A Crosslayer-aware Bandwidth Aggregation & Network Condition Determination System Using Multiple Physical Links

Soma Bandyopadhyay, Shameemraj M Nadaf
Innovation Lab
Tata Consultancy Services
Kolkata, India
soma.bandyopadhyay@tcs.com, sm.nadaf@tcs.com

***Abstract*—This article presents a cross layer aware bandwidth aggregation and network condition determination of a fixed computing system dynamically. The proposed system transmits and receives data simultaneously through multiple physical interfaces, also determines the condition of communication channel/network associated with those physical interfaces while performing the bandwidth aggregation. This proposed cross layer system enhances the download and upload data transmission rates of the applications. Its core functional block resides in between the data link/MAC (medium access control) and network layer and encapsulates the existing multiple physical interfaces. Aggregation of bandwidth and determination of network condition by using the proposed cross layer system is presented with case study and experimental results in detail along with future research scope here.**

Keywords—TCP; UDP; PPP; Ethernet; Bandwidth aggregation; Cross layer architecture; Network Channel

I. INTRODUCTION

A system that enables information sharing across the layers is regarded as a cross layer system. In this article, focus has been given to design and develop a cross layer system to achieve bandwidth aggregation and to determine the network condition of multiple communication links connected with the active physical interfaces of a fixed computing system.

The cross layer system as presented here, enhances the bandwidth of a system significantly by adding up the available bandwidths of the existing active communication interfaces (wired and wireless) without performing any modifications in the physical and data link layer of the existing interfaces. At the same time it determines the channel/network condition associated with each active physical interface. It is mainly an adaptive bandwidth aggregation system. This system runs multiple sessions over multiple interfaces. The proposed system encloses existing network interfaces, i.e., both MAC and physical layers of those interfaces. It does not need any counterpart/negotiation in any node including the final destination or end system of the communication link. It can be used for any transport layer protocol like TCP (transmission control protocol) and UDP (user datagram protocol). Notably, it does not require any service level agreement and a proxy support.

The case study, as presented here, performs a video streaming by using one interface, and a file download by using another interface simultaneously with an improvement in throughput achieved. It also presents the results of channel condition determination obtained by cross layer co-ordination for the multiple physical interfaces.

The remainder of this article is organized as follows. First the related work in cross layer based bandwidth aggregation and network condition determination is presented, followed by an overview of the proposed system. The architecture of the system along with the details of experimental study and analysis are then described. The final section concludes this article with the future scope of the model.

II. RELATED WORK

Data transfer through multiple physical interfaces is broadly studied for improving the total available bandwidth entitled to the applications. Most of the related work in this area requires proxy architecture, service level agreement and counter component at the destination, to realize the bandwidth aggregation either in adaptive or non-adaptive manners. A network layer architecture consisting of an infrastructure proxy [1] or a multilink proxy [2] is applied for simultaneous use of multiple interfaces and aggregation of the throughput of heterogeneous downlink streams. The approach presented in [3] uses dynamic packet reordering mechanism of TCP streams over multiple links, also requires a network proxy. Service level agreement as well as proxy is used in the middle of the network for scheduling the packets through

 ISBN: 978-1-61208-141-0

multiple interfaces in [4]. Session-layer striping architecture over multiple links is proposed in [5] based on single virtual layer socket. Furthermore [6], describes a network middleware called Horde, which allows application to control certain aspects of data stripping over multiple interfaces. This middleware architecture comprises three layers in which the higher layer provides an interface to interact with Horde, middle layer handles packet scheduling, bandwidth allocation, and the lower layer deals with network channels.

The framework based on cross layer concept, presented in [7], proposes adaptation across many layers of the protocol stack to support delay-critical applications in adhoc scenario, such as conversational voice or real-time video. The work presented in [8] makes changes to the five layers namely physical, data link, and application, network, and transport layer like TCP to provide seamless delivery of multimedia services in heterogeneous wireless networks. Some modifications have been made to the transport protocols which make deployment of such an approach difficult. A dynamic QoS negotiation scheme has been proposed in [9], to do bandwidth aggregation for video streaming in wireless networks.

