A Novel Bandwidth Aggregation System Using Multiple Physical Links

Soma Bandyopadhyay
Innovation Lab
Tata Consultancy Services
Kolkata,India
soma.bandyopadhyay@tcs.com

Arpan Pal
Innovation Lab
Tata Consultancy Services
Kolkata,India
arpan.pal@tcs.com

Shameemraj Nadaf
Innovation Lab
Tata Consultancy Services
Kolkata,India
sm.nadaf@tcs.com

***Abstract*— This article presents an approach towards aggregating bandwidths of multiple physical layers i.e. transmitting and receiving data simultaneously through the multiple physical interfaces of a fixed computing system. The data over physical layers can communicate over both wired and wireless communication medium at the same time. This proposed bandwidth-aggregation-system (BWA) enhances the download and upload data transmission rates of the application and hence the QoS (quality of service). It resides in the application layer and does not deal with any interaction/modification in the physical and data link layer modules of the interfaces. Numerical results confirm that the enhancement of bandwidth is directly proportional to the summation of data transmission capacity of the physical interfaces and also proportional to the number of physical interfaces having same data transmission capacity. The same concept can be extended to achieve adaptive QoS based on a cross layer communication model.**

***Keywords*—QoS; TCP; UDP; PPP; Ethernet; Bandwidth**

I. INTRODUCTION

Bandwidth is a resource of communication and is measured as the amount of data that can be carried from one point to another in a given time period (usually a second), expressed as bits/sec.

It is used as a parameter to determine the condition/capacity of channel/communication link.

A communication link has its own bandwidth. Reduction in bandwidth decreases the data rate of the link; this can cause the loss of information. Information loss degrades the quality of service (QoS). Whereas increase in bandwidth enhances the data rates and improves the quality of service, reduces download and upload data transmission time. In most of the cases the system has more than one communication links connected to different physical interfaces. Simultaneous use of these physical interfaces for data transmission and reception reduces the total transmission and reception time and hence increases the overall bandwidth of the system. Therefore, there is a need for a procedure which can use the bandwidth of multiple physical interfaces concurrently. Particularly, aggregating bandwidths of multiple active physical layers associated with different physical interfaces, and performing of communication with different communication links of wired and wireless network through these interfaces, increase overall bandwidth of the system. To achieve bandwidth aggregation multiple active physical interfaces are essential.

Transferring data through multiple physical interfaces is broadly studied in cases like:

1) Socket implementations using parallel communication streams known as multiple network interface socket (MuniSocket), the common practice is to use it only for UDP (user data gram protocol) with the provision for counterpart in the destination [1],

2) Using one of the physical interfaces at a time specially in the case when one is not working [2],

3) Using service level agreement and a proxy sitting in the middle of the network which schedules the packets for transmitting through multiple interfaces [3],

4) Using a network layer architecture consisting of an infrastructure proxy [4], a multilink proxy [5] for simultaneous use of multiple interfaces and aggregation of the throughput of heterogeneous downlink streams. The approach suggested in [6] uses dynamic packet reordering mechanism of TCP streams over multiple links, also demands a network proxy.

Other research efforts are based on striping mechanism, for instance, architecture for session-layer striping is proposed in [7] based on single virtual layer socket, while architecture of network striping based on networking middleware for providing capabilities to applications with high demands on uplink throughput is presented in [8]. All these approaches demand modifications in both ends (i.e. server and client) to allow the full potential of bandwidth aggregation. The simultaneous employment of multiple active physical interfaces has not been applied to that extent particularly without modifying any features/configuration parameters of data link and physical layers for any transport protocol and without having any counter component in the destination system and without using any proxy in between the source and the destination.

Bandwidth-aggregation-system (BWA) as presented here can enhance the bandwidth of a system significantly by adding up the fixed available bandwidths of the existing active communication interfaces (wired and wireless) without performing any modifications in the physical and data link layer. This does not need any counterpart 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. Importantly, it does not require any service level agreement and a proxy support.

The remainder of this article is organized as follows. First an overview of the proposed system is introduced followed by the presentation of the architecture of the system along with technical achievements. The details of experimental study and analysis are then described. The final section concludes this article with the future scope of the current proposed model.

