Improving scheduling performance in congested networks

Arif Husen [Corresp., 1, 2] , **Muhammad Hasanain Chaudary** [1] , **Farooq Ahmad** [1] , **Muhammad Imtiaz Alam** [3] , **Abid Sohail** [1] , **Muhammad Asif** [4]

[1] Department of Computer Science, COMSATS University Islamabad, Lahore Campus, Lahore, Punjab, Pakistan

[2] Department of Computer Science and Information Technology, Virtual University of Pakistan, Lahore, Punjab, Pakistan

[3] Department of Computer Engineering, University of Engineering and Technology Taxila, Taxila, Punjab, Pakistan

[4] Department of Computer Science, National Textile University, Faisalabad, Punjab, Pakistan

Corresponding Author: Arif Husen
Email address: arif.husen@vu.edu.pk

With continuously rising trends in applications of information and communication technologies in diverse sectors of life, the networks are challenged to meet the stringent performance requirements. Increasing the bandwidth is one of the most common solutions to ensure that suitable resources are available to meet performance objectives such as sustained high data rates, minimal delays, and restricted delay variations. Guaranteed throughput, minimal latency, and the lowest probability of loss of the packets can ensure the quality of services over the networks. However, the traffic volumes that networks need to handle are not fixed and it changes with time, origin, and other factors. The traffic distributions generally follow some peak intervals and most of the time traffic remains on moderate levels. The network capacity determined by peak interval demands often requires higher capacities in comparison to the capacities required during the moderate intervals. Such an approach increases the cost of the network infrastructure and results in underutilized networks in moderate intervals. Suitable methods that can increase the network utilization in peak and moderate intervals can help the operators to contain the cost of network intrastate. This article proposes a novel technique to improve the network utilization and quality of services over networks by exploiting the packet scheduling-based erlang distribution of different serving areas. The experimental results show that significant improvement can be achieved in congested networks during the peak intervals with the proposed approach both in terms of utilization and quality of service in comparison to the traditional approaches of packet scheduling in the networks. Extensive experiments have been conducted to study the effects of the erlang-based packet scheduling in terms of packet-loss, end-to-end latency, delay variance and network utilization.

Improving the Scheduling Performance in Congested Networks

5 Arif Husen[1,2], Muhammad Hasnain Chaudary[1] , Farooq Ahmed[1] , Muhammad Imtiaz Alam[3], Abid Sohail
6 Bhutta[1], Muhammad Asif [4]

8 [1] Department of Computer Science, COMSATS University Islamabad, Lahore, Pakistan

9 [2] Department of Computer Science and Information Technology, Virtual University of Pakistan, Pakistan

10 [3] Department of Computer Engineering, University of Engineering and Technology, Taxila, Pakistan

11 [4] Department of Computer Science, National Textile University, Faisalabad, Pakistan

14 Corresponding Author:

15 Arif Husen[1,2]

16 Email address: arif.husen@vu.edu.pk

Abstract

20 With continuously rising trends in applications of information and communication technologies in diverse
21 sectors of life, the networks are challenged to meet the stringent performance requirements. Increasing the
22 bandwidth is one of the most common solutions to ensure that suitable resources are available to meet
23 performance objectives such as sustained high data rates, minimal delays, and restricted delay variations.
24 Guaranteed throughput, minimal latency, and the lowest probability of loss of the packets can ensure the
25 quality of services over the networks. However, the traffic volumes that networks need to handle are not
26 fixed and it changes with time, origin, and other factors. The traffic distributions generally follow some
27 peak intervals and most of the time traffic remains on moderate levels. The network capacity determined
28 by peak interval demands often requires higher capacities in comparison to the capacities required during
29 the moderate intervals. Such an approach increases the cost of the network infrastructure and results in
30 underutilized networks in moderate intervals. Suitable methods that can increase the network utilization in
31 peak and moderate intervals can help the operators to contain the cost of network intrastate. This article
32 proposes a novel technique to improve the network utilization and quality of services over networks by
33 exploiting the packet scheduling-based erlang distribution of different serving areas. The experimental
34 results show that significant improvement can be achieved in congested networks during the peak intervals
35 with the proposed approach both in terms of utilization and quality of service in comparison to the
36 traditional approaches of packet scheduling in the networks. Extensive experiments have been conducted

37 to study the effects of the erlang-based packet scheduling in terms of packet-loss, end-to-end latency, delay
38 variance and network utilization.

Introduction

40 Packet Scheduling refers to allocation of sharing network resources among different types of flows with an
41 objective to maximize the utilization and fair or ensure a targeted share for each flow. It is a process of
42 selecting packets for outgoing transmission based on certain criteria. Incoming packets are added to queues
43 based on certain criteria depending on algorithm used and packets from these queues are selected for
44 transmission on outgoing transmission link. Different packet scheduling schemes exhibit different delay,
45 jitter, packet drop and throughput characteristics for users. Most of the time, the focus is to minimize the
46 degradation of quality of service and maximize link utilization.

47 The existing approaches use different criteria such as priority, fairness, class of service and packet size for
48 making the packet scheduling decisions. Although some of these parameters can be used to classify the
49 packets efficiently, none of the existing parameters can represent the traffic characteristics in terms of
50 capacity or volume. The capacity or volume of traffic characteristics across time and origin domains is
51 generally used in the network dimensioning and capacity planning, but don't have any explicit control over
52 controlling the scheduling of the packets in network nodes. The volume of traffic in a certain interval and
53 origin is generally described in terms of Erlang units. Erlang represents the volume of traffic (TV) generated
54 in a given busy period(BP). It was first described by Agner Krarup Erlang in 1908 (Angus, 2000). It has
55 been widely used for network dimensioning and capacity planning in cellular networks and Private
56 Automatic Branch Exchanges(PABX).

57 By calculating the Erlang traffic in an arbitrary period and origin, the traffic from different origins can be
58 assigned in a way that higher Erlang values get more processing time as compared to low values. The
59 allocation of the queue time in proportion to the traffic intensity provides efficient management of
60 congestion and better utilization of network resources. The scheduling algorithm proposed in this paper
61 utilizes above concepts to determine the period-based traffic intensity for each given origin of traffic. The
62 proposed algorithm schedules traffic more efficiently according to Erlang capacities of traffic from different
63 origins. The Erlang values when classified by origins are referred to Traffic Intensity (TI) of origin in a
64 given interval. TI corresponding to a traffic profile of an origin is dynamically calculated by a capacity
65 planning server. It can also be determined during the network development or planning phases using the
66 network dimensioning processes. The distribution of Erlang profiles to different network nodes can be
67 achieved in several ways such as a static configuration of network nodes and through dynamic
68 communications using a TI server. In up-stream nodes of traffic flow, TI is accumulated to minimize the
69 number of queues in core networks. The TI profiles can also categorize based on the type of service in order
70 to realize the commonly known service-based scheduling so that Erlang values impact the scheduling
71 decisions. The proposed algorithm can be used to implement integrated services or differentiated services
72 mechanisms for controlling the quality of service. The focus of this paper is to present a novel approach to
73 address the issues of network performance and utilization in congested conditions. Beside this, the
74 following are major contributions.

