers communicate with each other via ad hoc connections. Each mobile terminal effectively acts as a server or host depending on the nature of the transaction, and thus performs dual functions. Each mobile node needs to obtain a full copy of a file identified by its file name, time stamp, and possibly other characteristics. The file itself is segmented and each segment is identified by a hash value. Individual segments may exist on different mobile nodes at any time. Each node may have a partial or complete copy of the file. The advantage of the peer-to-peer model is that a node may obtain the required segments from various nodes that it can actively connect to at any given time and then assemble them. This concept lends itself well to a mobile environment, where nodes are subject to high mobility and short-range connections are short-lived. This is in contrast to a Mobile Ad Hoc Network (MANET), because devices are not responsible for forwarding and routing traffic unrelated to their own. There is only point-to-point communication between the nodes; a peer in current system model only learns of another if a direct pairing occurs. A Personal Area Network (PAN) modelled using Bluetooth, typically called a piconet. The IEEE 802.15 working group specializes in Wireless PANs (WPANs). When two devices come into proximity of each other, they plug in and start communicating as if in a wired network, with lockout of unauthorized peers being possible. One difference in current model is that there is no dedicated master or slave role as each device has the capacity to upload and download. As each party downloads a file, it becomes a candidate for sharing it with others, and thus contributes to a so-called swarm that eases distribution of the file among participants. The greater the contributors that are within range of a peer, the quicker the file contents may be downloaded. For peer-to-peer connectivity, generally considers only low-cost short-range connections such as Bluetooth that are standard and ensure interoperability in a heterogeneous mobile device

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 7, January 2016

environment. The nodes that described are similar to transient nodes on Tier 3 in Gnutella that only join the network for a short period of time and connect to a very limited number of other nodes. Thus, the network is not democratic; leeching is effectively limited by the presence of few nearby peers. A node that has more connectivity options such as 3G may download content directly from a server to obtain an initial seed, or from a desktop computer with peer-to peer connectivity itself. This so-called super-node will then share its file content with multiple other peers using a peer-to-peer transport. Note that a single peer may draw segments from multiple sources and assemble them locally. A peer may create multiple simultaneous connections or transfer its content using one connection at a time. The user initially subscribes to a service of interest, such as an OS update or video sharing service. Peers discover each other through the process of scanning. Compatible services are discovered, and content that is assigned to a service is indexed and may be requested to be placed on a transfer queue. Individual segments are transferred and concatenated at the receiver end. Once all segments arrive, they are assembled, and the content is registered with applicable application handlers so that it becomes visible to the user.

Chunking Algorithm

Chunking algorithm read file attributes and split file into appropriate chunks. Created chunks are stored on device internal memory. Steps to split le:-

1. Get filename and verify file is existing on super node folder along with required permission on file.

2. Read selected file metadata such as file format, type, size, header details etc. Once all meta data read out, select best suitable chunk size.

3.Calculate number of chunks of split out file using formula Numberofchunks =TotalSizeofFile=SelectedChunkSize

4. Create File Input Stream and divide whole file into number of chunks.

5. Create individual file named as file chunk and copy under =V SV ideos= folder in internal memory.

Return.

IV. MATHEMATICAL MODEL

Steps for Proposed work

1. Discover peer nodes using discover service
2. Get appropriate file.
3. Distribute file in chunks
4. Save chunks on peer nodes.
5. Receive chunks from peers and merge it to file.

Systems Model

Let S be a system that describes peer to peer content sharing network system

S = {I,SS, SA,MA,CARR}

Where,

I = Input file/s.

SS = Split Size.

SA = File Split algorithm.

MA = File Merge algorithm.

Input File

I is set of files with appropriate types such as image, multimedia or text files in dataset. I = {i1,i2,i3,i4.......................in}n < 40

Split Size

SS is file split size is defined as chunks in which file is to split and save on different peers on network.

File Split Algorithm

SA = {I, SS,SC}

Where,

I is input file/s

SS is split size

SC is array of split fragments.

File Merge Algorithm

MA = {SS,SC}

Where,

SS is split size in which file get spilt.

SC is array of split fragments.

V. RESULTS

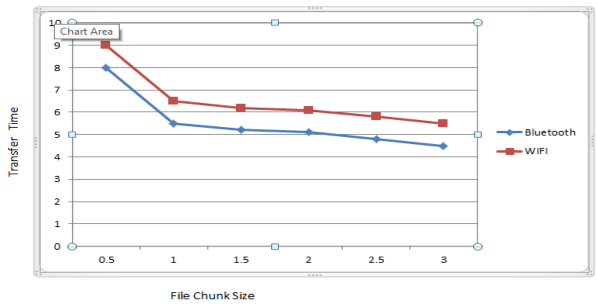

Fig 2

Figure 2, shows throughput vs time comparison of base system and existing system. This indicates transferring same file (much larger files) over WiFi accomplishes quickly without any burden. From the output graph it seems proposed system works well for different comparison factors such as throughput, distance range etc. Additionally chunk size, peer discovery works more dynamic in nature which provides further benefits over base system.

