International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Dynamic Packet Scheduler for Queuing Real Time and Non Real Time Internet Traffic

P. Shanmugaraja, K. Chokkanathan, J. Anitha, A. Parveen Begam, N.Naveenkumar

Abstract: Real-time traffic or flows also called as inelastic traffic is that which enforces timely delivery of flows within a specified period of time. There are several applications which generate these flows like multimedia, audio-video conferencing, webinar, Interactive gaming, webcam, Internet TV etc. An interactive application demands a speedy delivery of flows. Flows reaching the destination after this deadline are considered useless. Real-time traffic imposes rigid demands for the delivery. All the real-time flows should be timely delivered and accumulated at the destination. The proposed Dynamic Scheduler relies on Dynamic Packet Scheduling ratio which is dynamic in nature and the ratio changes dynamically with respect to the flows accumulated in both the queues. Flows are scheduled based on the maximum flows allowed on the path that is calculated by TP max. Packet scheduling is based on the available throughput of the network. This dynamic scheduling results in guaranteed fair treatment of both real-time traffic and non-real time traffic. In the this paper we propose a Dynamic Scheduler, [DPS] which dynamically works according to the available number of flows in both real-time and non-real-time queues.

Keywords: Real-time flows, Non-real-time flows, Scheduler, Queues, Throughput, Scheduling Algorithms, Bandwidth hogging, Jitter, Delay

I. INTRODUCTION

User satisfaction is a key factor for calculating the efficiency of a network. Usually efficiency is calculated using bandwidth, packet loss, and jitters and delays [5]. The functions of a real time application is significantly affected by the delay in the delivery of real time traffic flows [5]. Source node generates real-time flows (synchronous traffic) continuously to the destinations. Such flows consist of periodic and intermittent messages that are characterized by rigorous timing constraints. Periodic flows are generated at regular time intervals whereas intermittent flows are generated at irregular intervals[6].

Manuscript published on 30 September 2019
* Correspondence Author
P. Shanmugaraja, Department of Information Technology, Sona College of Technology, Salem, India
K. Chokkanathan, Department of MCA, Madanapalle Institute of Technology and Science, Madanapalle, India
J. Anitha, Department of Computer Science and Engineering, Dhayanand Sagar Academy of Technology and Management, Bangalore
A. Parveen Begam, Department of Computer Science and Engineering, Taibah University, Yanbu, Saudi Arabia.
N. Naveen Kumar, Department of MCA , Madanapalle Institute of Technology and Science, Madanapalle, India

Real time traffic needs instant delivery of flows and it cannot tolerate delays in transmission over the network. In this current Internet era ISP's give high priority to real time traffic over non–real-time traffic flows. Applications that generate Non-real-time traffic can adapt to delays in the network [5].

There are number of scheduling mechanisms such as FIFO, Simple Priority queuing, generalize processor sharing, Weighted Round Robin, Deficit Round Robin, Weighted fair queuing, Least attained Service etc[3]. These algorithms do not fairly and dynamically schedule the Internet traffic (both real time and non real time flows)[8]. The prediction of dynamic network conditions is not available in the above said scheduling disciplines. The proposed method schedules the real time traffic flow as swift as possible and also fairly treats the packets seated in the non real time traffic queues.

A scheduler should satisfy the demands of current internet applications including SMTP, HTTPS, FTP, Telnet, and (P2P) [2]. Scheduler must offer guaranteed service than the best effort service which that exist on today's internet. The motivation of the suggested Dynamic Scheduler is therefore to discover an alternative scheduling system to fulfill the demands of internet communication today[7].

II. RELATED WORK

Existing algorithms including Deficit Round Robin (DRR), Weighted Fair Queuing (WFQ), Least Attained Service (LAS), and Deficit Round Robin-Short Flow First (DRR-SFF) performs per-flow scheduling [4]. In the per-flow scheduling approach, the scheduler should maintain its flow's state [3]. Also, a complex mechanism is required for flow identification. With the tremendous growth in internet communication, it is impractical to maintain the state of all flows in routers. The benefit of the suggested system is that it does not require flow identification and does not require routers to maintain the flow state. The scheduler must be extremely scalable in the Internet because of its volume. Due to the huge amount of flows in the Internet, the flow control system faces a scalability problem. This problem can be overcome by an aggregated flow scheduler i.e., Dynamic Scheduler.