Multiple interfaces of the same technology can also be striped for better performance at the link layer. The main idea of bandwidth aggregation on the link layer is to stripe data across a bundle of physical channels, as done in [10]. An adaptive inverse multiplexing technique has been described in [11] where IP Packets are fragmented by the multiplexor and tunneled through multiple links using multiple-link PPP over a link layer transfer protocol. A channel aggregation method in cellular networks is described in [12] where parity codes are applied across channels rather than across packets to improve the resilience. Another interesting approach is followed in [13], where it is proposed that users of WLANs should be able to multihome and split their traffic among all available access points, based on obtained throughput and a charged price. However, a link-layer solution of striping data through heterogeneous networks and to different IP addresses is not feasible because the link layer has no notion of IP. All these approaches described above demand modifications in both ends (i.e. server and client) to achieve bandwidth aggregation, and most of the approaches are tested in simulations, often based on very simple assumptions.

Network condition determination in real-time has also attracted quite a few research works. Most of the works in this area are based on RTT (round-trip-time) measurement, using a single or multiple probe packets. Theory, improvements and some implementations related to Network condition determination based on RTT mechanism have been discussed in [14]. UDP based probe packet mechanism is used to determine the network condition and do rate control in a P2P (peer- to- peer) based video streaming application in [15]. Depending on the determined RTT values the upload bit rates are decided and the available bandwidth is calculated. The bandwidth determined decides whether to accept or reject a new P2P client requesting a connection at a random data rate. This method results in unstable RTT measurements since set of peers may be located anywhere in the world. Pair of probe packets with a fixed delay and packet pair dispersion technique is used to measure the available bandwidth in [16]. Along with the RTT calculation the time delay between the probe packets is measured and analyzed to predict the network bandwidth here. In [17] a variable size probe packet is used to measure the available bandwidths. Along with RTT the packet size is also considered to estimate the network capacity. A testing network including a test box measurement infrastructure is used here.

However, these research efforts are based on single interface and not use multiple interfaces simultaneously and need counterpart in the destination. These approaches also do not make use of any cross layer technique.

In [18], a bandwidth aggregation mechanism without network condition determination based on the application layer has been described. Here, the application data is distributed over multiple sessions communicating over multiple interfaces and assembled to get the complete application traffic. However, this system does not use any cross layer approach. Also importantly it is not adaptive in nature.

III. SYSTEM OVERVIEW

The cross layer aware bandwidth aggregation and network condition determination system as proposed here creates a ‘virtual physical’ interface by using its associated network driver module, which encapsulates all the existing active physical interfaces present in the computing system. It does not perform any modifications in the physical and data link layer of the existing physical interfaces. The virtual physical interface resides as the only default entry in the routing table, and provides a single communication pipe from IP (internet protocol) to the physical interfaces, starting from applications and vice-versa. The packets pertaining to different sessions are distributed to the multiple physical interfaces by this module for transmission, and also upon reception the packets are processed by this module, and pushed to the higher layers. Since there is no counterpart at the receiving side, the packets of a session are distributed through one physical interface only but packets of next session get assigned to the other physical interface. Depending upon the QoS requirement (if any) of an application the packets of that session can be assigned to a physical interface based on its network condition. Thus this proposed module enhances the bandwidth of a system significantly by adding up the available bandwidths of the existing active communication interfaces. It estimates the channel condition of the active physical interfaces by using probe-packet mechanism, and analyzing the transport header statistics.

 ISBN: 978-1-61208-141-0

The proposed system can be configured to operate in following modes:

•Bandwidth Aggregation mode: aggregates the bandwidth of multiple network interfaces.

•Bandwidth Aggregation with Network Determination mode: aggregates the bandwidth of multiple network interfaces and also determines the channel condition of each network interface dynamically at predefined time intervals (configurable).

It has two components - user space and kernel space as depicted in Fig. 1 respectively. User space component exposes APIs (application programming interface) to take the user inputs as well as system captured inputs.

