Performance Analysis of Ip Switching and Tag Switching[1]

Gargi Banerjee, Rob Rosenberry, and Deepinder Sidhu

Maryland Center for Telecommunication Research
University of Maryland Baltimore County
1000 Hilltop Circle, Baltimore, MD 21250
{gargi, rosenber, sidhu}@cs.umbc.edu
Tel: 410-455-3063. Fax: 410-455-3964

Abstract. To cope with exponential growth of traffic on the Internet, IP networks require routers capable of much higher performance capabilities. This has led to interest in combining speed of layer-2 switching with scalability of layer-3 routing, leading to the generation of layer3 switches. Various technologies like IP, tag and Multi-Protocol Label Switching are being developed to provide these features by performing IP routing over ATM. In this paper we perform several experiments to analyze and compare the performance of IP switching and tag switching on networks comprising traditional IP routers running over an ATM backbone. Using network traces we compare the performance gains of IP switching to the gains obtained from tag switching. Our findings indicate that performance of IP switching is dependent on flow classification parameters and router speed. Also we find that there are higher performance gains with tag switching at the cost of moderately higher resource usage.

1. Introduction

The remarkable growth of the Internet has created a wealth of technical challenges. To meet the growing demand of bandwidth, Internet service providers (ISPs) need higher performance routing products. ATM has received much attention because of its high capacity, bandwidth scalability and its ability to support multi-service traffic. Thus, the need to evolve the routing functionality of the Internet and IP networks and the fact that ATM switches have the potential to provide great performance improvements over earlier network technologies have led to the desire to integrate IP and ATM. Several organizations have come up with different solutions for integrating IP network protocols into an ATM model. However, because of its connection oriented paradigm, the overlay model proposed by The ATM Forum for mapping

[1] This research was supported in part by the Maryland Center for Telecommunications Research (MCTR) at the University of Maryland, Baltimore County with funds received from the Department of Defense. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of any sponsor of MCTR.

G. Pujolle et al. (Eds.): NETWORKING 2000, LNCS 1815, pp. 752-762, 2000

connectionless IP onto ATM has led to complex and inefficient implementations. The Internet Engineering Task Force (IETF) has advocated an alternative peer model. Several companies have published Request For Comments (RFC) describing various approaches to the technique known as IP label switching. Label switching technology enables IP control protocols to run directly on ATM hardware. ATM switches still use label-swapping mechanisms to forward cells but employ IP control protocols to set up forwarding of packets and allocate resources. This approach has the advantage of requiring only standard IP routing protocols and not any new signaling or address resolution protocols.

In 1996, Ipsilon published their label switching technology, which they called IP switching and claimed large performance improvements over traditional router [1]. Concurrent with Ipsilon's work on IP switching, engineers from Toshiba described their cell switch router [7], Cisco Systems announced tag switching and IBM published drafts on Aggregate Route-based IP Switching (ARIS). In 1997, the Internet Engineering Task Force (IETF) decided to create a standardized label switching technique from the incompatible offerings of the different vendors. Multi-Protocol Label Switching (MPLS) has been adopted as the standard name for layer3 switching technique [8].

This paper analyzes and compares the performance of the label switching techniques of IP and tag. Sections 2.1 and 2.2 give some background on the two techniques. Section 3 introduces the event driven simulator, which we developed to perform the analysis and comparisons for label switching techniques. In Section 4 we examine our simulations of the two techniques and compare important statistics. Finally, in Section 5, we present our conclusions about IP and tag switching.

2. Background

2.1 IP Switching

Ipsilon designed IP switching based on the belief that the peer model for mapping IP onto ATM was too complex as two address spaces, two topologies, and two routing protocols must be managed. Ipsilon also wanted IP switching to preserve the connectionless and stateless behavior of IP networks [2].

All IP switches contain two control protocols to manage the stream of IP flows: the Ipsilon Flow Management Protocol (IFMP) and the General Switch Management Protocol (GSMP). IP switches also contain an inherent flow classifier. This component identifies flows and decides whether they should be bound to ATM virtual channels for possible switching. Two types of flows are defined [2]:

1. Flow Type 1: a flow of IP datagrams between two applications, identified by the source and destination IP addresses along with the source and destination ports in the IP packet header.
2. Flow Type 2: a flow of IP datagrams between two hosts, identified by the source and destination IP addresses in the IP header.