75 ● Provides a novel approach for scheduling decisions in networks with considerations of traffic
76   planning and network conditions

77 ● Formulate a process to plan the traffic intensity profiles

78   ●   Provide algorithm to make and influence packet scheduling decisions

79   ●   Present comparison results of performance and utilization

80   ●   Identify potential machine learning based solutions for intelligent scheduling decisions

Related Literature

82   The existing techniques of reciprocal processing of packets in queuing systems is based on priority, class,
83   or a group of services for wired networks with objectives to either increase the fair access or some controlled
84   access to the network resources. The wireless networks in this regard also focus on the conditions of the
85   channels and energy conservation. Both wired and wireless networks employ common algorithms such as
86   First-In-First-Out(FIFO), Round-Robin (RR), Deficit Round-Robin (DRR), Random-Early-Detection,
87   General fairness (GF), Stochastic Fairness(SF) or service priority(SP) based scheduling algorithms.
88   Sungjoo et al. (2016) have proposed a packet scheduling scheme that can employ multiple wireless
89   networks on the outgoing side. In their proposal they have modified the TCP-friendly Rate Control (TFRC)
90   algorithm, and they can distribute the packet transmissions to a group of users using multiple outgoing
91   interfaces that increases the QoS by reducing packet loss in the process of scheduling or due to network
92   congestion. Originally their proposal is in fact, an alternate approach to add the network resources or
93   bandwidth to serve the same users, the result is optimization of quality of service, but it reduces the network
94   utilization as more resources are added up. In contrast, our proposed work in this paper focuses on
95   increasing the quality of services with given network resources and also optimizing network utilization. In
96   fact, we can see that if we can enhance quality of service with given network resources, it will improve
97   network utilization.

98   For wireless networks energy efficiency is an important research area. Xu et al. (2016) proposed a packet
99   scheduling approach to maximize energy efficiency and it specifically focuses on wireless sensor networks
100   where energy constraints are critically important due to the shorter battery lives of the sensor nodes. They
101   have used an efficient offline packet scheduling approach along with a rolling-window-based online
102   algorithm for scheduling decisions. Our proposed approach can significantly enhance the energy efficiency
103   by unnecessarily comparative long periods of packets in queues and enhancing the quality of service. It can
104   address the energy efficiency problems if we can intelligently correlate the traffic profiles of sending nodes,
105   so that network nodes have necessary information to make scheduling decisions.

106   Binary search algorithm(BSA) is also an approach to enhance the packet scheduling. In this regard,
107   Pavithira & Prabakaran (2016) has proposed a packet scheduling approach of Long-Term Evolution(LTE)
108   based wireless networks using the novel binary search algorithm to increase the efficiency. It is noted that
109   any such approach will lead to the degradation of the quality of service due to the fact that it incurs delays
110   and jitter for the services in worst case and normal loads. In our proposed work we propose rather simple
111   operations of constant complexity both in time and space. This cannot only increase the network utilization
112   and hence the efficiency, but also optimize the quality of service by minimizing the delay, jitter, packet
113   loss, buffer overflows etc.

114   For video streaming applications, network coding selection is a possible way to improve the throughput
115   and minimize the delay. Since it is difficult to synchronize users or peers, the bandwidth utilization
116   degrades. Huang et al. (2016) have proposed an approach which coordinates between layer selection
117   algorithms with a distributed packet scheduling algorithm to enhance the quality of video streaming. In our
118   work, we shall show that both the improvement of quality of service and bandwidth efficiency is achievable

119   without adding such complexities to the networks. Fairness is an important aspect that scheduling
120   algorithms need to ensure among users, this is applicable to both with the best effort models and guaranteed
121   service models. Users should be able to have access to a fair amount of network resources as per their
122   subscriptions. Deb et al. (2016) has evaluated the relationship between the packet delay and fairness of
123   network resource allocation among traffic flows or users. In our research we have shown that our proposed
124   scheme can ensure the end-to-end delays with fair network resource allocation. In fact, in our proposed
125   scheme, traffic profiles ensure the minimum number of network resources that shall be guaranteed across
126   the network's flows or users or services.

127   Deshmukh & Vaze (2016) have proposed another approach for an energy efficient packet scheduling
128   algorithm. They have used deadlines where a fixed number of packets is assumed to arrive. It assumes that
129   all packets arrive within equal intervals. This assumption is basically difficult to arrive at in practice as the
130   packet size and intervals may not be fixed rather different users send different packet sizes. In comparison,
131   in our proposed approach rather than knowing the total number of packets within a fixed interval we use
132   traffic intensities and intervals that can vary without hard-specified limits. Fair queuing schemes focus on
133   to ensure reasonable fairness of allocation of network resources to the users or traffic flows, in practice the
134   traffic characteristics changes, but users may send traffic up to their subscribed rates. Patel & Dalal (2016)
135   have proposed a mechanism to adjust the weights assigned to traffic classes, that improves the performance
136   in contrast to the fixed weights of Weighted Fair Queuing(WFQ) or Class Based Weighted Fair
137   Queuing(CBWFQ). However, as the number of classes increases, the number of queues shall be increased
138   so it may lead to scalability problems. In our proposed work, TIPS aggregate the queue on the upstream
139   that minimizes the required number of queues. With a lesser number of queues, the performance of the
140   network is increased, and complexity is decreased.

141   Miao et al. (2015) has proposed a preemption-based packet-scheduling technique to reduce packet loss and
142   enhance the global fairness for software defined data(SDN). (He et al., 2016) evaluated the resource
143   allocation problems for cases requiring preemption and queues are not properly processed. Due to this
144   there is an increase in the length of queues and for preempted flows the delay is increased. In such conditions
145   either it is required to decrease throughput for the preempted queue or add up delay for packets.
146   Furthermore, the approach cannot handle cases where multiple packets fulfill preemption criteria. This
147   condition is frequent in the upper layer nodes of networks. Furthermore, this approach also does not
148   consider time and origin-based problems of user traffic.

149   A dynamic core allocation to enhance the packet scheduling in multicore network processors is proposed
150   by Iqbal et al. (2016). They have considered a packet scheduling scheme that incorporates various
151   dimensions of the locality to enhance throughput for network processors and minimize out of order packets.
152   The specific problem addressed in this is the out-of-order packet arrival due to different processors handling
153   traffic of the same flows. In our proposed work, the traffic originated from one source node is always
154   handled in the same queue and this queue can be attached to a specific network processor and hence it can
155   regulate the packet transmissions and minimize the out-of-order packet delivery.

156   Sharifian et al. (2016) have proposed a scheme to handle the real-time and non-real time traffic in the
157   common Radio Bearers (RB) for LTE based wireless networks. The improvements made by such an
158   approach are augmented over all capacity. Such an approach is useful in access nodes; however the upper
159   layer's networks have to rely on some other mechanism. In our proposed work we employ the role-based
160   handling of the traffic, or we can use in-conjunction-mode where the access nodes can use a packet
161   scheduling approach suitable to wireless medium or wireless technology to get relevant advantages.

162   Mishra & Venkitasubramaniam (2016) has presented a quantitative trade-off analysis between anonymity
163   and fair network resource allocation. Accordingly, in anonymous networking, encrypted packets from
164   various sources are re-ordered at routers randomly before processing in the outgoing direction. The authors
165   have shown that it affects the fairness of network resource allocation. They have shown the results of First
166   Come First Serve (FCFS) and fairness-based packet scheduling approaches using information-theoretic
167   metric for anonymity and a common temporal fairness index that provides a degree of out-of-order packets
168   transmitted. In our proposed work, the packets from the same sources are only allocated in the same queues
169   within the specified busy periods and the change as Traffic Intensity(TI) indexes change hence through this
170   way it ensures that packets are not always processed in the same queues, which indirectly provides
171   anonymity.

172   Yu et al. (2015) have proposed another approach for energy efficient packet scheduling through awareness
173   of the delay experienced by the packets. We can see so far there are no proposals so far for the traffic
174   intensity-based packet scheduling that employs the queue reordering not the packet reordering. Han et al.
175   (2015) have proposed stochastic scheduling for optimal estimation of the parameters. In this regard, it is
176   required to note that Stochastic Fair Queuing(SFQ) based packet scheduling approaches suffer from the
177   issues of scalability, whereas in our proposed work we are proposing a scalable packet schedule scheme.