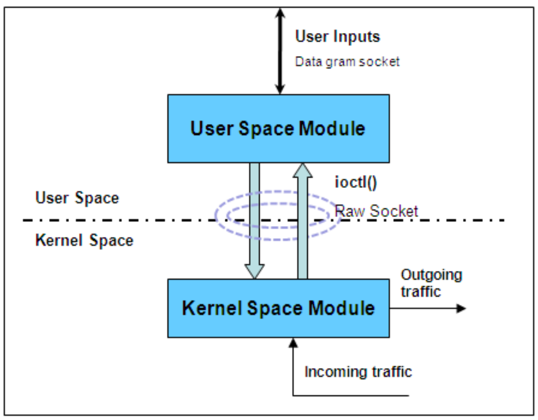

Figure 1. Building blocks of the system

User inputs:

1. Domain name of the server to which probe packets are sent for network condition determination
2. Time interval for performing network condition determination
3. QoS requirement of an application (optional)

System captured inputs:

- Information about the network interfaces (i.e. interface identifier, IP addresses, IP address of Gateways if any)

User inputs are provided by the user through command line. System captured inputs are obtained by doing an ioctl () function call with a datagram socket.

User Space and kernel space have a close bonding and both these components exchange control information by using ioctl () function call from user space component to kernel space component with a raw socket. The various operations performed by the user space components are as follows:

- GETINFO: to collect information available with kernel space module about the network interfaces for aggregation.
- SETINFO: to pass the information about the available network interfaces to the kernel space module.
- NDMINFO (Network Diagnostics): to pass information related to network condition determination, QoS requirement of an application to the kernel space module and also to collect the information about the channel condition of the network interfaces from the kernel space module.

IV. ARCHITECTURE OF PROPOSED SYSTEM

The architecture of the proposed cross layer aware bandwidth aggregation and network condition determination system is portrayed in the current section. The overview of the system architecture along with its functional components is depicted in Fig. 2. The proposed system acts as a gateway of all the data paths from upper layers to the physical interfaces and vice versa. It distributes the data flow among the active physical interfaces for transmission based on the network condition determination feedback. QoS requirement, incase available, is also going to be used for data flow distribution. It receives the data from the interfaces and passes it to the respective applications.

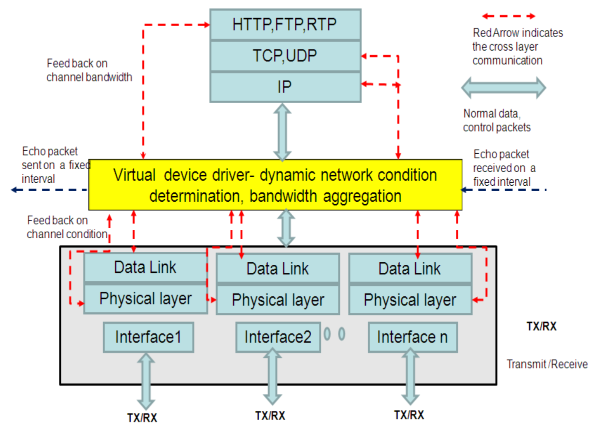

Figure 2. Functional blocks of the system

The proposed system creates a virtual interface and assigns an IP address, net mask to it and adds this interface as the default entry in the routing table. All the applications data coming from upper layer uses this IP address as the source address. The proposed system replaces its own IP address with the corresponding active physical interfaces IP addresses, while distributing the packets to those interfaces and performs the necessary checksum calculations for IP, and transport protocols headers as well. After receiving the packets it replaces the actual IP addresses of the interfaces with its own IP address and performs the necessary checksum calculations for IP and transport protocols headers. It distributes the data packets from application based on some predefined identifiers for example port number (HTTP packet or FTP packet etc.) and QoS requirement as specified by the application through the user space module. It sends the distributed data packets directly to the transmit queue of its slave interfaces i.e., the active physical interfaces. It uses packet filtering (net filter) mechanism and associates a hook function for processing the received packets. The hook function is used to filter the packets just after their reception by the active interfaces. The associated hook

 ISBN: 978-1-61208-141-0

function of the packet-filter performs the necessary modifications in the data packets, and assembles the data packet before sending to the application.

The proposed system uses a predefined ICMP echo packet for measuring the network channel condition. The ICMP echo packet is sent to the public IP address (for example-www.google.com) defined by user (defined via user input 1) simultaneously through the existing multiple active interfaces. The destination with the public IP address sends back the echo-reply to active physical interfaces. It determines the time difference between the sent ICMP echo-request and received ICMP echo-reply packets i.e. the round trip time (RTT) for the active interface and estimates the network condition. The time difference with a higher value signifies a poor network condition. The echo packet is sent at a fixed time interval (configurable, via user input 2), for this the proposed system maintains a timer. Based on the network condition and QoS requirement (via user input 3), along with the above described address mapping from virtual to physical interface and vice-versa, the data packets distribution among the physical interfaces is performed by this system. There is also a provision to judge the transport header (TCP) statistics and to take the average RTTs obtained from both the transport header and ICMP probe packet for assessment of network condition; this is a future scope of the current work.