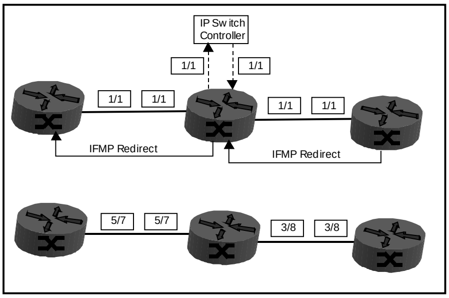

Fig. 1. IP Switch Changeover from Routing to Switching

Fig. 1 illustrates the redirection of IP packets from routing to switching. Initially, all IP datagrams are sent across the default virtual channel 1/1 between two IP switches. At each IP switch, the packets are routed by traditional IP routing mechanisms. When IP switch B decides to create a switched path for a flow, it sends its upstream neighbor switch A an IFMP Redirect message asking to assign a specific vpi/vci value 5/7 for the flow. Now A can set up a switched connection to B for this flow. At this point, all packets belonged to the selected flow are switched across the newly established connection instead of routed over that hop [3]. Similarly B's downstream neighbor C for this flow also sends a Redirect to B assigning a label 3/8 to the flow. Thus now a switched path is created for the flow from B to C. IFMP is a soft state protocol and the installed state at a switch needs to be updated periodically.

GSMP is a simple, generic switch management and control protocol designed to manage resource allocation within an IP switch. It is solely used internally within an IP switch to communicate switching commands to the controller for the ATM switch fabric. GSMP messages provide connection management, port management, statistics, configuration, and event notification functions [7].

2.2 Tag Switching

The major difference of Cisco's tag switching from IP switching is that while IP is a data driven approach, tag switching is control-driven [7]. A router that supports tag switching is called a Tag Switching Router (TSR). Tag switching consists of two components: forwarding and control. The forwarding component uses tag information carried by packets to perform forwarding. Every tag switch has a Tag Information Base (TIB). Each entry in the TIB contains the incoming tag and one or more sub-entries of the form outgoing tag, next hop, and outgoing interface [4]. When a packet with a tag is received by a TSR, the switch uses the tag as an index in its TIB. If it finds an entry with the incoming tag equal to the tag carried in the packet, then it replaces the tag in the packet with the outgoing tag. It also replaces the link level information in the packet such as a MAC address with the outgoing link level information, and forwards the packet over the outgoing interface. [4]

The control component is responsible for creating tag bindings, and then distributing that information among the tag switches in the network. Distribution of the tags may be carried out by piggybacking binding information on existing routing protocols or by using a separate Tag Distribution Protocol (TDP). There are three methods for tag allocation and TIB management:

1. Downstream tag allocation: the tag carried in a packet is generated and bound to a routing prefix by the tag switch at the downstream end of the link with respect to the direction of data flow.
2. Downstream tag allocation on demand: tags are only allocated and distributed by the downstream tag switch when it is requested to do so by the upstream tag switch.
3. Upstream tag allocation: a tag switch creates tag bindings for outgoing tags and receives bindings for incoming tags from its neighbors.

2.3 Multi-protocol Label Switching (MPLS)

MPLS was proposed by IETF as the standard for label switching technology. Since it combines layer three routing with layer two switching it is said to lie at layer two-and-half in the network stack. The MPLS label distribution protocol (LDP) operates in two modes- downstream on-demand distribution and downstream unsolicited distribution. Even though it resembles tag switching closely, MPLS has today evolved as one of the most powerful traffic engineering tools for next generation networks. LDP can also be extended to perform constraint-based routing (CR-LDP), where constraints may be an explicit route or a policy based route. Thus MPLS promises to have a distinct role in providing quality of service across the Internet.