The existing scheduling algorithm, DRR-SFF, uses the Two Queue (TQ) approach with a constant packet scheduling ratio. DRR-SFF schedules both short and long flows with equal priority. Hence, this approach does not satisfy the needs of current internet characteristics [5].

The proposed DS also uses the TQ approach, but with the Dynamic Packet Scheduling Ratio, DPS(i), which adapts to the nature of the flows in the two queues to satisfy the needs of current internet communication.

III. PROPOSED DYNAMIC SCHEDULING ALGORITHM

Although there are numerous scheduling techniques such as FIFO and WFQ, these techniques do not handle packets based on the network climate. Dynamic Packet Scheduling ratio is calculated by considering the flow count residing at both the queues. Moreover, flows are scheduled based on the maximum flows allowed on the path that is calculated through TP_{max}. TP_{max} calculates the Dynamic Packet Scheduling Ratio DPS(i) for each iteration using the available network bandwidth and also schedules the number of flows waiting in the queue. Using the estimated value of available bandwidth, the scheduler calculates the maximum number of packets available on the network path TPmax. Fmax is related to DPS(i). With the total number of flows in RTQ and NRTQ, a variable Dov (i) is initialized. The value DPS(i) decreases Dov(i) whenever the Dynamic Scheduler schedules flows from the queue.

The Dynamic Scheduler schedules real-time traffic flow as quickly as possible and the packets seated in the non real-time traffic queues fairly. A scheduler requires simple operations to schedule selection of the next packet or service [25]. The proposed algorithm uses Dynamic Packet Scheduling Ratio *DPS(i)* to process flows, which can be calculated easily by observing the number of flows in both the queues. This removes complexity from the deployment of the algorithm in all routers.

Dynamic Scheduler prevents bandwidth hogging issue in heterogeneous networks. It also avoids the starvation problem that is seen in LAS [24]. The Dynamic Scheduler completely eliminates the flow loss for non-real-time flows. The packet loss minimization achieves less re-transmission of TCP that leads to lower the mean transmission of all the flows in Dynamic Scheduler.

Algorithm for Dynamic Scheduler

Output: Packets Scheduled based on DPS
Input: Number of packets

1. Begin
2. Initially determine the total number of flows in both queues RTQ and NRTQ.
3. Calculate the number of packets in RTQ= $\sum_{i=1}^{n} B_{RTQ}(i)$ and number of packets in NRTQ= $\sum_{i=1}^{n} B_{NRTQ}(i)$
4. Initialize the Dynamic Offset variable
   Dov(i) = $\sum_{i=1}^{n} B_{RTQ}(i)$
5. The next step is to calculate the Dynamic Packet Scheduling ratio DPS(i) for the iteration r.
   $$DPS(i) = \frac{\sum_{i=1}^{n} BRTQ(i) + \sum_{i=1}^{n} BNRTQ(i)}{\sum_{i=1}^{n} BNRTQ(i)}$$
6. Estimate the available throughput of the link using the formula TPmax=Recv buffer size/RTT
7. If TPmax> DPS(i)
   a. If true then RTQ=DPS (i) and NRTQ = 1.
   b. If $\sum_{i=1}^{n} B_{RTQ}(i)$ ==0, then total flows to be scheduled in the queue is RTQ = DPS(i).
   c. Dov(i)=Dov(i)-DPS(i).
   d. If Dov(i)>DPS(i), then goto step 4 or else return to step 1 for the purpose of dynamic calculation of DPS(i) and to find the RTQ for the next round of scheduling.
8. If TPmax<DPS(i)
   a. RTQ=TPmax and NRTQ = 0 (number of flows to be scheduled)
   b. Dov(i)=Dov(i)-DPS(i)
   c. If Dov(i)>DPS(i), then goto step 4 or else return to step 1 for the purpose of dynamic calculation of DPS(i) and to find the RTQ for the next round of scheduling.

The number of packets in RTQ is assigned to DOV. The scheduler calculates DPS(i) by monitoring RTQ and NRTQ. Maximum number of flows is calculated by DOV with the help of TPmax.

