Flow-based Characteristic Analysis of Internet Application Traffic

Myung-Sup Kim[1], Young J. Won[1], Hyung-Jo Lee[1], James W. Hong[1], and Raouf Boutaba[2]
[1]Department of Computer Science and Engineering, Pohang University of Science and Technology
[2]School of Computer Science, University of Waterloo
{mount, yjwon, hyungjo, jwkhong}@postech.ac.kr, rboutaba@uwaterloo.ca

Abstract **–** The necessity of IP traffic analysis is growing dramatically in many areas. With increasing network demands from individual users as well as business communities, the Internet is overwhelmed by diverse and complex types of traffic from various network-based applications. To analyze this sophisticated Internet environment, efficient methods and tools are desperately required. This paper presents general characteristics of the Internet application traffic together with application-level analysis from the perspective of flows. We analyze the IP traffic traces collected on the Internet junction at POSTECH, a university with over 6,000 end hosts and servers. In addition, using our flow grouping method we classify the IP traffic according to the corresponding applications, and we explore the flow-level characteristics of current network-based applications.

Keywords **–** Passive Measurement, Internet Application Traffic, IP Traffic Analysis, Flow-based Analysis

I. INTRODUCTION

The need of network traffic monitoring and analysis is growing dramatically with the increasing network usage demands from individual users as well as business communities. Traditional areas, which heavily depend on traffic monitoring, are ranging from the network capacity planning to the study of network behavior. In addition, emerging new areas, such as SLA, CRM, security attack analysis, and usage-based billing, also have great needs for traffic monitoring and analysis. A number of current systems for traffic monitoring and analysis focus on flow-based investigation. The process of flow-based traffic monitoring starts from classification of packets according to the 5-tuple packet header values and generates flow data. Such systems (Ntop [1] and NG-MON [2]) capture raw packets from network links or network devices, and generate flow data with their own flow format. Cisco routers and switches are equipped with a function to export flow data in the NetFlow format [3]. In addition, IPFIX [4], a working group under IETF, is trying to define the standard IP flow format.

One of the key assets of the flow-based traffic analysis is to compress a significant amount of packet data into flows. However, the compression ratio from the packet data to flow is highly variable in the recent IP network environment. We believe that diversity and complexity of traffic generated by numerous applications, such as the traditional applications (e.g., Web, Ftp, Telnet, and etc.) and the newly emerging applications (e.g., P2P file sharing, streaming, and gaming applications), is responsible for the phenomenon of highly variable compression ratio. Moreover, the frequent appearances of abnormal traffic (e.g., Scanning, DoS/DDoS, and Internet worms) also contribute to a large number of traffic flows. The wild fluctuation of flow counts in current network environment negatively influences the performance of the traffic analysis system. The traffic analysis performance depends on lots of factors, such as link utilization, pattern of packet arrival, number of flows, etc. Among them, the number of flow counts influence the whole phases of traffic analysis system. It is essential to have a deep insight into flow-based IP traffic characteristics to understand the Internet traffic behavior and to improve the traffic analysis system performance.

This paper presents the characteristics of the Internet traffic from the perspective of flow by analyzing various flow measurement metrics with the traffic traces collected on the Internet junction at POSTECH. Using our flow grouping method we were able to classify the IP traffic according to the corresponding applications. We analyzed the flow-level characteristics of current network-based applications and compared with one another.

The organization of this paper is as follows. Section II describes the traffic data collection method and the flow grouping method. The flow-based analysis of IP traffic traces is mentioned in section III. Section IV illustrates the detailed analysis of each application traffic flows. Finally, concluding remarks are drawn and possible future work is discussed in section V.

II. IP TRAFFIC COLLECTION AND CLASSIFICATION

In order to collect IP traffic data, we used the NG-MON Flow Store [2], which is deployed on the Internet junction of POSTECH, as illustrated in Figure 1. NG-MON is a real-time traffic monitoring and analysis system for high-speed network links, which is developed by our research group from 2002. The role of NG-MON Flow Store is to receive the flow data from Flow

This work was supported in part by the Electrical and Computer Engineering Division at POSTECH under the BK21 program of the Ministry of Education, Korea.