3. Label Switching Simulator

The simulator that we designed and implemented for the analysis of label switching techniques using ATM is an extension of the NIST ATM/HFC Network Simulator [12]. Our discrete event simulator models an IP label switching network using existing components such as B-TEs, ATM switches, and high speed links. We have designed label switching specific components like IP switches, tag edge routers, and tag switch routers. The IP switch component models the operation of the IFMP protocol. It does not implement the GSMP protocol as that protocol operates entirely within the IP switch. Both type1 and type2 flows may be recognized by the switch. The flow classification algorithm, which we have implemented, uses a static X/Y classifier [1]. For our simulations, we have used values of 5 packets for X and 10 seconds for Y with a 30-second timeout. The tag edge router component for the simulator is the router at the edge of the tag switched network. It receives untagged packets from the traffic generator and tags them with labels from its TIB. It also participates in TDP by sending out BIND_REQUESTS to its neighbors for each route entry in its FIB. When it receives BIND_INFO messages from other TSRs, it makes an entry in its TIB for that route. Since our simulator is based upon an ATM network, the tag edge router also performs the added function of segmentation and reassembly of IP packets into ATM cells using AAL5.

4. Performance of Label Switching Techniques

In order to test the performance of our simulated label switches, we obtained IP trace files from the Association of Computing Machinery's Internet Traffic Archive. These files were collected from real traffic flowing into and out of the Lawrence Berkeley Laboratory in 1994 and Digital Equipment Corporation in 1995. [1] Indicates that the 'DEC1' and 'DEC2' files contain 2,983,221 and 3,467,733 packets respectively with an average of over 800 packets per second. 'DEC1' has 75,438 Ipsilon type 1 flows and 45,560 type 2 flows. The 'DEC2' file has 75,739 type 1 flows and 37,252 type 2 flows.

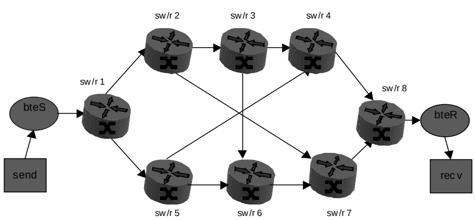

Fig. 2. Network of 8 switches

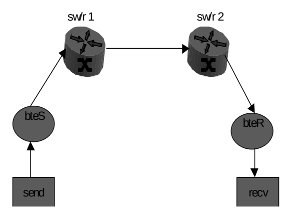

Fig. 3. Network of 2 switches

Fig. 2 and 3 illustrate the label switched networks that we used in our simulations. The 'send' component is the traffic generator for the simulations. It is responsible for injecting the packets from the trace file into the network at the appropriate time based upon a timestamp attached to each packet. The 'bteS' and 'bteR' components are artifacts of the design of the simulator and not relevant to the performance of this simulation. The 'SW/R' components are the label switch routers. They are either IP switches or a combination of tag edge routers and tag switch routers depending upon the label switching technique being simulated. The 'recv' component receives packets sent through the network.

4.1 Single Switch Performance

In the first experiment we verified the IP switching results as presented in [1]. Since these results were obtained for a single switch, we also used the network configuration of Fig. 3 where cells were switched only at a switch1. The trace files used in this experiment had also been used in [1].

Fig.s 4.1a-b shows the percentage of datagrams switched for both type1 and type2 flows using all the corporate traces. The results show that about 80-85% of datagrams is switched for all the traces. Fig. 4.1c shows how larger flow creation delays decrease percentage of switched datagrams. Fig. 4.1d indicates that a flow deletion delay of 50-60secs achieve a switching percentage of 75-80% for both type1 and type2 flows. All these results were found to be similar to results observed in [1].