If TPmax>DPS(i), then DPS(i) = RTQ and NRTQ=1. This condition is checked or iterated until RTQ =0, if RTQ=0 then there is no flow in NRTQ. The scheduler checks the condition Dov(i)<DPS(i) as the packets are scheduled in RTQ. If the scheduler does not detect this condition, then it calculates a new value of DPS(i) and TPmax for the next iteration.

If TPmax<DPS(i), then the number of flows to be in RTQ=TPmax and NRTQ = 0. This dynamic scheduling results in guaranteed fair treatment of both flows. If the scheduler gives importance to only RTQ[30], then there will be starvation in NRTQ and packets will get dropped. Scheduler treats NRTQ fairly. Dynamic Scheduler dynamically changes according to the available number of both real-time and non-real-time traffic flows. Also, the scheduling is based on the available throughput in the network.

IV. EXPERIMENTAL SETUP

The performance of the proposed Dynamic scheduling algorithm in a heterogeneous network is tested by connecting to a single congested router. To analyze the bandwidth hogging problem, the performance of the heterogeneous network is evaluated by varying RTT. Dynamic Scheduler performance is compared to other protocols like FIFO, LAS, and RuN2C in a heterogeneous network. Dynamic scheduler is evaluated based on certain parameters. These parameters include throughput, flow loss and mean transmission time.

A. Heterogeneous Network With Single Congested Router

Bandwidth hogging is an important parameter to analyze in varying propagation delay heterogeneous network. Internet has different link capacities with different propagation delays.

Internet traffic flows through different paths. Throughput utilization for competing flows is reduced due to TCP's calculation of available throughput. TCP's available throughput is inversely proportional to RTT of the established connections. TCP allocates higher amount of throughput to connections with lower RTT and lesser amount of throughput to higher RTT.

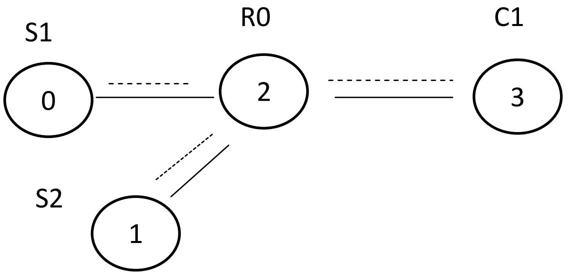

Fig. 1. Heterogeneous Network with Single Congested Router for Varying Propagation Delays

The heterogeneous network shown in Figure 1 is used to evaluate the working of Dynamic Scheduler. Real-time flows are non bursty and constant in nature where as non-real-time flows are flooded into the network. S1 and S2 are the two nodes which generates flows to the receiver C1. The flows propagate through a link with bottleneck (R1-C1). The propagation delay or RTT value of each link is varied. The access link between S1-R0 is set to 5 Mbps with a propagation delay of 100ms and the link between S2 –R0 is set to 10Mbps with a propagation delay of 10ms.

The bandwidths of the access links are lower in S1 in which the arriving real time flows data rate is less bursty whereas in S2 the arriving non real time data rates are bursty. This connection completely utilizes the bottleneck capacity of the link.

B. Throughput Analysis of FIFO with Different RTT

In a heterogeneous network, the FIFO scheduler is chosen by varying RTT for evaluation. The reason for choosing FIFO is that, according to their arrivals, it adopts the service. FIFO ignores flow control for all the services. At time 0, s2 starts and at time 10 s1 starts. Simulation results of FIFO with drop tail mechanism are shown in Figure 2. These data indicate that the burst y packets from S2 receive more service than S1.

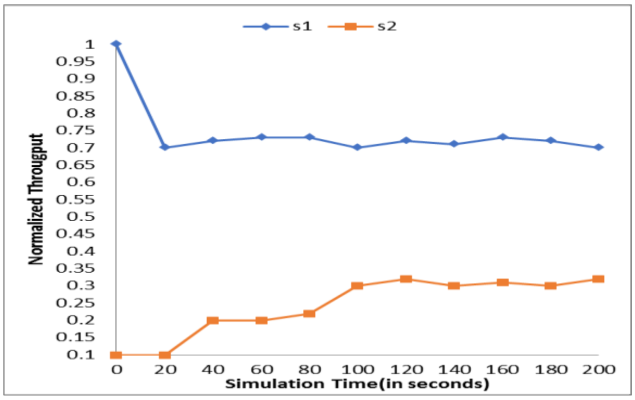

Fig. 2. Normalized Throughput versus Simulation Time of FIFO Scheduler by Varying RTT in Heterogeneous Networks

Hence, S2 utilizes most of the bottleneck bandwidth (i.e., bandwidth hogging). FIFO provides more bandwidth for the source s2 which is transmitting over low RTT. Whereas bandwidth of RTT S1 is lower. FIFO does not make any distinction between packets, regardless of whether coming from source S1 or S2. All packets are serviced, based on arrival time.