Generators, store them for some time, and provide them to traffic analysis applications.

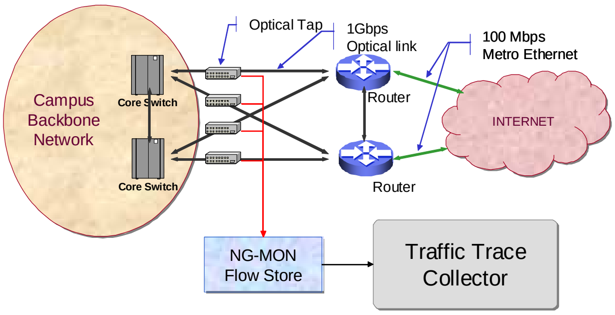

Figure 1. Traffic Trace Collection Method

To collect the IP traffic trace we developed a traffic trace collector system, whose function is to retrieve the flow data from the NG-MON Flow Store and keep the entire flow data for future on-line or off-line traffic analysis. In this study we define flow as a unidirectional stream of packets with same 5-tuple packet header values: source IP, source port, destination IP, destination port, and protocol number. We used 48 bytes to specify a single flow data, which is derived from the Cisco NetFlow V5 format [3] with some modifications.

For application-level detail analysis of IP traffic flows, we classified the flow data according to the corresponding applications using the Flow Grouping Method [5], which is developed by our research group. The Flow Grouping Method uses the correlation information among flows to determine the origin application name of individual flows along with the application specific well-known port number. The followings are the fundamental assumptions of grouping among flows:

① If a flow (f_a) belongs to an application group (A_a) , then the reverse flow, $({}_r f_a)$, of the flow (f_a) also belongs to the same application group, (A_a).

② If a flow (f_a) belongs to an application group (A_a) and the *sport*, *sip*, and *proto* a flow (f_b) are equal to the those of the flow (f_a), then the flow (f_b) also belongs to the same application group, (A_a).

③ If a flow (f_a) belongs to an application group (A_a) and the *dport*, *dip*, and *proto* a flow (f_b) are equal to the those of the flow (f_a), then the flow (f_b) also belongs to the same application group, (A_a).

④ Flows between the same two end hosts at the same time are generated by one application with some probability.

⑤ Flows generated by a single host at the same time are generated by one application with some probability.

Most newly emerging Internet-based applications use dynamic port numbers for significant amount of data transfer. Using this method, we could efficiently identify the application flows with dynamically generated ports as well as the application flows with static and well-known ports.

III. IP TRAFFIC CHARACTERISTICS

We collected IP traffic for three weeks during two months (Feb. and Mar.) in 2004. The overall summary of the two week traffic trace is illustrated in Table 1. The total number of flows captured during three weeks was 2,610x10^6, and the total bytes are over 40 TB. Among them, we considered only TCP, UPD, and ICMP traffic in the analysis categories, which occupies more than 99% of total traffic in bytes.

Location		Internet Junction of POSTECH Campus Network					
Collection Period		2/1/'04 – 2/7/'04			2/17/04 – 2/23/04		
Total File Size		41 Gbytes			43 Gbytes		
		Flows (x 10^6)	Packets (x 10^6)	Bytes (GB)	Flows (x 10^6)	Packets (x 10^6)	Bytes (GB)
Total	TCP	295 (34%)	18,345 (93%)	13,697 (98%)	325 (35%)	19,246 (92%)	13,619 (97%)
	UPD	537 (62%)	1,089 (5%)	190 (1%)	543 (59%)	1,381 (6%)	327 (2%)
	ICMP	33 (3%)	190 (0%)	16 (0%)	39 (4%)	177 (0%)	15 (0%)
	Others	0.1 (0%)	1 (0%)	0.6 (0%)	0.1 (0%)	1 (0%)	0.2 (0%)
	Total	866 (100%)	19,636 (100%)	13,905 (100%)	908 (100%)	20,806 (100%)	13,962 (100%)

Table 1. Traffic Trace Summary

The average bytes per packet were calculated as 642 bytes from Table 1. The average bytes per TCP packet was 678 bytes, which was greater than that of UDP traffic (239 bytes). The average packet count per flow was 28 (average TCP and UDP packet counts of flow were 98 and 3, respectively). We assume that a large number of UDP flows are composed of less than 3 packets. The average bytes per flow were 18,239 bytes. Average TCP and UDP bytes per flow were 67,043 and 756 bytes, respectively. TCP is used to transfer important and large amount of data between a client and server using its reliable service mechanism, while UDP is usually used to send short messages, the drop of which could be tolerable.