178   Lee & Choi (2015) have proposed a group based multi-level packet scheduling for 5G wireless networks.
179   This approach considers the issue of user number and enhanced methods to reduce the interference between
180   beams. This approach is in fact, an effort to address the issues of access nodes operating in the wireless
181   domain. As with other approaches of wireless enhancements-based algorithms TIPS can allow the in-
182   conjunction mode operation where the wireless network related issues can be addressed and in upper layer
183   nodes TIPS can provide better quality of service and network utilization.

184   There was another approach proposed for multi-resource environments that is a fair and efficient packet
185   scheduling method called Active Time Fairness Queueing(ATFQ)(J. Zhang et al., 2015). The authors have
186   recommended the approach for middleware devices in data center environments. In fact, this addresses the
187   different processing time requirements on diverse resources. The issue of different processing time on
188   middleware or server can be addressed with our approach by using a traffic profile suitable for related
189   servers.

190   Kaur & Singh (2015) have proposed a Weighted Fair Queue(WFQ) based on Send Best Packet Next(SBPN)
191   algorithm to enhance quality of service with a specific focus on multimedia application in mobile ad-hoc
192   networks. With the WFQ based algorithms we have seen that in practice traffic intensities may vary, but
193   with given weights either class based or service-based algorithms, the network may remain utilized and
194   there may be higher traffic intensity traffic whose performance is subject to degradation. The TIPS
195   algorithm assigns period-based weights according to the traffic intensities of source nodes and handles these
196   weights by queue reordering and short consecutive resource allocation. We shall show in the results section
197   that in fact, such an approach dramatically increases the throughput, reduces jitter and end-to-end delay,
198   and optimizes the network utilization. This approach also reduces out-of-order packet arrival that is a typical
199   problem at the receiver side where the received packets shall be required to re-order and application shall
200   wait till all packets are arrived for a certain limit.

201   A class based weighted fair queue algorithm with generic traffic shaping mechanism is proposed by
202   Zakariyya & Rahman (2015). The class based weighted fair queuing is unable to respond to origin and

203   time-based problems i.e., time and origin of packets might be different but may carry the same classes, but
204   class based weighted fair queuing will handle them in a similar way.

205   Striegel & Manimaran (2002) have proposed a scheduling technique that relies on signaling protocol for
206   the resource reservation between end nodes; however, our proposed algorithm does not necessarily rely on
207   any signaling for resource reservation; rather it works on scheduling packets based on planned or forecasted
208   traffic intensities. In contrast to the signaling from end hosts for the resource reservation, the proposed
209   algorithm uses the concept of Traffic Intensity signaling which is efficient in scalability and efficiency.

210   Differentiated services approach was a result of efforts to overcome limitations of integrated services and
211   the basic idea in this was to aggregate traffic of similar characteristics based on bandwidth, delay guarantee
212   requirements. In comparison, the proposed in this paper, although follows a similar methodology, but it
213   limits the class or group-based decisions on access nodes and upper-level nodes relies on traffic intensity
214   according to time, the region, and seasons.

215   Static Earliest Time First (SETF) and Dynamic Earliest Time First (DETF) approach, proposed by Zhang
216   et al. (2001) is an aggregation-based packet-scheduling technique that works based on the First-In-First-
217   Out(FIFO) principle and uses timestamps to schedule packets. These methods also do not consider traffic
218   intensity, time and origin-based parameters in scheduling packets and nodes manage schedules in FIFO
219   manner. Erlang based traffic Intensity has been extensively used for capacity planning on voice and Private
220   Automatic Branch Exchange (PABX) systems (Angus, 2000) Public Switched Telephone Network (PSTN)
221   Switches and Gateways. The commonly used Erlang distributions are erlang B and erlang C. For erlang B,
222   it is assumed that traffic packets are discarded when free resources are not available whereas the traffic
223   packets are put in a waiting queue to be served upon the availability of the resources for Erlang C.

224   In this work, the erlang type B and type C are transformed into packet transmission with arbitrary busy
225   periods. Furthermore, the traffic intensities are classified based on their origin or serving area. Generally,
226   the serving area is represented by an access node known as a Point of Presence(POP). The algorithm
227   proposed in the article concentrates on a novel approach for packet-scheduling decisions aiming to improve
228   the quality of service, network utilization for Congested Network Conditions(CNC) serving the different
229   types of traffic generated by various sources with traffic intensity. This paper will show a comparison of
230   performance, simplicity, scalability for Moderate Network Conditions (MNC). Erlang is widely used for
231   network capacity planning and dimensioning of networks (Angus, 2000; Choi, 2008; Dahmouni et al., 2012;
232   Davies et al., 2004; Glabowski & Stasiak, 2012; L. Zhang et al., 2000) in telecommunication industry. This
233   work highlights the significance of Erlang TI for minimizing the delay, jitter, packet loss and maximizing
234   the data rates for CNC by using the TI dynamics in terms of time and origin-based characteristics for
235   optimizing the scheduling. The proposed algorithm offers dynamic interval-based priorities of queues and
236   proportionate scheduling decisions based on Erlang distribution. It allows making scheduling decisions
237   based on the role of the nodes, origin, and time-based features of the traffic on each node. The roles of
238   nodes are different and depend on network topology and architecture such as in a hierarchical network that
239   uses different layers such as access, code, edge, and aggregation. The role of nodes determines how traffic
240   is scheduled in upstream and downstream interfaces.

241   ## Materials & Methods

242   The existing Erlang Traffic Intensity definitions are based on the assumptions for the Time Division
243   Multiplexing(TDM) domain. In order to extend the concept for the statistically multiplexed traffic Internet

244 Protocol (IP) networks, some terminologies are required to be defined specifically. This section discusses
245 those definitions and description of various terminologies that are used to describe the working of the
246 proposed algorithm.

The Role of a Node

248 It is common to implement the IP networks in hierarchical layers where each layer provides dedicated
249 functionalities. The nodes in these layers generally have different technologies and protocols that are used
250 to perform certain functionalities. The hierarchical networks generally have the access network layer,
251 aggregation/distribution network layer, core network layer, transport network layer and edge network layer.
252 In the telecom industry there are other layers such as the service network layer. The service layer deals only
253 with the signaling traffic and hence has distinct characteristics and features. The service traffic also traverses
254 the transport nodes in order to provide connectivity with physical distributed service nodes. The scheduling
255 decisions are required to be made on each node; however, the objectives of scheduling vary per node. The
256 roles considered in this work include access, aggregation distribution node, core, transport, edge and service
257 node. The names of different layers discussed may vary from operator to operator, however, some other
258 names may have been used to represent the same functionality. The proposed algorithm uses the roles
259 defined above to decide if the Erlang TI needs to be accumulated or distributed.

Traffic

261 Since the IP packets may have different types of flows such as Transmission Control Protocol(TCP) or User
262 Datagram Protocol(UDP) flows and some packets may not belong to any flow definition, however those
263 packets play a role when scheduling them over the interfaces. So, for the purpose of this work, traffic is
264 referred to as the packet stream having some common attributes like a source, destination IP/port addresses.
265 The traffic definition considered here incorporates the effect of the packet size. In practice, there are
266 different units that can be used to describe it but, in this article, the traditional $xBPS$ routine is used. The x
267 refers to K for Kilo, M for Mega, G for Gigabits per second.

Traffic Volume (TV)

269 In contrast to the traffic definition described in BPS, Traffic Volume (TV) refers to the aggregated count
270 of bytes transmitted in an arbitrary interval and specific origin.

Busy Interval (BI)

272 Erlang TI is defined in terms of busy hours(BH). In this work in order to address the granularity issues the
273 Busy Interval(BI) is any arbitrary interval over which the traffic intensity has been specified. The BI could
274 be measured in seasons, week, day, hours are seconds and sub seconds. The values of TI depend on BI and
275 TV and consequently the packet size in given traffic.