C. Throughput Analysis of LAS with different RTT

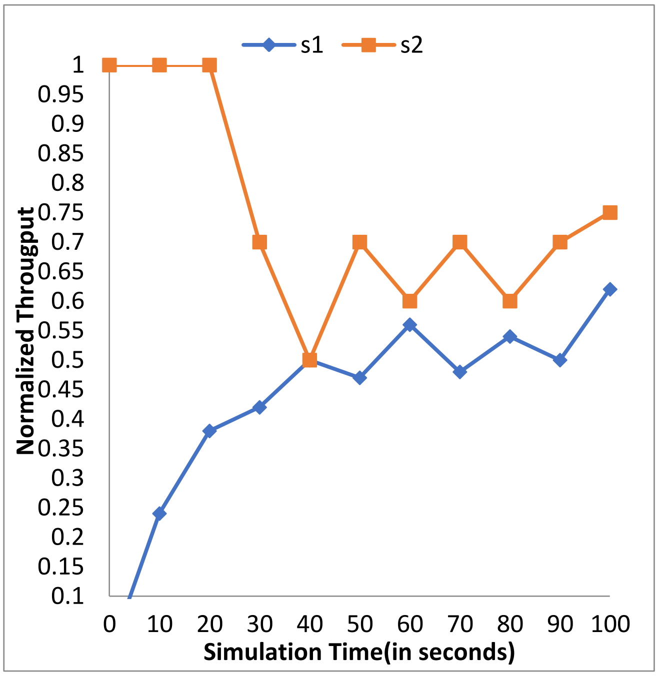

Fig. 3. Normalized Throughput versus Simulation Time of LAS scheduler by varying RTT in Heterogeneous Networks.

As shown in figure 2, S1 consumes 30% of the bandwidth while S2 utilizes 70% of the bandwidth.

When queue is full then a packet is dropped from the tail and the arrived packet is inserted in to the relative position in LAS algorithm. Due to this problem the flows from a source is denied a service until the other sources have been serviced. LAS use this principle to avoid the bandwidth hogging problem. figure 3 shows that source S2 utilizes an average bandwidth of 56% while the S1 average is 44% which means S2 utilizes bandwidth more than S1.One limitation of LAS is that it either drops or buffers those packets that received the most service; in doing so, LAS reduces throughput.

D. Throughput Analysis of RuN2C with different RTT

The simulation results of RuN2C in a heterogeneous network are presented in Figure 4. RuN2C follows the TQ approach in that packets from both S1 and S2, which are based on the output of Hybrid Classifier, are inserted in the appropriate queues. The packets from the real-time queue are completely scheduled in FIFO and then packet in the non-real-time queue gets scheduled. S1 and S2 get service in the first queue that avoids the bandwidth hogging problem, but only for real-time flows. From the simulation results, for non-real-time flows, S2 utilized nearly 60% of the bandwidth and S1 utilized 40%.