VI. CONCLUSION

In enhanced or improved system tried improve many aspects of content sharing business use case in peer-to-peer model. Thereby, the below conclusion is put forward in accordance with their original system as follows:

1. Peer-to-Peer model of content sharing is tried understood, done with analysis. After literature survey, finalized what are basic systems pitfalls and recommended ways to overcome.

2. In enhanced or improved version, tried to figure out every aspect of challenges faced by basic system right from authentication, registration to flexible chunk sizes. In enhanced system improved both hardware and software dependent design factors.

3. A sincere attempt has been made to borrow from the literature and conceived idea of uplifting the base system for range, availability, downtime attributes as it faces limitation on base system. Previous working is well understood and replaced with more appropriate and accurate system.

4. With achieved content sharing and in comparison with older system, chunk splitting algorithm works much better than existing system. Since base system was targeting Windows phones only and proposed system targeting Android OS, future enhancement can be targeted for across OS with considering leading OS platforms. Also, future enhancement can be done on cross platforms so that coding effort will one time.

REFERENCES

[1] Piotr K. Tysowski, Pengxiang Zhao, Kshirasagar Naik, “Peer to Peer Content Sharing on Ad Hoc Networks of Smartphones”, In proceeding of: Proceedings of the 7th International Wireless Communications and Mobile Computing Conference, IWCMC 2011, Istanbul.

[2] M. G. Williams, “Directions in Media Independent Handover”, IEICE Trans. Fundam. Electron. Commun. Comput. Sci., vol. E88-A, no. 7, pp. 1772–1776, 2005.

[3] R. Winter, T. Zahn, and J. Schiller, “Dynamo: A topology-aware p2p overlay network for dynamic, mobile ad-hoc environments”, Telecommunication Systems, vol. 27, no. 2, p. 321, 2004.

[4] A. Rowstron and P. Druschel, “Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems”, Lecture Notes in Computer Science, vol. 2218, 2001.

[5] A. Datta, “Mobigrid: P2P overlay and MANET rendezvous — a data management perspective”, in CAiSE 2003 Doctoral Symposium”, 2003.

[6] K. Aberer, P. Cudr´e-Mauroux, A. Datta, Z. Despotovic, M. Hauswirth, M. Punceva, R. Schmidt, and J.Wu, “Advanced peer-to-peer networking: The p-grid system and its applications”, PIK Journal - Praxis der Information sverar beitung und Communication, Special Issue on P2P Systems, 2003.

[7] Symbian OS, the mobile operating system. [Online]. Available: www.symbian.com

[8] R. Schollmeier, I. Gruber, and F. Niethammer, “Protocol for Peer-to-Peer Networking in Mobile Environments,” in Computer Communications and Networks, 2003. ICCCN 2003. Proceedings. The 12th International Conference on, 20-22 Oct. 2003, pp. 121–127.

[9] M. Matuszewski, N. Beijar, J. Lehtinen, and T. Hyyrylainen, “Mobile Peer-to-Peer content sharing application”, in Consumer Communications and Networking Conference, 2006. CCNC 2006. 2006 3rd IEEE, vol. 2, 8-10 Jan. 2006, pp. 1324–1325.

[10] T. Hakkarainen, V. Savikko, and A. Lattunen, “Generic engine for collaborative mobile applications,” in Proceedings of the IADIS International Conference WWW/Internet 2005 (ICWI2005), 2005, pp. 243 – 246.

[11] M. Bisignano, G. Di Modica, and O. Tomarchio, “JMobi Peer a middleware for mobile Peer-to-Peer computing in MANETs”, in Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on, 6-10 June 2005, pp. 785–791.

[12] G. Kortuem, J. Schneider, D. Preuitt, T. Thompson, S. Fickas, and Z. Segall, “When Peer-to-Peer comes face-to-face: collaborative Peer-to-Peer computing in mobile ad-hoc networks,” in Peer-to-Peer Computing, 2001. Proceedings. First International Conference on, 27-29 Aug. 2001, pp. 75–91.

[13] K. Kant, “An analytic model for peer to peer file sharing networks”, in IEEE International Conference on Communications, May 2003, pp. 1801 – 1805 vol. 3.

[14] A. Legout, G. Urvoy-Keller, and P. Michiardi, “Rarest First and Choke Algorithms Are Enough”, in IMC’06, Rio de Janeiro, Brazil, 2006.

[15] Yang, X. et al, “Service capacity of peer to peer networks”, in 23rd Joint Conference of IEEE Computer and Communications, 2004.

[16] Hu, T. H. et al, “Supporting mobile devices in Gnutella file sharing network with mobile agents”, in 8th IEEE Int. Symposium on Computers and Communication, Sep. 2003, pp. 1035 – 1040 vol. 2.

[17] Dhurandher, S. K. et al, “A swarm intelligence-based p2p file sharing protocol using bee algorithm”, in IEEE/ACS Int. Conf. on Computer Systems and Applications, 2009.

[18] Asadi, M. et al, “A scalable lookup service for p2p file sharing in manet,” in Proc. of the 2007 Int. Conference on Wireless Comm. and Mobile Computing. New York, NY, USA: ACM, 2007.

[19] Huang, C.-M. et al, “A file discovery control scheme for P2P file sharing applications in wireless mobile environments,” in Proc. of the 28th Australasian Conference on C.S., 2005.

.