Traffic Intensity

277 With the extended definitions in the above sections, the traffic intensity represents the ratio between the
278 expected volume of traffic and the actual capacity of the network or link in a given arbitrary busy interval
279 and place in the network. Further details of the traffic intensities and methods to calculate them shall be
280 covered in the following sections. For the end user point of view the traffic intensity is defined in the traffic
281 volume generated by the user vs. the user subscription data rate in an arbitrary busy interval. In this article
282 erlang C is used to calculate traffic intensities of end devices. On a given time, there may be several services

283   running such as IPTV, voice over IP and the high-speed internet. Considering σ the maximum data rate for
284   normal operation of a service and active time be h, traffic intensity t per service type is defined as below.

285    $$t = \sigma h$$

286   Furthermore with q be the number of queues corresponding to a traffic type in given POP, d the probability
287   of delay for an arbitrary queue and packet generation according to Poisson process and delay as exponential
288   distribution. The probability of delay of   a packet is given by the following relation.

289    $$d = \frac{t^q\ q}{q!\, q - t} \cdot \frac{1}{\left(\sum_{i=0}^{q-1} \frac{t^i}{i!} \right) + \frac{t^q\ q}{q!\, q - t}}$$

Point of Presence (POP)

291   The term Point of Presence (POP) represents, in fact, the area of an access node of the network where
292   subscribers are connected directly or indirectly. POPs can be classified based on origin, network function
293   or layer such as Residential POP and Commercial POP represents the service node that serves the users in
294   a residential area and commercial areas respectively. Similarly, access POP, aggregation POP and core POP
295   represent the functionalities provided by nodes in the access layer, aggregation layer and core layer
296   respectively. Other categories follow a similar method.

Traffic Profile (TP)

298   Traffic profile is a representation of the traffic intensity of a user or node in the form of a histogram that
299   has the BI, Packet Size, and other related parameters. These profiles are used by the proposed algorithm to
300   determine the information of the arbitrary intervals and intensities and packet sizes. The TPs are aggregated
301   on the upstream nodes while distributed on the downstream nodes. There are different methods to determine
302   the TP that includes the Forecast Based (FOB), Historical Network Usage Based (HNUB), Application
303   Signaling Based (ASB) and Operator Policy Based (OPB). The details of these methods shall be covered
304   in the next sections. However, for the purpose of evaluation and comparison in this work, the FOB method
305   has been used to determine the traffic profiles. FOB provides the upper bound to traffic estimates (represents
306   what networks are designed to support) as compared to the ASB or HNUB that represents a more real state
307   of the network.

Time and Origin Characteristics (TAOC)

309   Different users in different types of serving areas have different network usage patterns. Some POPs could
310   be seasonal, others could be generating traffic in different hours of the day or week based on geographical
311   statistics. Furthermore, there would be different characteristics of the traffic generated by the different POPs
312   in terms of the packet size, latency requirements and data rates. All these characteristics are referred to as
313   TAOC characteristics and mainly used in traffic profiling processes.

Scheduling Constants

315   Scheduling Constant SK and K depends on the traffic profile information of POPs. These constants are
316   used to control the scheduling decisions and may also represent policies. The SK constant is linear additive
317   to administratively manipulate scheduling behavior and is configurable per node whereas the constant K is
318   defined globally on a network or layer domain.

319 **Congested Network Conditions (CNC) and Moderate Network Conditions (MNC)**

320 The Congested Network Conditions(CNC) represent a network condition where maximum users generate
321 the traffic over a network. It is the situation where POP has high activity. Generally, not all POPs always
322 operate in CNC state. There are only some periods that may have this condition. This fact allows the
323 network provider to define their oversubscription policies based on the activity of a POP. On other hand,
324 Moderate Network Conditions (MNC) refers to a POP condition that has moderate or less than maximum
325 activity and users in those POPs may generate traffic that is intermittent or sparsely distributed over the

---

Algorithm I

```
Initialize ()                    RECEIVE_SIDE(Packet P) {          TRANSMIT_SIDE() {
{                                VAR ln ← LastNode();              VAR p  ← Packet *P;
Bind_local_variables ();         VAR bp ← get_BP();               VAR p  ← Determine Profile;
Load_profiles ();                VAR p ← get_Profile();           VAR bp ← Determine BP;
Initialize_queues ();            VAR ps ← get_PS();               VAR nr ← Determine NodeRole;
VAR Special queue.               VAR nr ← get_NodeRole();         VAR tg ← TrafficGroups;
VAR GLOBAL TI;                   VAR tg ← get_TrafficGroups()     VAR ql ← GET_SORTED_QUEUE-LIST (DESCENDING TI);
}                                VAR qto ← ENQ_TO;                VAR q  ← SELECT Serving_Queue();
                                 VAR sCriteria ← Evaluate SCriteria();   VAR i  ← GET_SCHEDUALAR_Instance;
                                 IF (sCriteria == True) {
                                 Qs(P);                           IF (SIZE(Qx) < TI) {
                                 }                                DDQ(Qx);
                                 ELSE {                           }
                                 IF ( SIZE(qto) > TI ) {          ELSE {
                                 Drop Packet();                   Qx->DEQUEUE();
                                 } ELSE {                         }
                                     qto ->ENQUEUE(P);            SET_NEXT_QUEUE;
                                     }                            RETURN *p;
                                 }                                 }
```

---

326 timescale. In this state, free resources available on the network exceed the traffic load. The networks remain
327 under-utilized because operators generally plan the network according to the CNC requirements. The
328 proposed algorithm considers these considerations and helps to control extravagant investments on the
329 network infrastructure to reduce the cost of network services.

330 **Comparison Parameters**

331 The evaluation and comparison of the effects of using the proposed algorithm will be based on the certain
332 parameters that includes QoS measures such as sustained throughput delay, latency and jitter and network
333 utilization. These parameters shall be measured with the proposed algorithm and with other existing
334 algorithms over a single network topology that consist of the hybrid simulated and emulated environments.
335 The details of topology and environment shall be discussed in the subsequent sections.

336 # The Proposed Algorithm

337 This section covers the details and operation of the proposed algorithm called Traffic Intensity based Packet
338 Scheduling (TIPS). It uses the TI profiles for scheduling decisions with consideration of various factors
339 such as the node role, scheduling factors and TAOC. There are few parameters that need to be defined for
340 description of the algorithm. These includes the $Q, sQ, BI, TI, SBI, SQ, enQ, deQ, QL, SC$ and xQ that
341 represents a packet scheduling queue, special queue for control purpose, buys interval, the traffic intensity,

342 selection procedure for busy interval, queue selection procedure, procedure to add a packet to queue,
343 removal procedure from a queue, queue length, scheduling criteria and the queue selected by the SQ
344 procedure respectively. The TIPS algorithm exploits a set of traffic profiles on nodes. On the down-stream

---

345   it distributes the profiles into multiple profiles based on the traffic information and on upstream nodes it
346   aggregates the profiles received from the other nodes. Through this way the number of queues is reduced
347   consciously as traffic traverses the upper layers of the network hierarchy and increases the number of queues
348   on the downstream nodes or nodes facing the users. The lesser number of queues in upper layer nodes
349   reduces the processing delays and time spent whereas on the downstream nodes it allows the queues as
350   required by the user types. The traffic profiles on the network nodes can be configured in several ways as
351   enlisted below.