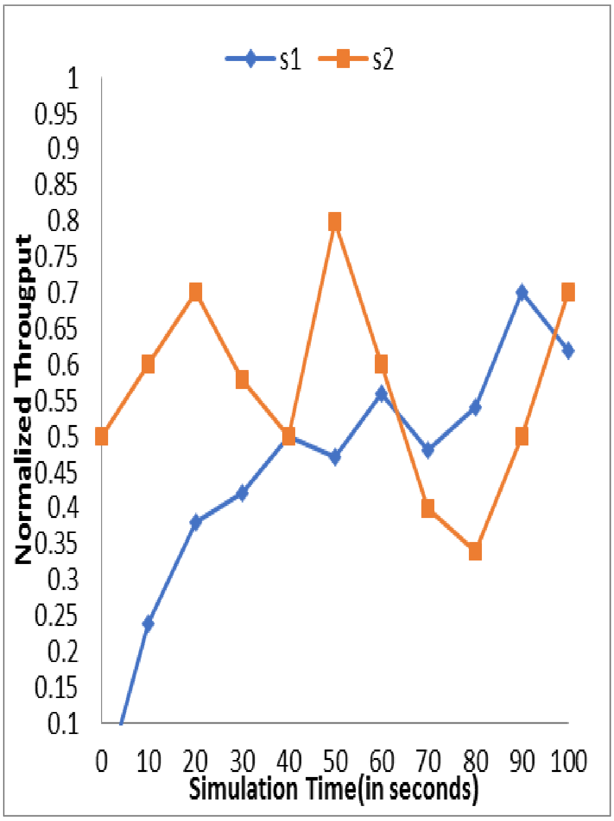

Fig. 4. Normalized Throughput versus Simulation Time of RuN2C Scheduler by Varying RTT in Heterogeneous Networks.

E. Throughput Analysis of Dynamic Scheduler with different RTT

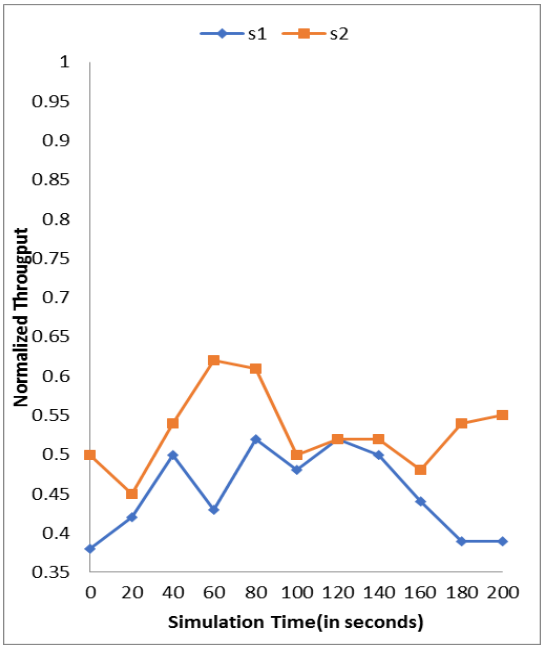

Fig. 5. Normalized Throughput versus Simulation Time of Dynamic Scheduler by Varying RTT in Heterogeneous Networks.

Figure 5 shows the Dynamic Scheduler simulation results in which RTT was varied. Simulation environments are assumed to be the same as in FIFO. The results show the benefits of Dynamic Scheduler over FIFO and other algorithms. S2 sends the packets to the router in a bursty manner than source S1. S1constantly delivers packets to the router. The Hybrid Classifier differentiates the flows into real-time and non-real-time flows and inserts S1and S2 packets into the appropriate queues. With *DPS(i)* and *TPmax*, the packets in both SFQ and LFQ are scheduled. Hence, packets from S1 and S2 receive service. S2 receives an equal share of throughput as source S1 that avoids the bandwidth hogging problem.

V. CONCLUSION

The normalized throughput of FIFO, LAS, RuN2C, and Dynamic Scheduler was analyzed by varying RTT. The proposed Dynamic Scheduler avoids the problem of throughput hogging which is bursty in nature and shares an equal amount of throughput irrespective of RTT. Varying propagation delay heterogeneous network suffers with a problem called bandwidth hogging. From the analysis it is clear that in heterogeneous network the algorithm FIFO allocates unequal bandwidth among several connections. But the Dynamic Scheduler avoids bandwidth hogging in heterogeneous networks regardless of RTT. Other protocols like LAS and RuN2C avoid the bandwidth hogging problem but the throughput decreases in LAS. The closed loop feedback mechanism increases the transfer time when there is a packet loss in TCP. Dynamic Scheduler decreases retransmissions of TCP flows which results in reduced packet loss and mean transmission time. The throughput is also almost constant than other scheduling algorithms.
Dynamic scheduling is the needed solution for queuing real time and non real time flows. Using FIFO degrades the performance of network with respect to packet loss rate and delay. Real time flows are given more priority when LAS is used. This leads to starvation in non-real time flows. RuN2C starves non-real times flows and considered only when real time flows are completely scheduled. Hence, RuN2C does not provide fairness. The proposed Dynamic Scheduler schedules both real time and non real time flows by the Dynamic Packet Scheduling Ratio DPS(i). This approach provides fairness and guaranteed service.