II. SYSTEM OVERVIEW

The bandwidth aggregation system as proposed here, determines the active physical interfaces capable of data transmission and reception viz ppp0, eth0 etc. It determines the IP (Internet Protocol) addresses of the interfaces, and the IP address of the gateway of those interfaces. The proposed system divides the total data to be handled among the concerned interfaces, keeping the ratio of data division identical to the ratio of default data rates of the interfaces.

The BWA system increases the overall data transmission and reception rate, and thereby reduces the overall operation time of an application for fetching or transmitting data keeping the functionality of various layers of OSI (open systems interconnection) or TCP/IP stack like physical, data link, medium access, network, transport, etc. unchanged.

The bandwidth aggregation system requires multiple active physical interfaces and more than one network connectivity. It adds these interfaces to the routing table as new routes. Following pre-requisites are required as user defined inputs in order to realize the system:

User Inputs

- The total number of interfaces to be aggregated;
- The interface identifier like eth0/eth1(Ethernet), ppp0/ppp1 (PPP) etc.;
- The data rates of the interfaces in kbps (killo bits per second);
- The 'url' (uniform resource locator) of the data operation; example- url of video for streaming or url for a file to download or upload.

The proposed system uses web scripts (like PHP scripts [9]) to operate at the remote end system as per the user requirement. Example: Merging the components of an uploaded file at the remote system.

III. ARCHITECTURE OF PROPOSED SYSTEM

The proposed architecture of the system for enhancing the bandwidth usage of a computing system comprising multiple physical interfaces, and multiple network connectivity is portrayed in the current section. The overview of the proposed architecture is depicted in Fig. 1.

Bandwidths of the respective interfaces are getting aggregated, and the operation time of fetching and transmitting data is getting reduced keeping the radio/physical layer, MAC (medium access control) layer functionalities unchanged.

The BWA system reduces the data transmission and reception time of a computing system by distributing the data to be processed (transmission/reception) amongst various physical interfaces connected to different/same access network present in that computing system.

A computing environment having TCP/IP communication stack and multiple physical interfaces connected to the multiple access networks is used. The active interfaces are identified and added as default route in the routing table as next-hops. All the interfaces hold equal priority for handling data traffic.

The bandwidth aggregation (BWA) system is used to maintain multiple operating threads in parallel. Numbers of operating threads are identical with the numbers of interfaces the bandwidth of which is to be aggregated.

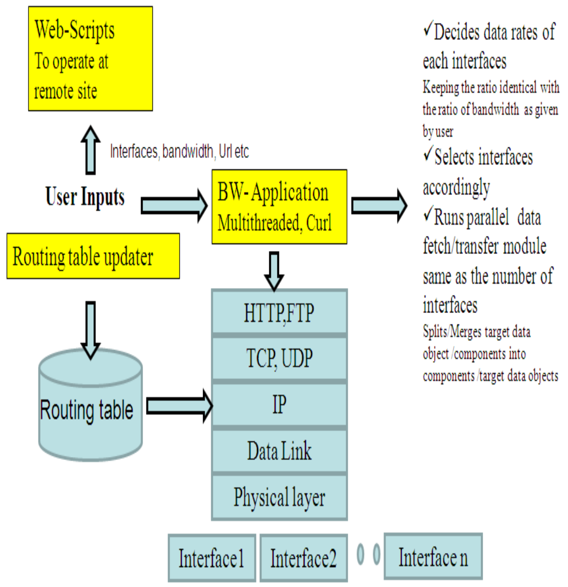

Figure 1. Functional block of bandwidth aggregation system

Operating threads can perform data fetch as well as data transfer activity.

Referring to the accompanying drawing, Fig. 1 shows the architecture of bandwidth aggregation (BWA) system that uses multiple physical interfaces connected to different/same access network.

The system comprises the following:

- User Inputs/Interface;
- Routing table updater;

- Routing table;
- Core control module or BW-Application module-Multithreaded bandwidth aggregation control module in application layer;
- Application (HTTP (hyper text transfer protocol), FTP (file transfer protocol)), Transport (TCP/UDP), Network (IP), MAC, Physical are the various protocol layers in a TCP/IP stack; and
- Physical interfaces.