352   1.   Add traffic profile database to each network node statically by administrator

353   2.   Deliver the selected set of traffic profiles from a profiling server

354   3.   Allow upstream nodes to learn or receive profiles from downstream nodes

355   **Figure 1** shows the initial setup phase corresponding to the Initialize () procedure TIPS algorithm pseudo
356   code (**Algorithm I**). In this phase, a number of queues are created according to the traffic profiles received
357   or configured on the network node. The queue limit is determined based on the traffic intensity values in
358   the given busy interval and these limits change with changes in the BI. Different queues are created with

359   a size determined by the TI values for each profile on each node running the TIPS algorithm. On the upper
360   layer nodes of the network hierarchy, profiles are either received from the downstream nodes or fetched
361   from a local node database or any other method as discussed in the earlier section. This step is important as
362   it defines the number queues and their size. In addition to this, the user traffic queues, a special queue is
363   created to handle the packets generated for the control protocols that carry various information such as
364   routing updates, link or channel setup, resource reservation messages. In the upstream direction the profiles
365   are aggregated and on downstream nodes, the profiles are distributed according to the number of profiles.
366   The special queue is processed immediately if its size is greater than zero. Furthermore, any other queue
367   that is being processed is preempted and the enQ and deQ operations are performed on the special queue.
368   The special queue can be considered as the highest priority queue with preemption. All the elements that
369   are contained in the TIPS setup block are used for the initialization of the TIPS algorithm. During the
370   initialization of TIPS, nodes have sQ, current BI and different queues according to the traffic profiles.

371   In each BI, the enQ and deQ processes functions according to the information in the profiles available. The
372   size of the queue is chosen according to the volume of the control traffic existing in the network and it is
373   ensured that control packets do not encounter any packet loss. Receive side processing of the TIPS
374   algorithm is shown in **Figure 2**. On the reception of a user packet the TIPS check if there are any control
375   packets, if the sQ is empty then the received packet is processed. For the user packets received on the
376   interface, packets are processed by enQ according to the QL limits. If the QL exceeds the packets are
377   immediately dropped without any further considerations processing, however, such a situation rarely
378   occurs as while the packets are being enqueued during the BI, the deQ processes are also working to empty
379   the queue according to TI values. This process is repeated on each incoming interface of the node. The
380   processing of the TIPS algorithm on the outgoing side is shown in **Figure 3** where sQ is de-queued with
381   high priority. If the size of sQ is zero, the queues having higher sizes are processed first according to the
382   number specified by the algorithm. The current size of any queue is determined by the TI values, so higher
383   the TI values for a given queue, a greater number of packets shall be processed repeatedly to equilibrate the
384   queue system. During the deQ processes the SK and K parameters play an important role that defines the
385   limit of the packets that can be processed by deQ regardless of the TI values. Generally, these factors are
386   the upper limit to the TI values hence eliminating the chances of the non-processing of other queues.

387   If a Q is deQ by a limit specified by the TI values or the upper limit of SK, the next Q is selected by the
388   SQL for deQ processing, and this process is repeated until all queue size reaches zero.

389   During packet departure, SK_x (where $SK_x = SK\ (ij),\ i = Q\ number\ and\ j = BI$) confirms the traffic
390   with higher TIs does not interfere with the latency of queues having the smaller values of TI. It also makes
391   sure that the traffic corresponding to the high values of TI is allocated sufficient resources.

Methods of Traffic Profiling

393   In the earlier sections, different traffic profiling techniques have been mentioned. This section focuses on
394   specific details of each method and discusses how these can be used for the proposed algorithm.

A. Forecast or Business Planning Based (FOB)

396   This is the simplest method of building traffic profiles. This is similar to the procedures as used in the
397   capacity planning of the network where the objective is to identify the services, the number of users, future
398   margins, and over-subscriptions, Service Level Agreement (SLA) for each of POP in the given network.
399   Generally, these are the forecasts part of the business plans of an organization. The typical classifying is
400   based on the marketing product being offered and the relevant SLAs. The parameters that required
401   converting a forecast to profile are as followed,

402   - Number of subscribers per service per POP

403   - Bandwidth per service

404   - Service classification

405   Based on the demographic survey, it provides the following four values for each service for each POP.

406   - Hour wise Usage (HWU)

407   - Day wise Usage (DWU)

408   - Week wise Usage (WWU)

409   - Month wise Usage (MWU)

410   Apparently, this looks to be a very tough task of suggesting the usage for each hour of the day of the week
411   of the year. For HWU, the important thing that we need to know is to type POP. If POP is residential and
412   the community is a city, and man and woman are employed, then we simply need to know the official
413   working hours. In working hours residential usage will be low, whereas the commercial POP will have
414   higher usage. For DWU , the major task here is identifying the working days and weekend days, holidays.
415   In case of WWU, we differentiate the weeks based on any specialty attached to it, if nothing important is
416   scheduled then simply choose it as 100%. For MWU , the month level specialty differences required, if
417   nothing different, then simply fill it with 100%.

B. Historical Network Usage Based (HNB)

419   This technique is based on monitoring a given network for its network utilization. Monitoring is required
420   at the access nodes, and the upper layer nodes shall use the fundamental concepts of profile aggregation.
421   The traffic monitoring can be achieved with open-source tools like CACTI (Ian Berry et al., 2012) and
422   MRTG (MRTG, 2012). The things that need to be monitored included the following.

423   ● Required information is access interface utilization, packet size accords the time scale.

424   ● The minimum, maximum and average packet size.

425   ● Identify the traffic classes or groups if required.

426   ● Obtain the policies for traffic groups

427   The process of building the traffic profiles five pages as shown in **Figure 4**. It consists of   five steps as
428   discussed below.

Statistics collection Phase (SCP)

430   This phase requires the collection of hour wise interface utilization, subscriber information, services
431   information, network topology & service prioritization and bandwidth guarantees.

Statistics Analysis Phase

433   This phase involves the categorization of access POPs, defining the usage patterns with respect to time
434   (hour, day, week, month, season, day, and night), defining the usage patterns with respect to POP type and
435   analyzing the packet sizes.

Traffic Profiling phase (TPP)

437   This phase calculates the traffic volumes per hour per day per week per month/day/night/season/special
438   event, determine the average packet size, calculate the Traffic intensities, define the distinct busy periods,
439   and define the average packet sizes per BI.

Profile Optimization Phase (TPOP)

441   In this phase the traffic policies like priority and bandwidth guarantees per service, time specific policies to
442   traffic profiles, origin specific policies to traffic profiles, administrative policies to traffic profiles are
443   applied to the output of the previous phase.

Final phase (FP)

445   Using above information this phase is used to generate traffic profile files and apply to network nodes as
446   part of the TIPS.

Application Signaling Based (ASB)

448   In this case, we use an application based signaling protocol to inform the current status of the client machine
449   by analyzing the following.

450   ● Current running applications

451   ● Over-the-Air (OTA) settings

452   ● Operating System (OS) update settings

453   ● System up down behavior

454   We look at the networks enabled, running applications and their capabilities such as if it can initiate a voice
455   session, file transfer, browsing, video session and file download. It will monitor when a session started, file
456   download started, conferencing started, email retrieval started, retrieval frequency. In this scheme an agent
457   is installed on each host that communicates with access nodes updating its current expected usage. An agent

458 monitors the user's behavior, OTA settings, browsing behavior. We used a scenario where the user hosts
459 communicate with the Customer Premises Equipment (CPE) which is generally a xDSL Modem or Wi-Fi
460 Router. Router maintains a list of running applications and host capabilities which is updated by the agent.
461 Generally, in residential cases the user's hosts may have mobile, notebooks and desktop, gaming, or IPTV.
462 The simple case may be just to enumerate the number of devices and their capabilities and apply common
463 basic characteristics to each of them. We take this case in this study where the CPE just reports the user
464 devices and their capabilities. The ASP traffic profiling scheme is shown in **Figure 5** and its integration
465 with the nodes running TIPS is shown in **Figure 6**. CPE may collect the traffic details such as applications
466 and their expectation and network usage pattern and   communicate with the TIPS instance running the
467 access node and provide information such as total interfaces, traffic volume, traffic intensity, service classes
468 and a group per interface. This information is compared with existing traffic profiles associated with the
469 CPE and if the changes are found the profiles are updated and communicated to other network nodes also.
470 The subsequent scheduling decisions shall follow the updated profiles to reflect changes in the traffic
471 characteristics.