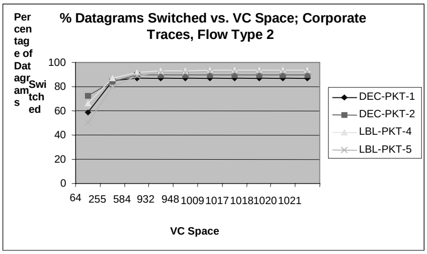

Fig. 4.1a. Effect on VC space (type2)

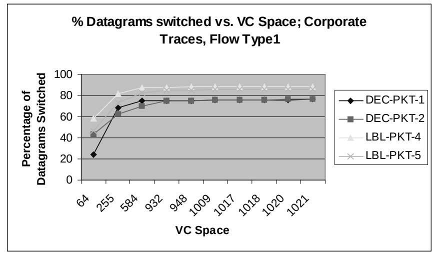

Fig. 4.1b. Effect on VC space (type1)

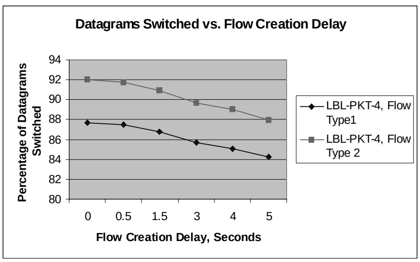

Fig. 4.1c. Effect of flow creation delay

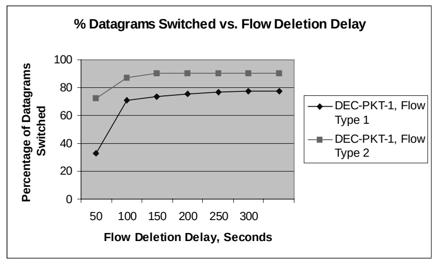

Fig. 4.1d. Effect of flow deletion delay

4.2 Effect of Router Speed on Switching Performance

In this experiment we investigate the effects on routing speed on IP and Tag switching respectively. The network configuration of Fig. 2 is used for this experiment. Only type1 flows are considered. Router processing time is the time taken by a router to process each incoming datagram. This mainly includes delay due to re-assembly of ATM AAL5 cells into an IP datagram, layer 3 routing lookup based on destination address in the IP header as well as flow classification decisions, if necessary and fragmentation of IP datagram back into ATM AAL5 cells. The input and output buffer delays are not considered as part of the router processing delay. For data-driven switching approaches, the percentage of switched datagrams is largely dependent on flow classification routines, and hence on the second factor in the delay component. We used two values of routing delay for this experiment: 100 ms and ms.

Fig. 4.2a and Fig. 4.2c show the percent of packets that are switched instead of routed at the edge switch and a representative interior switch respectively. The performance of the interior IP switch drops dramatically from over 80% packets switched to fewer than 20% while the performance of the tag switch remains constant. Fig. 4.2b and Fig. 4.2d show the performance of the same label switches using values of 1 millisecond for the routing delay and 10 microseconds for the switching delay at each of the label switches. In this case, the operation of the interior IP switch remains above 80% packets switched while again the tag switches maintain constant 100%

packets switched. The results showed that while for tag switching the percentage of switched datagrams remained the same, for IP the performance gain obtained from the percentage of switched datagrams reduced considerably with the large routing delay. This is due to the fact that IP being a data-driven approach, flow classification is done as and when data comes in. Since a large routing delay, implies a slow flow-classification module, hence most of the datagrams are already sent to route-buffers before a switching decision can be taken. Once the packets reach route buffers they have to be routed, and hence the switched percentage drops. However in a control driven approach like Tag, once the tags are set up, all cells may be switched and there is no need for any routing decisions. Hence the tag switching performance remains largely independent of the large router processing delay.