REFERENCES

1. Ashok, A, Fahad, RD, Dongsu, H, Boyan, L, Hyeontaek, L, Michel, Wenfei, W, AdityaAkella, David, GA, John, WB, Srinivasan, S, & Pseter, S 2011, 'XIA: An Architecture For An Evolvable And Trustworthy Internet' proceedings of Tenth ACM Workshop on Hot Topics in Networks, no.2, pp. 117-121.
2. Avranchenkov, K, Ayesta, U, Brown, P & Nyberg, E 2004, 'Differentiation Between Short and Long TCP flows: Predictability of the Response Time' proceedings of the IEEE INFOCOM, vol. 2, pp. 762-773.
3. K.Chokkanathan, S. Kotteswaran 2018, 'A study on flow based classification models using machine learning techniques' International Journal of Intelligent Systems Technologies and Applications, Vol 17,pp. 467-482.
4. Changhua Sun , Lei Shi , Chengchen Hu , Bin Liu, DRR-SFF: A Practical Scheduling Algorithm to Improve the Performance of Short Flows, Proceedings of the Third International Conference on Networking and Services, p.13, June 19-25, 2007
5. K. Chokkanathan and S. Koteeswaran, "An Integrated Approach For Network Traffic Analysis Using Unsupervised Clustering And Supervised Classification", Int. J. of Internet Technology and Secured Transactions-Vol. 9, No. 4, Sep-2019.

6. K. Chokkanathan and S. Koteeswaran, "A Study on Flow Based Classification Models Using Machine Learning Techniques", Int. J. of Intelligent Systems Technologies and Applications Vol. 17, No. 4, pp. 467-482, 2018 July.
7. K. Chokkanathan and S. Koteeswaran, "Privacy Protection and Perfect Classification Nature of C4.5 Algorithm", International Journal of Engineering & Technology, vol.7, 7 (2.24) 235-238, May-2018.
8. S. Dhivya, P. Shanmugaraja 2015, 'A study of Internet Traffic Classification using Machine learning algorithms', International Research Journal of Engineering and Technology, Vol 2, pp. 154-157.
9. Feldmann A, Gilbert AC, Willinger W, Kurtz TG. The changing nature of network traffic: scaling phenomena. ACM SIGCOMM Computer Communication Review 1998; 28(2): 5–29.
10. Gu, C, Zhuang, S, Sun, Y & Yan, J 2010, 'Multi-levels Traffic Classification Technique', proceedings of International Conference on Future Computer and Communication, vol. 1, pp. 448-452.
11. Jeong-Hwa, L, Jung-Min, P 2013, 'Dynamic scheduling for usable service in Network Robot', in Proceedings of Internet and Data Structure, pp. 252-259.
12. T. Kalaiselvi, P. Shanmugaraja 2016, 'Hybrid Algorithm for Classification of Traffic Flows' International Journal of Emerging Technology in Computer Science and Electronics.
13. Livadas C, Walsh R, Lapsley D, Strayer WT. Using machine learning techniques to identify botnet traffic. In Proceedings of the 31st IEEE Conference on Local Computer Networks, IEEE: Tampa, Florida, 2006; 967–974.
14. Nguyen TTT, Armitage G. A survey of techniques for Internet traffic classification using machine learning. Communications Surveys & Tutorials, IEEE 2008; 10(4): 56–76.
15. Shang ning, Tan Minggui, Wang Yaqing, Chui Zhongfa, Chui Yan, Yang Yong, "A Bp Neural Network Method for Short-term Traffic Flow", Computer Applications and Software, Vol. 23, No. 2, Feb. 2006, pp. 32-33.
16. M. N. Kasirian and R. M. Yusuff, ''An integration of a hybrid modified TOPSIS with a PGP model for the supplier selection with interdependent criteria,'' Int. J. Prod. Res., vol. 51, no. 4, pp. 1037–1054,2013.
17. Dr. Hussein A. Mohammed, Dr. Adnan Hussein Ali, and HawraaJassimMohammed."TheAffects of Different Queuing Algorithms within the Router on QoS VoIP ap& Communications (IJCNC) Vol.5, No.1, January 2013
18. Karagiannis, T, Broido, A, Faloutsos, M &Claffy, KC 2005, 'Transport Layer Identification of P2P Traffic', proceedings of IMC, pp. 121-134.
19. Kumar S, Turner J, Williams J. Advanced algorithms for fast and scalable deep packet inspection, In Proceedings of the 2006 ACM/IEEE Symposium on Architecture for Networking and Communications Systems, ACM: Hilton San Jose, San Jose, CA, USA, 2006; 81–92.
20. Kumar S, Dharmapurikar S, Yu F, Crowley P, Turner J. Algorithms to accelerate multiple regular expressions matching for deep packet inspection. ACM SIGCOMM Computer Communication Review 2006; 36(4): 339–350.
21. Kandula S, Sengupta S, Greenberg A, Patel P, Chaiken R. The nature of data center traffic: measurements & analysis. In Proceedings of the 9th ACM Sigcomm Conference on Internet Measurement Conference, ACM: Chicago, IL, USA, 2009; 202–208.
22. Viadutu, Alina, Dragos Comaneci and Ciprian Dobre, "Internet traffic classification based on flows' statistical properties with machine learning, International Journal of Network Management, vol-27, issue 3 2017.
23. Wang X, Zhang Q, Ren J, Xu S, Wang S, Yu S. Toward efficient parallel routing optimization for large-scale sdn networks using gpgpu. J Netw Comput Appl. 2018b.
24. Zander S, Nguyen T, Armitage G. Automated traffic classification and application identification using machine learning, In 30th Anniversary of the IEEE Conference on Local Computer Networks, IEEE: Sydney, Australia, 2005; 250–257.