The ratio of TCP and UDP traffic in bytes and packets are similar to each other; over 90% of total packets and total bytes are TCP traffic. Still, TCP is used by the majority of current Internet applications. However, the flow ratio of TCP and UDP traffic is opposite to the previous case. The number of total UDP flows is about two times greater than the number of total TCP flows, as illustrated in Table 1. A small number of UDP packets with small bytes than TCP packets cause a significant amount of flows. This implies that the UDP traffic in the current network environment highly influences the flow-based traffic analysis system negatively, because the performance of these systems depends on the number of

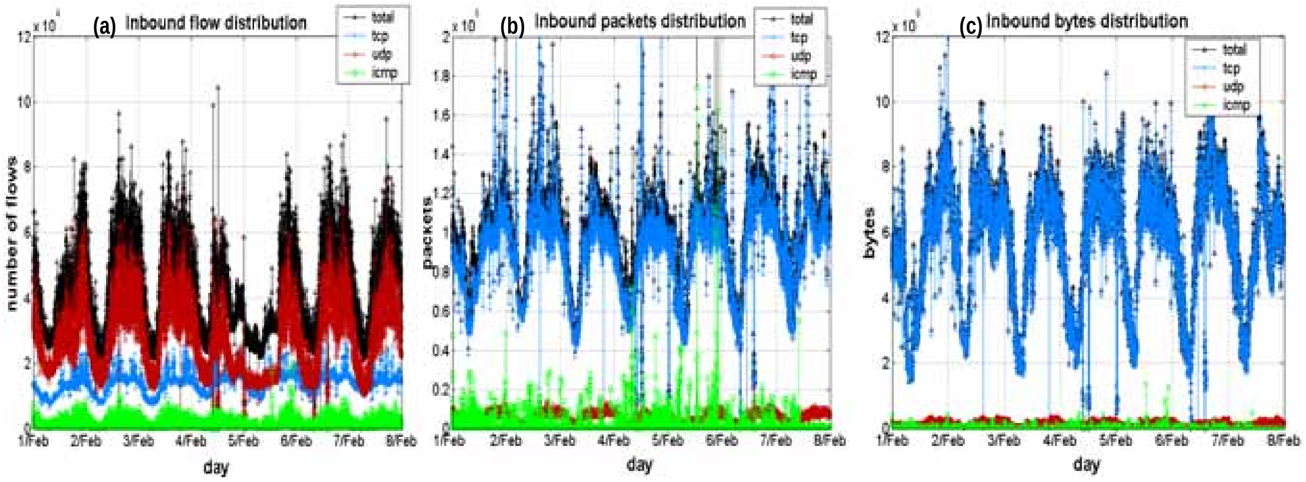

Figure 2. Time-series Graphs of Three Analysis Metrics (flow, packet, and byte) for Traffic Trace

generated flows rather than the number of packets and link utilization.

Another interesting fact about the IP traffic is that the inbound and outbound traffic shows almost one to one ratio in terms of flow count and packet count. Considering bytes, the total outbound traffic is 1.41 times greater than inbound traffic, which is commonly reported in many university networks [6]. This implies that the inbound packet size is smaller than the outbound packet size, and the inbound byte size of a flow is also smaller than that of outbound flow.

A. Distribution of IP Traffic over Time

Figure 2 illustrates three time-series graphs of the traffic trace. Each graph shows variance of inbound three-transport layer protocol (TCP, UDP, and ICMP) traffic and the sum of them in three analysis metrics (flow, packet, and byte). The outbound traffic variation is very similar to that of inbound traffic. The total flow distribution is mainly affected by the UDP flows, as illustrated in Figure 2(a). The inbound and outbound flow distribution has a similar shape and the average number of outbound flows is slightly larger than that of the inbound flow.