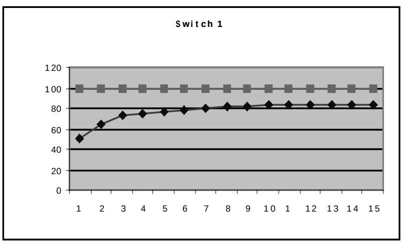

Fig. 4.2a. Edge Switch–100ms delay

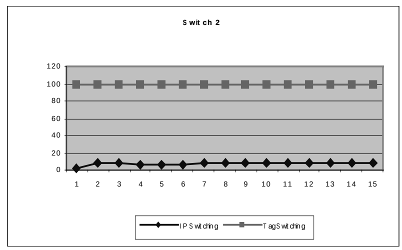

Fig. 4.2c. Interior Switch-100ms delay

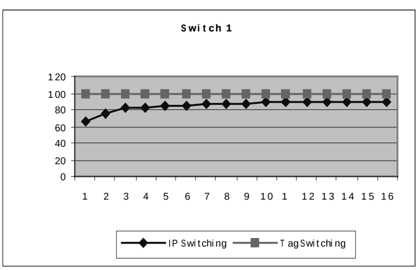

Fig. 4.2b. Edge Switch–1ms delay

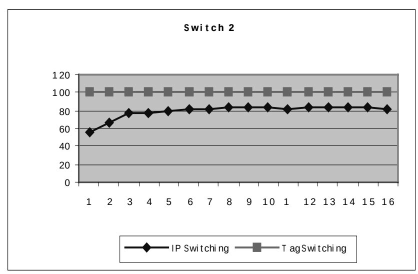

Fig. 4.2d. Interior Switch–1ms delay

4.3 Type1 vs. Type2 Classification

In this experiment we investigated the performance of two different types of flow classification for IP switching using the DEC-PKT1 trace file. The configuration of Fig. 2 is used for this experiment.

From the graphs 4.3a-e we see that flow type 2 achieves a higher switching percentage, a lower VC usage and a marginally lower average packet delay, when compared to flow type 1. The drawback though is that with flow type 2, due to larger number of datagrams being sent on the same circuit, delays due to network congestion may offset some of the benefits of having a type2 classification.

4.4 IP Switching vs. Tag Switching

In this experiment we compared the performance gains obtained from tag switching versus those obtained from IP switching. All the runs in this experiment were for trace files DEC-1and DEC-2. The network configuration of Fig. 2 is used for this experiment. Only type1 flows are considered. The comparison was based on the percentage of cells and packets switched VC space usage, message overhead and average packet delay.