Improving the Scheduling Performance

473 The TIPS algorithm operates on the fact that POPs generally exhibit different characteristics in terms of
474 usage and users have different applications that generate different types of traffic. In addition to the above,
475 different POPs have different characteristics in terms of the time when traffic is generated by users and the
476 type of region. Furthermore, the network operators are interested in having control over the behavior of
477 unexpected traffic originating from users. The existing commonly used packet scheduling techniques such
478 as Deficit Round Robin (DRR), Round Robin(RR), Priority Queueing(PRQ), Fair Queue(FQ) and Class
479 Based Fair Queuing (CBFQ) do not exploit these considerations in their operations. Consequently, the
480 network providers opt to use additional techniques to control the scheduling in network nodes. Similar is
481 the case with scheduling schemes for wireless networks that considers channel conditions and other factors.
482 Due to the absence of above features in packet scheduling techniques following consequences are observed.

483 ● The techniques are not aware of capacity and network dimensioning constraints

484 ● The commonly used algorithm cannot cope with the diversity in traffic characteristics

485 ● Their objectives are relative remain static

486 ● Existing techniques cannot cope with the dynamic nature of the traffic.

487 The absence of these features degrades the resource allocation fairness and quality of service parameters
488 such as End-to-End Delay (E2ED) and its variance incurred by unpredicted traffic management in queues
489 with static criteria, for example, the priority queuing is unable to distinguish between traffic generated by
490 different POPs having common priority. Furthermore, it also cannot distinguish traffic with the same
491 priority on different timescales. Similar is the case with round-robin or deficit round-robin and weighted
492 fair queuing or class based weighted fair queuing as all traffic classification, their types and mechanisms of
493 fairness are fixed over time and for each POP. In order to address these limitations, the proposed TIPS
494 algorithm is devised to consider all these aspects and offer improved scheduling for a different quality of
495 service parameter such as packet loss, delay, jitter, and sustained throughput.

A. Management of Unpredicted Delays

497   In BI_x, the TIPS algorithm chooses the highest TIs available in traffic profile, and enQ corresponding
498   packets in Q_x. After this, it selects next highest TI and enQ its packets in $Q_{\{x+1\}}$. This process repeats till
499   all traffic profiles are processed and all packets are enQ to corresponding queues. The arrived packets are
500   dropped if the corresponding size of the queue exceeds TI value and the SK factor. At any given time,
501   packets in all queues determine the total size of the queues (S_Q) and is equal to the sum of all TIs from all
502   POPs.

$$S_{q,bp} = \sum_{i=1}^{n} TI_i \qquad (1)$$

503   If $S_{q,bp}$ surpasses QL then packets belonging to the corresponding profile are dropped where TI is exceeded.
504   For the deQ case, the maximum limit of the count of packets that are allowed to remain in a queue is the
505   product of the average packet time and sum of all TIs. In the experiments, it is seen that the maximum time
506   a packet occupies in a queue is five to seven times of their packet-size. The minimum time occupied in any
507   queue is most of the time equal to one packet time.

B. Regularizing the Traffic Bursts

509   The spikes in network traffic generally increase in delay while making decisions of packet scheduling. In
510   cases the decisions are priority based, or class or round-robin based then such a situation could lead to
511   packet loss. In the proposed algorithm the spikes in traffic are regulated with allocation of the length of the
512   queue in proportions to size of TI. The Traffic having the same class or priority or group with different
513   origin or time get distributed allocations.

C. Handling the Large Packets in Traffic

515   Larger size packets lead to an increase in E2ED due to their large transmission time and occupancy time.
516   The proposed algorithm inherently solves this issue with a cautious profiling of TI in each BI. TI values are
517   based on traffic volume that considers the aspects of the size of the packet's users, sessions, or duration.
518   Due to this the TIPS algorithm is capable of effectively controlling the additional delays due to large size
519   of packets and jitter normalization.

D. Variations in Delay

521   Generally, end-to-end delay variations occur in the traffic due to different network conditions and non-
522   linear distribution of decision of scheduling. The proposed algorithm TIPS handles this problem by
523   successively dequeuing packets according to TI values. Since the TI values are directly dependent on the ,
524   hence the proportionate processing time is allocated to different types of the packet leading it to decrease
525   delay variations. Furthermore, the proper planning of traffic profiles helps to eliminate such variations.

E. Minimizing the Packet Loss

527   Packet loss in the network is due to various reasons. One of the reasons could be inefficient scheduling
528   decisions where the queues are filled rapidly, and limits are reached. In such cases the network nodes have
529   no option but to drop further packets. The dropping of few packets part of connection-oriented flows leads
530   to impact on the congestion control mechanism adopted by the upper layers of the TCP stack. Furthermore,
531   the large spikes of packets with lesser priority for PQ and lesser weight traffic for CBFQ don't get enough
532   resources and queues quickly overflow that lead to the loss of packets. Due to the burst regularization in

533 the TIPS algorithms, larger packets get resources according to their calculated TI values. This leads to
534 reducing the probability queue overflow that subsequently minimizes the packet loss due to scheduling
535 issues.

F. Role Based Scheduling

537 In hierarchical topology of networks, nodes are arranged in layered fashion, such as access, distribution, or
538 aggregation and code nodes, where each layer has specific network functionalities. The proposed algorithm
539 provides the node role-based scheduling mechanism. The scheduling decisions in access nodes differ from
540 other types of nodes. On access nodes, TIPS can be used to provide the class-based or group-based
541 scheduling. These classes are groups that can be handled differently according to the needs of the class of
542 groups. These groups or classes must be matched to the traffic profiles in order to benefit from this feature.
543 On the nodes that implement the aggregation role, the TIPS algorithm makes decisions by classifying the
544 traffic based on TI values received from other nodes. This reduces the queues count and simplifies the
545 processing of queuing operations. With core roles it aggregated and redefined TI values. These aggregated
546 values are then used in the making decisions for the scheduling of packets. This directly results in lowering
547 the queues further and minimizing the processing delays.

G. Shorter Queue Sizes

549 The proposed algorithm regularizes the traffic bursts by manipulating TI values. The short burst of the
550 packets causes momentary rise in TI values and consequently more processing time allocated in such cases.
551 This leads to minimal queue growth and maintains the average queue size within the capacity of interface
552 buffers. Generally, shorter queue sizes or the lower rate of queue growth improves the throughput,
553 minimizes the delay, and jitter.

H. Joint Mode Support

555 The proposed algorithms support the joint operation with other scheduling techniques. The joint operation
556 is based on the different nodes i.e., some nodes in the network may be using the traditional scheduling
557 algorithms and other nodes use the TIPS Scheduling. With this support the objectives like prioritized
558 handling or class-based handling of traffic can be retained.

Experimental Setup for Evaluation

560 For evaluation of the proposed scheduling technique, A setup consisting of network simulation using
561 Network Simulator (NS2, 2012) and containers (Dockers, 2012) was used. The end points were created in
562 the Linux environment where each user and server were created standard docker containers. The traffic
563 generated by the container hosts was connected to the access nodes in the NS2 simulated environment. The
564 host containers generate traffic according to a selected scenario type of POP. The traffic generated by the
565 hosts follow a random pattern following the BIs and given traffic intensities. The real- time traffic passes
566 through core, aggregation and access nodes with topology as shown in **Figure 7** and the respective
567 bandwidth parameters are given **Table I**. The network topology was created in a simulated environment.
568 The simulated network nodes run the TIPS instances with statically configured roles as per topology of the
569 network that follows the three-layer hierarchy commonly used in practical networks. In the analysis of the
570 results of the TIPS algorithm, the color scheme as shown in the figure will be used and traffic traversing
571 over the links will be defined with node colors. For example, the traffic generated on node A_1 will follow

572   black color, A_2 will be blue and similarly for the remaining nodes. For IEEE 802.3 based scenarios as shown

573   in topology, following interface capacities are defined both for CNC and MNC models.