The shapes of packet distribution and byte distribution graphs are primarily affected by the amount of TCP traffic, which contradicts the shape of flow distribution. The time-of-day effect appears in all three kinds of graphs. The traffic increases from afternoon and marks to the peak between 10 p.m. and 1 a.m. of next day, and it goes down in the morning, which is a typical Internet usage behavior of our university since all of our students live in the campus dormitories. The fluctuation of incoming bytes is higher than the outgoing bytes. That is because the number of outside users are much higher than the inside users. In other words, the more users access a network, the less fluctuation of download traffic appears.

B. Distribution of Duration, Packets and Bytes in Flow

The average flow duration of TCP flows is 57.32 seconds, which is 5.3 times greater than that of UDP flows - 10.72 seconds. We can observe some long-lasting flows over 105 seconds (about 1 day). The median of TCP and UDP flow duration is 1 second, which indicates that the flow duration of more than 50% of total flows is less than 1 second. The number of UDP flows less than 80 seconds long is greater than the number of TCP flows. By contrast, above 80 seconds long TCP flows are much more than the number of UDP flows. The duration of TCP flows are more evenly distributed between 0 and 1000 seconds than UDP flows.

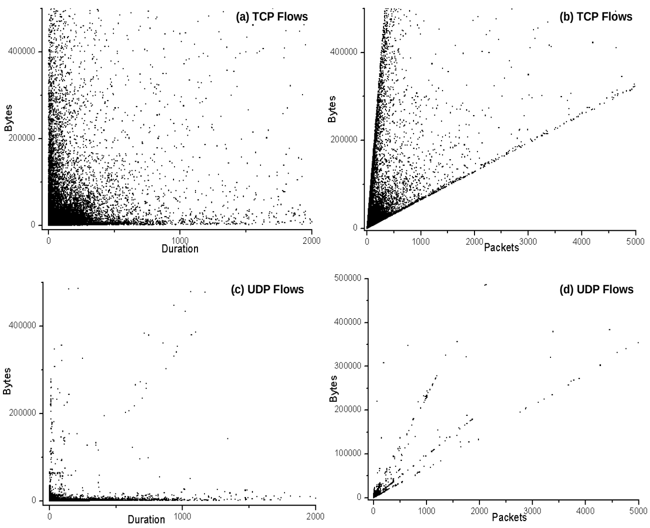

Figure 3. Relationship among Duration, Packets, and Bytes

The ratio of TCP flows with only 1 packet is about 6% of the total TCP flows, compared to about 76% in the case of UDP flows. The number of TCP flows with less than 1000 packets occupies a large portion of the total TCP flows. By contrast, the number of UDP flows with less than 10 packets takes a large portion of the total UDP flows. Particularly, the number of UDP flows with a couple of packets takes about 92% of the total UDP flows. Consequently, the number of packets belonging to the TCP flows is greater than the number of packets of UDP flows. Considering TCP flows, the bytes of TCP flows are evenly distributed until 1000 bytes with some fluctuation. Considering the flows having less than 1000 bytes, the number of TCP flows is 72% of total TCP flows. About 90% of TCP flows are composed of less than 4000 bytes. The 64 bytes UDP flows are 53%, which means that half of the total UDP flows are single packet flow. 90% of UDP flows are less than 200 bytes.

Figure 3 illustrates the relationship among three fields (duration, packets and bytes) in flows. Figure 3 shows that the bytes and packets in the flows are highly independent of the flow duration. The bytes in most UDP flows are less than 5000 bytes regardless of the flow duration. But the bytes of TCP flows are spread widely in the chart. The flows with large bytes and low duration and flows with small bytes and high duration appear together in this graph. The bytes in flows are proportional to the number of packets, as illustrated in Figure 3(b) and Figure 3(d). In the bytes vs. packets graph of TCP flows, two thick boundary lines appear, and all TCP flows lie between these two boundary lines: 64 bytes/packet line and 1500 bytes/packet line. We have a considerable amount of TCP flows with 1500 bytes per packet, while no UDP flows has this amount of bytes per packet value. Most UDP flows have less than 500 packets and 50,000 bytes.