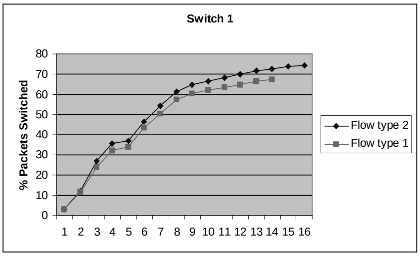

Fig. 4.3a. % packets switched at switch1

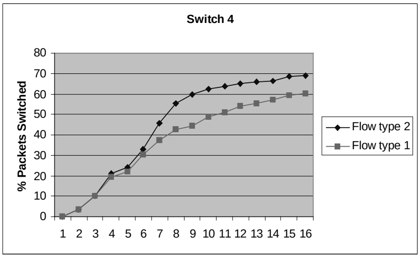

Fig. 4.3b. %packets switched at switch4

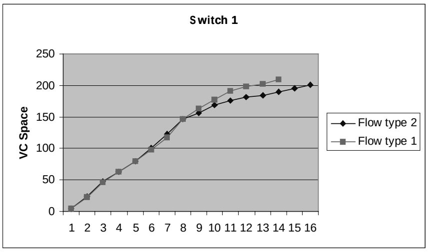

Fig. 4.3c. VC space usage at switch1

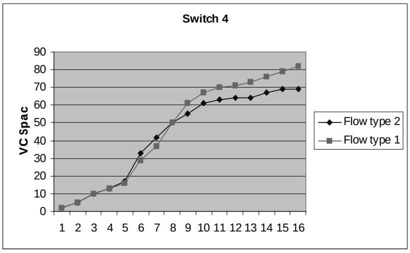

Fig. 4.3d. VC space usage at switch4

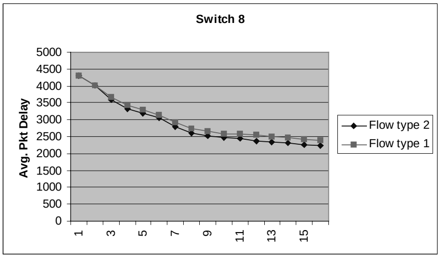

Fig. 4.3e. Average Packet delay at egress switch8

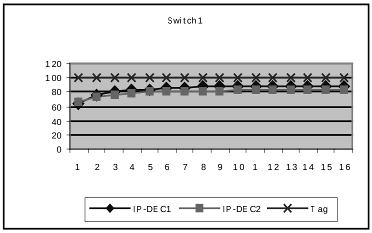

Fig. 4.4a. Switching at switch1

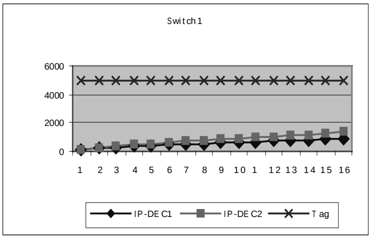

Fig. 4.4c. VC-space usage at switch1

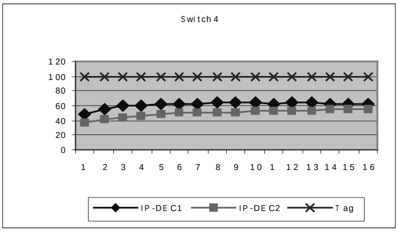

Fig. 4.4b. Switching at switch4

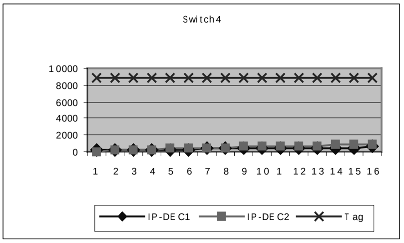

Fig. 4.4d. VC-space usage at switch4

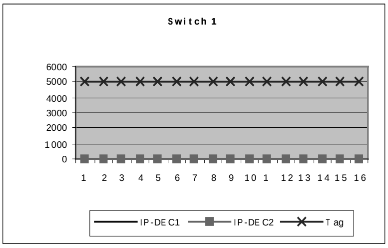

Fig. 4.4e. Message Overhead at switch1

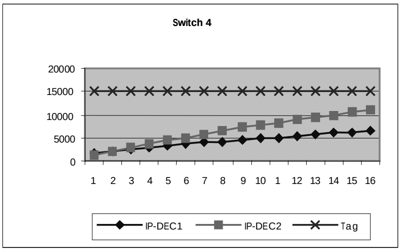

Fig. 4.4f. Message Overhead at switch4

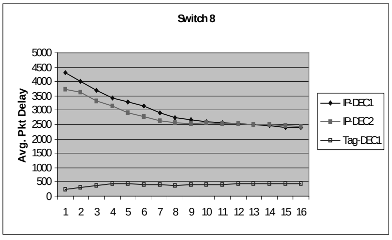

Fig. 4.4g. Average Packet Delay at egress switch8

Fig. 4.4a, b graphs the performance of the ingress (switch 1) and a representative interior switch (switch 4) in our simulated network for the two DEC trace files. Similar results were noted for the other interior switches and for other trace files. The traces on each graph show how each label switching technique performed against the three IP trace files that we used. Only the DEC-1 trace is drawn for the tag switching method as all of the trace files performed identically. The IP switching technique performed differently for each file since the number and length of flows changed from one trace file to another. The graphs show that the tag switching technique performs consistently against any type of traffic patterns, as it is a control driven label-switching technique. All switched paths are pre-established based upon the routes in the forwarding databases of the switches. Since the IP switching technique is data driven, it is more sensitive to variances in the data flows. The ingress and egress switches achieve approximately 80% packet switching while the interior switches only manage from 40% to 70% switching of packets. The tag switching technique maintains a very high amount of VC space utilization as shown in 4.4c, d. One virtual channel is allocated for every route path through the switch. The VC space used remains constant since our route tables are static. For the IP switching technique, the VC space utilized begins at zero and grows as the number of switched flows increases. However, since a VC is released as the traffic in the flow diminishes, the total number of VCs in use at any point remains much lower than with the tag switching method. Fig.4.4e, f represents the message overhead incurred to establish and maintain switched traffic throughout the network. The amount of overhead traffic is shown to directly relate to the amount of VC space utilized by the label switching technique. Since the tag switching mechanism uses more virtual channels, its message overhead is also correspondingly higher. Fig.4.4g shows the average packet delay across the network. This parameter measures the average end to end delay encountered by a packet and is measured at sample intervals spaced 1 sec apart. From the measurements taken during our simulations in experiment4, we conclude that the tag switching technique results in higher performance across the network. For a five to ten-fold increase in the number of virtual channels used, we gain a 20% to 40% performance increase. With today's ATM switching equipment, an increased VC utilization translates into higher memory requirements for the switch table and VC control storage. The switch table lookup hardware must also be able to handle searching through larger tables. However, we believe that this approach is much more feasible than trying to increase the route processing hardware. Trying to improve either software based routing system or hardware based routing mechanism is more complicated and difficult to achieve than improving the switching components.