V. EXPERIMENTAL ANALYSIS

Here the proposed system is implemented using a Linux based PC, two CDMA 1XRTT based interfaces are used for internet access and aggregation of bandwidth, and the Wireshark network analyzer tool running on the PC is used for analyzing the network traffic. In this Experiment, the identifier used for data flow distribution to multiple physical interfaces is the port number (HTTP packet or FTP packet). Here, the numbers of applications running are equivalent to the numbers of physical interfaces. The experiment consists of following two phases.

1. Phase 1- tasks are carried out with a single CDMA 1XRTT interface in absence of the proposed system.
2. Phase 2 - tasks are carried out with two CDMA 1XRTT interfaces in presence of the proposed system.

Following two tasks are carried out simultaneously in each phase of the experiment:

1. Video streaming from YouTube link.
2. File download from ftp link.

In case of the first phase both video streaming and file download take place via a single interface whereas in case of second phase the video streaming takes place through one interface and the file download takes place via the other interface. Throughputs achieved in both the phases are mentioned in Table 1. The average throughput obtained by single interface in phase 1 is 5.7602 kbps (kilobytes per second), this throughput is shared amongst the two tasks namely video streaming and file download. In case of phase 2, average throughput obtained by interface 1 is 6.1565 kbps (kilobytes per second) and average throughput obtained by interface 2 is 5.8978 kbps (kilobytes per second). As mentioned earlier the two tasks video streaming and the file download take place via interface 1 and interface 2 respectively, hence the throughput received by the tasks is almost twice the throughput received in case of phase1.Thus a significant improvement is seen in the throughput achieved by use of proposed aggregation module. Throughput graphs shown in Fig. 3 & Fig. 4 depict the variation of throughput with elapsed time in case of Phase 1 and Phase 2 of the experiment respectively, whereas the Fig. 5 shows the effective aggregated throughput achieved by the proposed system.

Table 1. Improvement in Throughput

Phase of Experiment	**IP address of Interfaces**	**Average throughput achieved in kilobytes per second**
Phase 1 : one CDMA 1xRTT interface	Single interface : 14.96.13.4	Single interface: 5.7602
Phase 2 : two CDMA 1xRTT interfaces	Interface 1 : 14.96.25.157 Interface 2 : 14.96.17.240	Interface 1 : 6.1565 Interface 2 : 5.8978 Combined : 12.0543

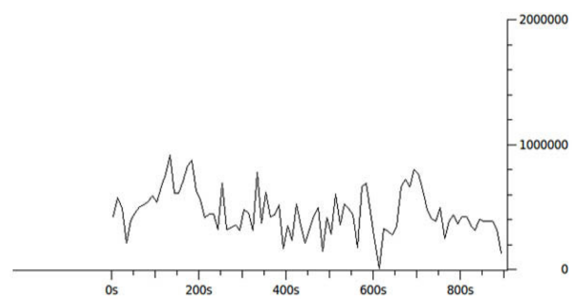

Figure 3. Throughput in Phase 1 (x-axis = time elapsed in seconds, y-axis = bits per second)

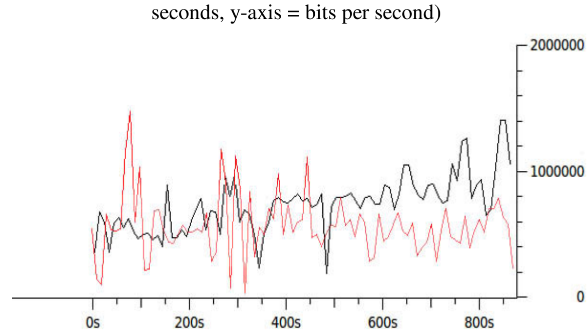

Figure 4. Throughput in Phase 2 (x-axis = time elapsed in seconds, y-axis = bits per second, black – interface 1 throughput, red – interface 2 throughput)