IV. APPLICATION-LEVEL CHARACTERISTICS

Using Flow Grouping Method, we could determine the application name of IP traffic flows. The proportion of determined traffic from the total traffic trace was 99.5%, 94%, and 92% in terms of flows, packets, and bytes, respectively. The identification ratio of flows is greater than those of packets and bytes, because the proposed method is based on flow correlations.

We found an interesting fact that most IP traffic was generated by less than 100 applications. Table 2 shows the 10 heaviest applications in three perspectives of traffic metrics (flows, packets, and bytes). As Table 2 shows, the flow distribution does not follow the packet and byte distribution, while the packet and byte distribution is almost in accordance with each other.

Flows			Bytes		
Top 10 apps	ratio(%)	(In:Out) (%)	Top 10 apps	ratio(%)	(In:Out) (%)
eDONKEY	48.5	(51 : 49)	*eDONKEY*	24.2	(48 : 52)
SORIBADA	29.6	(50 : 50)	*HTTP-WEB*	18.1	(66 : 34)
V_SHARE	4.5	(54 : 46)	*FREECHAL*	9.5	(15 : 85)
HTTP-WEB	4.1	(49 : 51)	*FTP*	8.7	(21 : 79)
MSN	2.2	(50 : 50)	*V_SHARE*	5.8	(45 : 55)
BATTLE_NET	2.0	(11 : 89)	*MSN*	2.8	(45 : 55)
AFS	1.8	(50 : 50)	*mIRC*	2.3	(06 : 94)
DNS	1.4	(50 : 50)	*SORIBADA*	2.0	(35 : 65)
SAYCLUB	0.9	(50 : 50)	*BITTORENT*	2.0	(26 : 74)
FREECHAL	0.6	(50 : 50)	*WMedia*	1.3	(91 : 09)
Total	95.6	(50 : 50)	Total	76.7	(43 : 57)

Table 2. Top 10 Most Popular Applications in Flows, Packets, and Bytes

The top 10 most popular applications occupy the 95.6% of total flows, the 76% of total packets, and the 76.7% of total bytes, respectively. This indicates that the flow distribution is more skewed than the other two distributions. Six applications in flow distribution and seven applications in packet and byte distributions in the above table are P2P applications which use dynamic port numbers. Our results are in accordance with the results of several previous results in P2P traffic analysis [7, 8]. The Web traffic is still one of the most traffic-consuming applications, while the FTP application is less than web application. The world-wide P2P applications such as eDonkey and MSN Messenger occupy a large part of Internet traffic. In addition, the nation-wide P2P applications such as V_SHARE, FREECHAL, SAYCLUB, and SORIBADA are located in the top 10 list of three different distributions and occupy a large part of Internet traffic.

A. *Traffic Statistics of Selected Applications*

Among the top-10 applications, we have selected seven applications from the following categories: Traditional, P2P file sharing, instant messaging, and streaming applications. Web and FTP are representatives for the traditional applications which still take a significant portion of Internet traffic. SORIBADA (a Korean version of Napster), V_SHARE, and eDonkey are our choice of P2P file sharing applications widely used in Korea as well as in the rest of the world. MSN is the most popular instant messaging application without a question. Finally, we investigate the streaming media traffic of Microsoft's Windows Media application.