AUTHORS PROFILE

Dr. P. Shanmugaraja, obtained his doctor's degree in Information and Communication Engineering from Anna University. He has authored and co-authored over 20 publications in the areas of Computer Networks, Machine Learning etc. He has guided more than 20 UG and PG scholars. His current research focusses on machine learning for flow classification and routing algorithms. He is a life member of ISTE. He has more than 19 years of experience in teaching field. He is currently working as Associate Professor, Department of Information Technology, Sona College of Technology, Salem, Tamilnadu.

Dr. K. Chokkanathan, Ph.D in Computer Science and Engineering Discipline from Veltech University, Chennai, India. He received his M.C.A degree from Madras University in the year 2000, M.E degree from Anna University in the year 2009. He has 19 years of teaching experience from various Engineering Institutions. Currently working for Madanapalle Institute of Technology, and Science, Madanapalle, A.P., as an Assistant Professor in MCA Department. He is having membership in various International Societies such as IACSIT, CSTA, ICSES and IAENG. He has published papers in 13 International journals. He attended and presented papers in 07 International conferences and 12 national conferences.

Dr J Anitha, is currently working as a Professor, Department of Computer Science and Engineering, DSATM, with an experience of 17 years. Graduated BE from KIT, Tiptur under Bangalore University in Computer Science and Engineering. She obtained her Master degree ME in Computer Science and Engineering from Anna University, Chennai and PhD degree from Anna University, Chennai in the area of Knowledge Based systems. She has 50 publications in International and National conferences, 20 publication in National journals and International Journals. Life Member of ISTE, CSI.

Dr. A.Parveen Begam, has co-authored over 20 publications in the areas of Distributed Systems, Mobile Computing, Grid Computing etc. She has guided more than 25 PG scholars. Her current research focuses on Security in Grid Computing, Performance Analysis in Mobile Grid. She has more than 18 years of experience in teaching field. She is currently working as Associate Professor, Department of Computer Science and Engineering, Taibah University, Yanbu, Saudi Arabia.

Dr. N.Naveen Kumar, working as Assistant Professor, in Madanapalle Institute of Technology and science, Madanapalle, Andhra Pradesh. He has done his Doctorate from Sri Venkateswara University, Tirupati, in the area of "An Efficient Approach to Remove High Density Salt and Pepper Noise Using Stationary Waverley Transform" in the year 2014". He completed his Master of Computer Applications from Sri Venkateswara university, Tirupati, in the year 2005. He has 10 years of experience in research and teaching. His areas of interest include Nural Networks, Image Processing and Data Analysis. He guided 50 PG Projects.