 ISBN: 978-1-61208-141-0

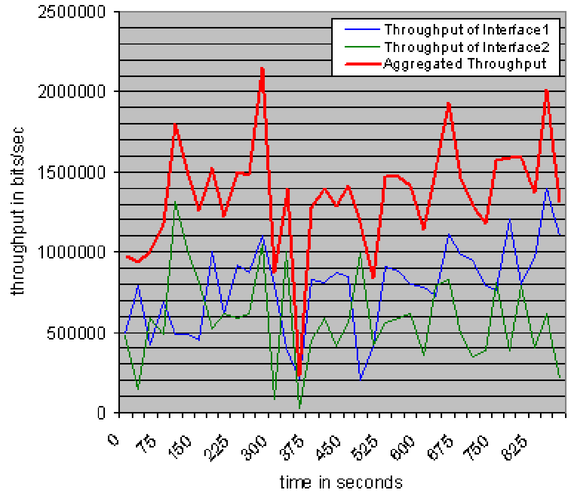

Figure 5. Aggregated throughputs in Phase 2

Determination of network condition of active physical interfaces is done by the aggregation module using the ICMP Echo packets here. Variation of RTT values with elapsed time at intervals of 20 seconds for the interfaces in phase 2 (with the proposed system) is shown in Fig. 6. It is observed from Fig. 6 that as the throughput increases values of RTTs obtained is low and the RTTs values obtained is high with reduction in throughput. However the variation in RTT values is not consistent due to the dynamic network condition of wireless interface and queuing of received packets by net filter hook mechanism. The ICMP Probing is done to a public IP address (e.g. www.google.com) here and also there is no proxy involved in between the source and destination. Hence the delay encountered at the hops in between the source and destination is not considered for RTT measurement. It should be noted that since the proposed system is based on a cross layer solution different use cases can be implemented by making negotiations at the multiple layers of the protocol stack as required in general.

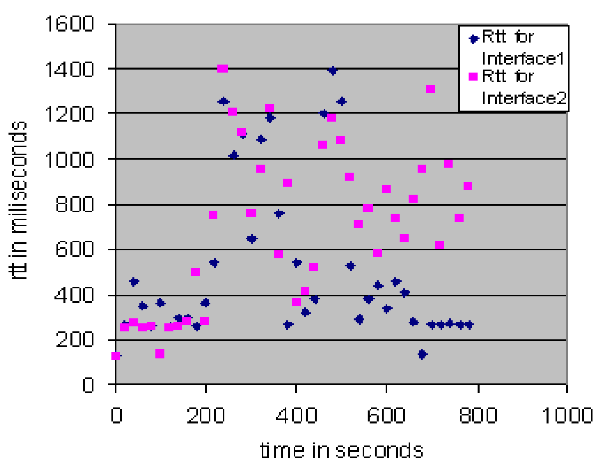

Figure 6. RTT measurement in Phase 2

VI. CONCLUSION AND FUTURE SCOPE

In this paper a cross layer aware bandwidth aggregation and network condition determination system with multiple physical interfaces is proposed, which performs the bandwidth aggregation and also determines the channel condition of the existing interfaces. Bandwidth aggregation is performed based on the determined channel condition of the existing interfaces. The experimental results with multiple interfaces, as presented, have shown a significant improvement in throughput achieved. The network condition determination is performed by sending ICMP echo packet to a fixed public destination periodically. The proposed system coordinates with application layer to get the QoS requirement of the applications through its exposed APIs. This does not need any counterpart in any node including the final destination or end system of the communication link, therefore, easy to manage, deploy and is cost effective. It can be used for any transport layer protocol like TCP (transmission control protocol) and UDP (user datagram protocol). Importantly, it does not require any proxy support and service level agreement.

There is a scope for future research work to decouple the network condition determination module from the bandwidth aggregation module. Optimal utilization of network bandwidth i.e., to use unused bandwidth of any interface, at a particular time interval based on the feedback of network condition determination function, and managing multiple number of applications, when the number of applications are greater than the number of active physical interfaces. Network condition determination by taking the average of RTTs obtained from both ICMP echo packet and transport (TCP) header statistics, and also transferring the session from one interface to other with out breaking the connection when the network condition degrades is a part of our future research activity. This proposed system can be further customized to achieve enhancement of QoS of different multimedia applications like video conferencing, remote healthcare system etc.