	Duration (sec)			Packet			Byte		
	min	max	average	min	max	average	min	max	average
WWW	0	120047	61	1	1,078E3	197	64	1.380E9	154603
FTP	0	77409	20	1	2.491E6	558	64	3.545E9	464653
Window Media	0	106554	337	1	658364	1696	64	9.868E8	1.806E6
eDonkey	0	42906	30	1	375090	21	64	4.819E8	14313
Soribada	0	85970	9	1	357403	3	64	6.625E7	626
V-share	0	26432	2	1	1.341E6	33	64	1.819E9	31271
MSN Messenger	0	141228	547	1	894156	239	64	1.326E9	173030

Table 3. Statistics of Selected Applications

Table 3 illustrates packets, bytes, and flow duration summary of each application. One interesting behavior of SORIBADA is to generate excessive number of flows which contain only a couple of small size packets. This characteristic results in the relatively low average number of packets per flow - 2.768 packets per flow. The mean number of packets in eDonkey’s flow is 21.49 packets which is slightly larger than SORIBADA’s. The architectural differences, such as node structure and search mechanism, between the two applications are responsible for this phenomenon. The maximum values of flow bytes of all the 7 applications are very high (over 10E7). This indicates that all these applications have a functionality to transfer large amount of data in a flow like FTP. P2P file sharing and instant messaging application support the data transferring functionality; they create full size packets within the data session, like the FTP’s data session. Streaming media applications also require sending out the full packets

Flow duration is low for P2P files sharing applications. We believe that short query and search messages are responsible for this phenomenon; the average flow duration of SORIBADA and V-SHARE is 9 and 1.98 seconds, respectively. The mean duration of Web is greater due to the user behavior of the Web browsing applications where they involve the frequent user interaction. However, eDonkey does not fall into this category due to the large user population in the campus and relatively well optimized query and search mechanism.

Finally, MSN messenger, an instant messaging application, has the longest mean value of flow duration among the selected seven applications (547 seconds). There is a better chance for longer flow duration if the application is capable of producing packets constantly with the short inter-packet generation time. Consequently, the MSN messenger service acquires constant buddy list updates (periodic interaction with the central server) and has usual user behaviors of chatting tools.

B. Number of packets and bytes of application traffic flows

Figure 4, Figure 5, and Figure 6 illustrate the relationship between bytes and packets for each application. One common characteristic of the plots is that they all have clear upper and lower boundaries The upper and lower bounds indicates the range of Ethernet frame size - 64 ~ 1500 bytes.

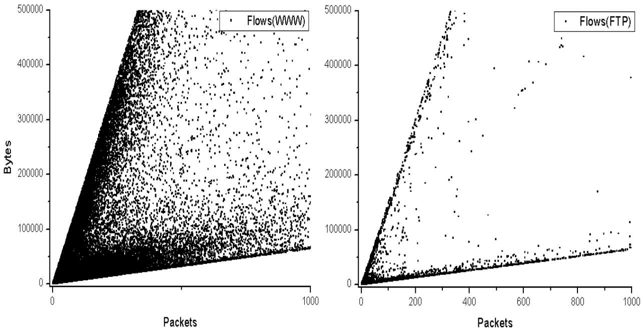

Figure 4. WWW vs. FTP

In traditional applications, the Web traffic consists of packets with wide range of byte sizes. Although it seems the density around the boundaries is quite high, it is simply because there are a large number of packets generated by Web (16% of all packets). On the contrary, FTP packets are concentrated on the two boundaries in Figure 4. This reflects the difference in bytes of the packets generated by two separate connection sessions of FTP: a control session and a download session. The control session usually contains simple command messages, such as start and stop, so the generated packets are small. The download session sends out the full packets (with maximum 1500 bytes). Thus most FTP related packets are either minimum or maximum of Ethernet frame unit.

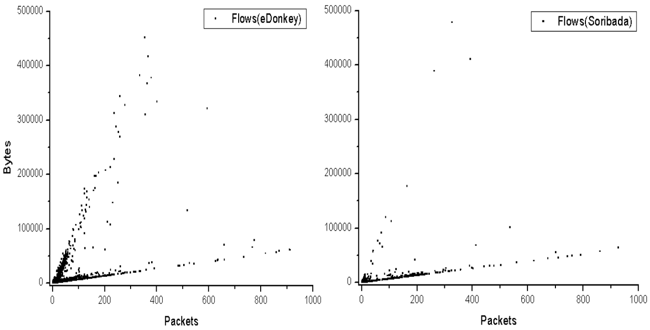

Figure 5. eDonkey vs. Soribada

In Figure 5, P2P file sharing traffic shows a somewhat similar shape to FTP traffic. However, there is less number of full packets than FTP traffic has due to the following reasons: unstable connection and low successful download ratios. Unlike FTP's stable connection, P2P systems can not guarantee a reliable connection with the content provider. Also, the connection speed varies to the network condition, so users have the tendency to frequently cancel the established download session. In the case of SORIBADA, one additional factor causes the phenomenon described above. The content being exchanged is mp3 music files which are relatively small, usually less than 10 Mb. Furthermore, packets, which stay close to the lower boundary on the graph, consist of query, search, and ping-pong messages of typical P2P applications.