The user interface receives the user defined inputs like interface identifier, default bandwidth of the interfaces, url /destination addresses, mode of operation (upload, download, streaming etc.), filename to be processed in case of a file download or upload operation by the BWA system. The routing table updater determines the active interfaces present in the concerned computing system.

In addition, the routing table updater determines the IP address of the interfaces and the IP address of the gateway of these interfaces. In case of default interface already existing as default route, it searches with keyword 'default' in the route table to determine the IP address of the gateway of that interface. For other interfaces it searches for that interface identifier in route table and reckons the first IP address as gateway IP address.

The routing table updater creates the routing table for the interfaces that are chosen for bandwidth aggregation. It adds these interfaces as default route along with their gateway addresses as next-hops to the existing routing table of the system. It assigns the equal priority for data transmission and reception to these interfaces.

The core control module (BW- application) resides in application layer of TCP/IP protocol stack. It can interact with the multiple application layer protocols like FTP (file transfer protocol), HTTP (hyper text transfer protocol). It takes user inputs. Depending on the default data rates of the interfaces it determines the amount of data to be distributed for either transmission or reception by the interfaces. If the data rates/bandwidth of the n interfaces are n1, n2, … nn the ratio of the data to be handled by the n interfaces is n1:n2:..nn. i.e. it keeps the ratio of the amount of data to be handled by the interfaces identical with the ratio of the default data rates of the interfaces.

The core control module creates operational threads for each interface and remains as the main master thread. Operational threads independently perform data transmission or reception of the data allocated to them by the control module, through different physical interfaces. It merges the data received by the interfaces in the local system. It also merges the data transmitted to the same remote system. The rates of data transmission and reception/bandwidth of the system happen to be the summation of the data rates of the physical interfaces.

Following section summarizes the steps which are invoked sequentially to aggregate the bandwidth of active physical interfaces enhancing the overall bandwidth of a system:

1. Determining the active interfaces in the system by executing 'ifconfig' command.
2. Determining the IP address of the interfaces and the IP address of their gateway by searching the routing table. Search 'keys' to be used are 'default' and the identification of interfaces (like eth0, ppp0 etc.).
3. Creating routing table for each active interface.
4. Updating routing table which is maintained by operating system by adding the active interfaces along with their gateways. The gateways and interfaces are added as default routes and as next-hops with equal priority for traffic handling.
5. Determining the ratio of data distribution for transmission and reception among the interfaces based on the given data rates of the interfaces. It keeps ratio of data distribution same as the ratio of the given data rates of the interfaces.
6. Creating multiple operational threads, where the numbers of threads are same as the numbers of concerned interfaces.
7. Distributing the data to the concerned interfaces for transmission or reception.

The physical interfaces through which data transmission and reception are occurring are bound by the respective operational thread of the control module (BW-application). The physical interfaces are bound by their specific IP addresses/interface identifiers. Therefore the data distributions and coupling with the physical interfaces are happening in the application layer only.

The technical achievements are depicted below:

- Providing a system which enhances the data rates of a system for its communication to the external network (applicable for both wired and wireless) by n times of the data rates using a single physical interface, where 'n' is the number of physical interfaces consisting of identical data rates/bandwidth and communicating with similar network;
- Providing a system which can aggregate the data rates/bandwidth of multiple physical interfaces of a computing system, i.e. if there are n interfaces comprising data rates b1, b2…. bn then the data rates of that computing system becomes b1+b2+… bn i.e. ∑bi where i = 0 to n;
- Providing a system where the data can be transferred or received through multiple physical interfaces simultaneously;
- Providing a bandwidth aggregation system which can reduce the total operational time of data handling by transferring and receiving the data through multiple physical interfaces simultaneously;
- Providing a system which assembles the data transfer and reception path of physical interfaces keeping the physical layer, MAC and other upper layer functional blocks / functionality unchanged, excepting making addition to routing table of the network layer; and

- Providing a system which interfaces with application layer of the TCP/IP stack only, as well as aggregates the data transmission and reception rates and thus increases the total bandwidth of the computing system under operation.