574   **Application for the Traffic Generation**

575   A custom-built model application (MAP) is used to produce traffic as per BIs, TIs and s for different nodes.

576   MAP is intended to operate in twenty-four BIs and s. Each BI has a distinct TI allocated for each node that

577   is calculated using the FOB approach discussed in the earlier sections. The theoretical volumes of traffic in

578   packets shown in **Table II (a)** for each of the notes for the experimental setup. The packets are set to 1k

579   Bytes, which may vary in practical applications. In such cases the total bytes transferred are divided by the

580   average for a common base for comparison. Above table shows traffic volumes for five periods; however,

581   the simulation is run for several periods. **Table II (b)** represents the start busy interval that ends with the

582   beginning of the next busy interval. Although the tables show the same values for five busy intervals having

583   different traffic volumes, the experimental setup is run for several busy intervals. The equivalent TI derived

584   for these intervals is shown in **Table III**. The process of derivation of TI values is discussed in previous

585   sections. The main objective in this experimental setup is the delivery of the volume of traffic efficiently

586   and reliably without loss of packets and optimized quality of service and performance parameters such as

587   delay, jitter, and highest possible throughput. **Table III** shows Traffic intensities corresponding to each BI.

588   It shows TIs for the first five BIs and for the remaining BI, the TI are calculated in a similar way. The TI

589   table follows the relationship as per Equation (2) where TI_{ij} is the intensity of traffic for node j in busy

590   interval i, TV_{ij} is the volume of traffic in node j and the busy interval I and SK values are constant. TV_T is

591   the traffic volume of outgoing links on nodes.

$$TI_{ij} = \{1 + SK\,(TV_{ij})/(TV_T) + SK \qquad 0 > \left(\frac{TV_{ij}}{TV_T}\right) < 1 \left(\frac{TV_{ij}}{TV_T}\right) \geq (\,2\,)$$

592   The scheduling fact SK values chosen are 2,1,2,1,2 and 2 respectively for N1 to N6. The values of this

593   parameter as derived using the Equation (3) where TI_{ij} refers to the traffic intensity allocated to i_{th} queue

594   in j_{th} BI, and SK_{ij} represents the SK values for i_{th} queue in j_{th} BI. The i_{th} queue is assigned dynamically to

595   i_{th} profile in the corresponding busy interval.

$$SK_{ij} = \{0\ TI_{ij}/TI_{ij} \qquad TI_{ij}/TI_{ij} < 1\ otherwise \qquad (\,3\,)$$

596   In the experimental setup the K constant is set to 1 for all the nodes of the network.

597 # Results

598   As discussed earlier, the experimental setup uses the docker container for creating the communication end

599   points and NS2 for the network functionalities. The MAP application is used to generate traffic

600   corresponding to the traffic profiles. The type of traffic was set to real-time RTP.

601   **A.   Performance in Congested Network Conditions (CNC)**

602   This section discusses the measurements carried out for TIPS performance in congested network

603   conditions. The network setup was created with the link allocation as given in the earlier sections. On CNC,

604   TIPS algorithm performance is significantly better as compared to traditional scheduling techniques.

Measurement Results of Delay

605 **Measurement Results of Delay**

606 End-to-end Delay is an important parameter in the measure of quality of service of the applications. The
607 end-to-end delay in the network could be due to various reasons where queuing delay is one factor. The
608 TIPS algorithm is intended to minimize the delay incurred due to the waiting time of packets in queues.
609 Waiting time in the queue for packets can be lowered by considering the rate of packet origination from
610 source nodes and adopting the scheduling behavior accordingly. Large values of traffic intensity require a
611 faster processing of corresponding packets in comparison to smaller values of TI. The applications running
612 on the host machines may have similar end-to-end requirements but different TI values in different BIs may
613 increase the delay in the scheduling process for lesser and higher TIs. The TIPS algorithm manages the
614 queue allocation time by selecting suitable profiles in the queues and manipulating the order of enQ and
615 deQ functions. **Figure 8** shows the measurement results of end-to-end delay exhibited by the TIPS
616 algorithm in the CNC case. It depicts the latency experienced by packets in the network in milliseconds(
617 ms) including the delay due to the propagation in the medium. Some nodes like Black, Red, Green, and
618 blue are at a distance of three hops and other nodes with brown and yellow color are at a distance of two
619 hops. In the experimental setup the propagation delay is set to the value of 10ms for all links.

620 **Results of Throughput Measurement**

621 The **Figure 9** shows the measurement results of sustained throughput attained with the TIPS algorithm. It
622 represents the packets- per-second(PPS) on y-axis and timescale on x-axis. The significance of these results
623 is the smooth curve of throughput in every busy interval. The traffic was generated using a Model Source
624 Application (MAP) that generates the Real-Time Protocol(RTP) traffic following the deterministic TI
625 distribution. The comparison of the FOB traffic volumes and the actual traffic volumes successfully
626 delivered is shown in **Figure 10**. FOB traffic volumes are calculated from hypothetical forecasts of services
627 and their expected usage pattern. It represents the sum of packets on the y-axis and the respective source
628 node on the x-axis. The overhead of the MAP application is about 3.6 percent that includes the packets
629 sent/received for the keep-alive and acknowledgements during the lifetime of the session. The figure shows
630 that the TV received is exactly 3.6 percent lower than the FOB traffic volumes and the TV Delivery ratio
631 is 96.43 percent as shown in the figure.

632 **Delay Variance or Jitter Comparison**

633 The Delay Variance(DV) is also a significant factor that can degrade the QoS of service for applications
634 running on communication endpoints. It is a significant parameter for the jitter buffer and synchronization
635 of multimedia endpoints. As discussed in ITU-T-G.8261.1/Y.1361.1, (2012); RFC-3339, (2012), the jitter
636 is defined as the variance in end-to-end delay experienced by a given node between two successive packets.
637 For the experimental evaluation, the Equation (4) is used to derive the jitter in the received traffic where D_i
638 and D_j is delayed between two successive packets i and j.

$$J_{ij} = Abs(D_j - D_i) \qquad (4)$$

639 For DV or jitter, the average, minimum and maximum values are derived using the Equations(5) to (8).
640 For these equations the reordering of the packets has been ignored.

$$J = \frac{abs(D_j - D_i)}{j - 1} \quad i \neq j \qquad (5)$$

$$J_{avg} = \sum_{n=1}^{pc} \frac{abs(D_j - D_i)}{(j-1) * pc} \quad i \neq j \qquad (6)$$

$$\lfloor J \rfloor = \frac{abs(D_j - D_i)}{(j-1)} \quad i \neq j \qquad (7)$$

$$\lceil J \rceil = \frac{abs(D_j - D_i)}{(j-1)} \quad i \neq j \qquad (8)$$

641   In the above set of equations, pc refers to the count of packets, Abs refers to the absolute value delay
642   experienced and i and j are the sequence number of packets. **Figure 11** shows the jitter or delay variance
643   in the traffic received by different nodes. The values of jitter of each packet in the experiment duration were
644   recorded and the values shown in the figure are instantaneous. This helps to clarify 1st-order and higher-
645   order jitter for networks. The **subfigures (a) to (f)** represent the jitter in the traffic originating at N1 to N6
646   respectably. The values are shown in ms units and it shows both the negative and positive jitter values. The
647   negative jitter values represent the packet arriving in lesser duration in comparison to the last packet e.g.,
648   the change in the variance is negative. The figure shows that smooth jitter is experienced by the packets
649   originated at the node 1.