5. Conclusions

In this section, we present our conclusions based on the results obtained from our simulations of two label switching techniques. First we have succeeded in validating results of [1] which mentions an overall 80% switching in IP switches. We have also shown that switching performance of a data driven approach depends largely on the routing capacity of the layer 3 switches. A slow router with a processing time in the order of 100ms can to a large degree offset the advantages obtained form label switching. In the third experiment we have seen that if quality of service is not so much of a concern, then IP switching with type 2 classification achieves high level of switched datagrams with a lower channel usage. Finally, we compared the performance of Cisco Systems' tag switching and Ipsilon's IP switching methods. The tag switching technique delivers higher performance in percent of packets switched at a higher cost of more virtual channels used and increased message overhead. We believe that the higher performance gained with the tag switching system are worth the associated costs.

As future work, our IP Switching and Tag Switching studies can be extended to Multi-protocol Label Switching (MPLS) which is being proposed as an IETF standard for the Internet. The issues to be investigated are: network throughput, latencies due to Label Distribution Protocol (LDP) and its implications, mapping MPLS onto ATM Quality of Services, traffic engineering using MPLS labels, support for multicast with MPLS and security in MPLS networks.

References

[1] Steven Lin, Nick McKeown. „ A Simulation Study of IP Switching.“ *Proc. ACM SIGCOMM '97*, France, Sept. 1997.

[2] Peter Newman, Tom Lyon, Greg Minshall. „ Flow Labelled IP: A Connectionless Approach to ATM. Proc. *IEEE Infocom '96*, vol. 3, pp.1251-60.

[3] P. Newman, W.L. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, G. Minshall. „ Ipsilon Flow Management Protocol Specification for IPv4.“ *IETF RFC 1954*, May 1996.

[4] Yakov Rekhter, Bruce Davie, Dave Katz, Eric Rosen, George Swallow. „ Cisco Systems' Tag Switching Architecture Overview.“ *IETF RFC 2105*, Feb. 1997.

[5] Hao Che, San-qi Li, Arthur Lin. „ Adaptive Resource Management for Flow-Based IP/ATM Hybrid Switching Systems.“ *IEEE/ACM Transactions on Networking*. Vol. 6, No. 5, Oct. 1998.

[6] R. Cole, D. Shur, C. Villamizar. „ IP over ATM: A framework document.“ *IETF RFC 1932*, Apr. 1996.

[7] Bruce Davie, Paul Doolan, Yakov Rekhter, "Switching in IP Networks: IP Switching, Tag Switching, & Related Technologies", 1998.

[8] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow, A. Viswanathan, "A Framework for Multiprotocol Label Switching", Work in Progress, July, 1999.

[9] E. Rosen, A. Viswanathan, Callon, "Multiprotocol Label Switching Architecture", Work in Progress.'98

[10] B. Davie, J. Lawrence, K. McCloghrie, Y. Rekhter, E. Rosen, G. Swallow, P. Doolan, "Use of Label Switching With ATM", Work in Progress, September, 1998.

[11] Juha Heinanen, " Multiprotocol Encapsulation over ATM Adaptation Layer 5." *IETF RFC 1483*, July 1993.

[12] NIST, „ The Nist Simulator ATM/HFC Network Simulator. Operation and Programming.“