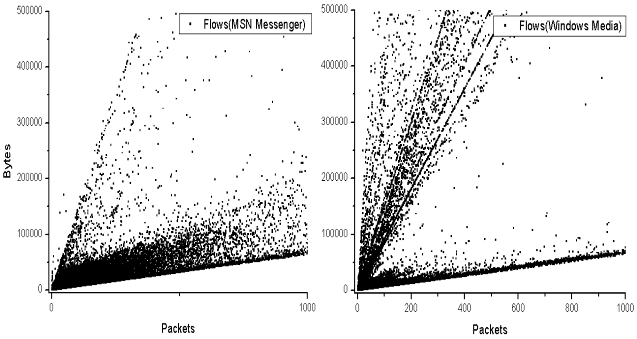

Figure 6. MSN Messenger vs. Windows Media

In Figure 6, we observe that there is high density of points near the lower boundary on the MSN messenger graph. This infers that a large portion of MSN messenger traffic consists of small size packets with simple text messages. We also monitor the full size packets that are used for P2P download sessions. In the Windows media example, the shape of the graph is again similar to FTP traffic. However, the wider distribution of packets around the upper boundary is present. We believe that rate control mechanism of streaming media applications is responsible for this distribution model. The data transmission ratio of the download session can be selected by the user or the provider (e.g. 100 Kbytes/sec, 300 Kbytes/sec, or higher) and has the influence on the size of data packets. In addition, some of the points are placed beyond the maximum bound, 1500 bytes. These points appear in the data set because we reassemble the fragment packets in the flow generator phase.

V. CONCLUSION

In this paper, we presented the IP traffic characteristics from the perspective of flow with the traffic trace from POSTECH Internet junction. We found that the flows with short time and small size occupies large amount of IP traffic, which can be problematic considering performance of traffic analysis system. Moreover, these short lived flows are mostly UDP flows and generated by new Internet applications, especially P2P file sharing applications. For future work, we plan to study the features of application traffic flows in more detail with long term traffic trace data.

REFERENCES

[1] Luca Deri, Ntop, http://www.ntop.org.

[2] Se-Hee Han, Myung-Sup Kim, Hong-Taek Ju and James W. Hong, "The Architecture of NG-MON: A Passive Network Monitoring System", Proc. of DSOM 2002, Montreal Canada, October 2002, pp. 16-17.

[3] Cisco Systems, "NetFlow Services and Applications," White Papers, http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.htm.

[4] IETF Working Group IPFIX (IP Flow Information Export), http://www.ietf.org/html.charters/ipfix-charter.html.

[5] Myung-Sup Kim, Hun-Jeong Kang and James W. Hong, "Towards Peer-to-Peer Traffic Analysis Using Flows", Proc. of DSOM 2003, Heidelberg, Germany, October, 2003, pp. 55-67.

[6] Stefan Saroiu, Krishna P. Gummadi, Richard J. Dunn, Steven D. Gribble, and Henry M. Levy, "An Analysis of Internet Content Delivery Systems", Proc. of OSDI 2002, Boston, MA, December 2002.

[7] Alexandre Gerber, Joseph Houle, Han Nguyen, Matthew Roughan, and Subhabrata Sen, "P2P The Gorilla in the Cable", Proc. of NCTA 2003, Chicago, IL, June 2003.

[8] Nathaniel Leibowitz, Matei Ripeanu, and Adam Wierzbicki, "Deconstructing the Kazaa Network," Proc. of WIAPP'03, June 2003.