REFERENCES

[1] K. Chebrolu and R. R. Rao, “Bandwidth aggregation for real-time applications in heterogeneous wireless networks”, IEEE Transactions on Mobile Computing, vol. 5, no. 4, pp. 388–403, 2006.

[2] K. Evensen, D. Kaspar, P. Engelstad, A.F. Hansen, C. Griwodz, and P. Halvorsen, “A network-layer proxy for bandwidth aggregation and reduction of IP packet reordering”, http://simula.no/research/nd/publications/Simula.ND.369/simula_pdf_file, 28.04.2011.

[3] K. Chebrolu, B. Raman, and R. R. Rao, “A network layer approach to enable TCP over multiple interfaces”, IEEE Transactions on Wireless Networks, vol. 11, no. 5, pp. 637–650, 2005.

[4] J.C. Fernandez, T. Taleb, M. Guizani, and N. Kato, “Bandwidth Aggregation-Aware Dynamic QoS Negotiation for Real-Time

Video Streaming in Next-Generation Wireless Networks", IEEE Transactions on Multimedia, vol. 11, no. 6, pp. 1082-1093, 2009.

[5] A. Habib, N. Christin, and J. Chuang, "Taking advantage of multihoming with session layer striping", http://www.andrew.cmu.edu/user/nicolasc/publications/gistripping.pdf, 28.04.2011.

[6] Horde: Flexible Application Driven Network Striping, http://publications.csail.mit.edu/abstracts/abstracts05/horde/horde.html, 28.04.2011.

[7] Cross Layer Design of Ad-hoc Wireless Networks for Real-Time Media,

http://www.stanford.edu/~zhuxq/adhoc_project/adhoc_project.html, 28.04.2011.

[8] T. Taleb, K. Kashibuchi, A. Leonardi, S. Palazzo, K. Hashimoto, N. Kato, and Y. Nemoto, "A cross-layer approach for an efficient delivery of TCP/RTP-based multimedia applications in heterogeneous wireless networks", IEEE Transactions on Vehicular Technology, vol. 57, no. 6, pp. 3801-3814, 2008.

[9] T. Taleb, T. Nakamura, and K. Hashimoto, "Bandwidth aggregation-aware dynamic Qos negotiation for real-time video streaming in next-generation wireless networks", IEEE Transactions on Multimedia, vol. 11, no. 6, pp. 1082-1093, 2009.

[10] H. Adiseshu, G. Parulkar, and G. Varghese, "A reliable and scalable striping protocol", ACM SIGCOMM, vol. 26, no. 4, pp. 131-141, 1996.

[11] A.C. Snoeren, "Adaptive inverse multiplexing for wide-area wireless networks", GLOBECOM, vol. 3, pp. 1665-1672, 1999.

[12] J. Chesterfield, R. Chakravorty, I. Pratt, S. Banerjee, and P. Rodriguez, "Exploiting diversity to enhance multimedia streaming over cellular links", INFOCOM, vol. 3, pp. 2020-2031, 2005.

[13] S. Shakkottai, E. Altman, and A. Kumar, "The case for non-cooperative multihoming of users to access points in IEEE 802.11 WLANs", INFOCOM, pp. 1-12, 2006.

[14] J. Curtis and T. McGregor, "Review of Bandwidth Estimation Techniques",

http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.779, 28.04.2011.

[15] M. Saubhasik and U. Schmidt, "Bandwidth Estimation and Rate Control in Bit Vampire",

http://www.uweschmidt.org/files/network_project.pdf, 28.04.2011.

[16] J. Navratil and R. Les. Cottrell, "ABwE: A Practical Approach to Available Bandwidth Estimation", http://www.nlanr.net/PAM2003/PAM2003papers/3781.pdf, 28.04.2011.

[17] T.G. Sultanov and A.M. Sukhov, "Simulation technique for available bandwidth estimation",

http://arxiv.org/PS_cache/arxiv/pdf/1007/1007.3341v1.pdf, 28.04.2011.

[18] Soma Bandyopadhyay, Arpan Pal, and Shameemraj Nadaf, "A Novel Bandwidth Aggregation System Using Multiple Physical Links", ICCAIE, pp. 34-37, 2010.