IV. EXPERIMENTAL ANALYSIS

The complete system was implemented using a Linux based PC where the bandwidth aggregation was performed by using two USB (Universal Serial Bus) CDMA 1xRTT wireless modems (and extended up to 4 interfaces). As part of the experiment, two files of sizes 30.94 MB and 141.04 MB were downloaded - first using a single modem and then using the proposed aggregation system using two modems. Comparison between the data rates/bandwidth of single interface and the aggregated bandwidths of the multiple interfaces are shown in the TABLE 1.

In the first case the average observed speed is noticed as 60 kbps (kilobits per second), and aggregated speed is noticed as 101.340kbps. In the second case the average observed speed is noticed as 57.031 kbps (kilobits per second), and aggregated speed is noticed as 110.43 kbps. Significant performance improvement i.e. enhancement of bandwidth is achieved by using the proposed system. Aggregated speed becomes almost twice (same as number of interfaces) of the speed of a single interface. It should be noted here that since the aggregation logic is implemented in Application layer, it is relatively simple to implement use cases other than file download.

TABLE I.
IMPROVEMENT OF PERFORMANCE USING BWA

The Performance Enhancement			
Download file size	Interfaces Used	Observed speed in kbps per interface	Enhanced data rates in kbps with aggregation
30.94 MB	Interface 1 & 2(Epivalley modem using CDMA)	Interface1 = 61.436 Interface2 =59.9	Aggregated rates = **101.340** 41.44 / 70% increased data rate (best case out of 4 attempts)
141.04MB	Same as above	Interface1 = 57.031 Interface2 =57.031	Aggregated rates = **110.43** 53.40 / 93.63% increased data rate

V. CONCLUSION

In this paper a bandwidth aggregation system which uses multiple active physical interfaces simultaneously is proposed. Depending on the user specified data rates of the interfaces this system distributes the amount of data to be transferred or received by the respective interfaces. The proposed system is based on the application layer of the TCP/IP stack where both TCP and UDP can be used as transport layer protocol. It does not require any corresponding module at the receiver side of the remote end system of the communication link or does not need any proxy node situated at the middle of the network for scheduling any packet to the multiple interfaces for simultaneous data transmission. It has components which take the user specified inputs and update the routing tables.

The experimental results have shown a significant performance improvement/bandwidth enhancement of the proposed system.

The limitation of the proposed system is runtime determination of bandwidth of physical interfaces depending on dynamical network condition; it needs a-priori knowledge of bandwidth of physical interfaces as input to perform optimally and fails to take full advantage of the available bandwidth pipe when the actual bandwidth varies significantly from the a-priori one.

The proposed system can be enhanced further based on cross layer communication model in automating the bandwidth estimation of each available link using dynamic bandwidth estimation.

There is also scope of further work in making the complete system application-agnostic so that the commonly used web applications like Browser based file download, upload or video streaming can take advantage of the aggregation by just installing a browser plug-in.

REFERENCES

[1] Nader Mohamed, Jameela Al-Jaroodi, Hong Jiang, and David Swanson, A user-level socket layer over multiple physical network interfaces; http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.92.1774.

[2] http://www.michaelbrumm.com/how-to-aggregate-bandwidth.html.

[3] Juan Carlos Fernandez, Tarik Taleb , Mohsen Guizani , Nei 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, October 2009.

[4] 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.

[5] Evensen, K.; Kaspar, D.; Engelstad, P.; Hansen, A.F.; Griwodz, C.; Halvorsen, P.; Simula Res. Lab., Lysaker, Norway "A network-layer proxy for bandwidth aggregation and reduction of IP packet reordering" in Local Computer Networks, 2009; http://simula.no/research/nd/publications/Simula.ND.369/simula_pdf_file.

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

[7] A. Habib, N. Christin, and J. Chuang, "Taking advantage of multihoming with session layer striping," in INFOCOM, 2006. http://www.andrew.cmu.edu/user/nicolasc/publications/gi-stripping.pdf

[8] A. Qureshi, J. Carlisle, and J. Guttag, "Tavarua: video streaming with WWAN striping," in MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia. New York, NY, USA: ACM, 2006, pp. 327–336

[9] http://www.buildwebsite4u.com/advanced/php.shtml