650   The N1 is located at a distance three hops from the destination node. The results show that there is steady
651   DV in every BI of the traffic profile. Some BIs have significantly low jitter in the range of 0.1ms to 0.3ms.
652   The other nodes show steady DV up to 0.5ms. As with N1 results it shows both the positive and negative
653   values of jitter. The negative results show the delay was lesser than when the last packet arrived. N2 is
654   located at a distance of 4 hops from the sink node. This is observed that N-2 also shows the same pattern
655   as N1, however, each BI has dissimilar values as compared to N-1. For N3, both of the positive and negative
656   values of variance are shown where the negative values represent the packet with shorter delay as compared
657   to the last packet received. The N4 is located at a distance of 3-hops from the sink node. It has maximum
658   variance up to $2ms$.

659   The maximum value for N4 is only for few packets and average jitter, a pattern like other nodes. The N5
660   is located at a distance of 2 hops from the sink node. It can be observed that it shows quite steady jitter for
661   N5 during the experimental transmission. The N6 is located at a distance of 2 hops from the sink node and
662   it follows a similar pattern as N5. In this section, a term steady jitter has been used. The steady jitter is
663   useful in many ways. It eliminates the presence of t of higher order variances such as 2nd order jitter.

664   ## Packet Drop Characteristics

665   Packet loss is likewise a significant parameter that depicts the performance of networks. Greater the packet
666   lost; the more are the re-transmissions on the network. Various methods exist to effectively react to re-
667   transmission requirements as opposed to retransmission of the entire portion of information such as
668   selective-repeat and TCP-Reno. This article focuses on limiting the packet loss because of buffer overflow
669   or queue overflow in the network with proficiently scheduling decisions. The packers are not allowed to
670   wait in queues to their max limits irrespective the class, weight, or priority excluding the case where
671   network allocations limits are reached, moreover, it guarantees an appropriate command over which packet
672   should be dropped if the system is working in CNC state. In this experimental evaluation, it used a MAP
673   application that produces traffic in such a manner that creates the CNC conditions of the network, and
674   traditional scheduling techniques begin to drop bundles. They drop packets as well as queuing time

675   overshoots to high values. Using the TIPS algorithm, the CNC conditions are handled with a suitable
676   intensity aware method to guarantee that packets are not dropped unless traffic limits are reached or
677   resources exhausted on the interface. **Figure 12** shows that packet drop results for all nodes. It shows that
678   there was not any packet loss at any point in the experimental network in CNC. Other traditional approaches
679   of the scheduling cause an irregular packet loss for given traffic profiles because of the surpassing of the
680   queue limits. Nonetheless, since the TIPS processes the queues according to TI values, queue limits remain
681   within limits.

B. TIPS in Moderate Operating Network Conditions (MNC)

Measurement Results of E2ED

684   E2ED is the difference in time of packet creation at a source node and the reception time at a sink node.
685   One side delay is called a one-way delay in contrast to the RTT that shows time of upstream and
686   downstream sides. E2ED also includes time consumed in packet transmission at source nodes, its
687   propagation, other processing at nodes and the waiting time in queues. The Equation(9) shows the
688   calculation method for the end-to-end delay where D_i represents the delay and T_{i+}^{S} represents the time
689   when a packet is created and T_{ir}^{R} represents the time when it is received at destination.

$$D_i = T_{i+}^{S} + T_{ir}^{R} \qquad\qquad (9)$$

690   Equation(9) i is ith packet sequence, $+, r$ represents the packet addition to the queue at the sending side
691   and the packet reception at the receiver side respectively. In the experimental setup, the propagation delay
692   is set to a constant value of $10\ ms$. In the comparison section, it will be shown that in MNC, E2ED generally
693   remains near to propagation delay value. However, if more traffic arrives, and the network gets into the
694   CNC state, the E2ED due to queuing processes becomes significant and values increase beyond the
695   propagation delay of single hop. **Figure 13** represents the E2ED of traffic originating in access nodes in
696   MNC conditions. The different colors in the figure represent traffic originated in the source nodes.

Throughput Results

698   Figure 14 illustrates the throughput of traffic originating in access nodes in MNC. It represents the PPS
699   (Packet Per Second) on y-axis and timescale on x-axis. The colors refer to the sustained throughput of
700   different nodes in different BIs. The Comparison of Figure 9 and Figure 14 shows that the proposed TIPS
701   algorithms exhibited similar results, both in CNC and MNC. The throughput in MNC shows that required
702   data rates are achieved. Other scheduling algorithms that are compared with TIPS algorithms also offer
703   similar results in MNC. But with CNC ,data rates were significantly degraded with all other scheduling
704   approaches.

End-to-End Jitter

706   As with the case of CNS, the jitter in MNC also follows the finest limits and average values remain less
707   than 1ms with spikes of $2ms$. **Figure 15 (a) to (f)** illustrates the jitter in ms units experienced by traffic
708   originated at N1 to N6 respectively. It shows both negative and positive values where the negative values
709   show that packet delay is less than the last sequence of the packet. It can be observed that it has a single hit
710   of jitter surpassing 1.5 milliseconds while normal jitter remains under 1ms limit. It can be observed that N2
711   has two spikes of jitter more than 1.5ms and the average value remains less than 1ms limit. The N3

712   experiences multiple spikes of jitter with more than 1.5ms limit and the average jitter remains at 1ms. N4
713   and has a single spike of jitter with maximum value more around $1.5ms$ and the average jitter remained
714   below 1ms. For N5 and N6 the jitter remains below $0.6ms$. These two nodes are two hops away, whereas
715   the first four nodes were at a distance of 3 hops from the destination. In a general network state, it can be
716   observed that delay variance remains around $0.3ms$ for each hop.

Packet Drop Behavior

718   Similar to the CNC results there was no packet loss observed for all nodes. However, it will be shown in
719   next sections that in CNC, The TIPS algorithm was able to achieve higher throughput and hence has higher
720   traffic volume delivery.

Discussion

Comparison with Other Algorithms

723   In order to compare the TIPS algorithm performance with others, the DRR, RR, FQ, RED, and SFQ were
724   also tested on identical network and traffic profiles as used for the TIPS algorithm evaluation.

A. Packet Drop Behavior

726   Different existing scheduling algorithms such as DRR, RR, FQ, RED, and SFQ were simulated on the
727   experimental network. The measurement results were captured both for the CNC and MNC. It can be
728   observed from the results that the TIPS algorithm provides a significantly improved performance in
729   comparison to the traditional schemes of scheduling. In the CNC state, there is an obvious distinction in
730   results of these approaches when contrasted with TIPS.

731   **Table IV** gives a synopsis of the rate of packet drop in CNC. The execution of all different methodologies
732   debased essentially though TIPS created the same outcomes as in the event of MNC. Packet loss greater
733   than 1% on a network significantly degrades the performance of real-time services and proves to be fatal
734   for critical and interactive services that require the responses to be received within stringent limits. For
735   other applications, greater packet losses decrease network utilization and efficiency due to frequent re-
736   transmissions. The frequent retransmission leads to further aggravation of congestion in the network.

737   Figure **16** illustrates the comparison of the successful packet delivery rate to the destination. The red line
738   refers to the successful delivery of packets. It shows a comparison of successful delivery of generated
739   packets at source and packet drop with different various packet scheduling algorithms. The SFQ has the
740   lowest successful delivery ratio of packets due to higher packet losses. On other hand, the proposed TIPS
741   algorithm exhibits the maximum successful delivery of generated packets. These results show that the TIPS
742   algorithm provides high efficiency in CNC and increases the network utilization, minimizing congestion
743   and providing optimization of overall performance.

B. Comparison of E2ED in CNC

745   This section provides the comparison of E2ED improvements with TIPS algorithm with other traditional
746   scheduling techniques. The method to calculate the E2ED has been also discussed in previous sections. The
747   E2ED comparison is from three different aspects such as the average, minimum and maximum delay
748   experienced by the packets from the source to destination over different nodes in CNC conditions. The
749   comparison of these aspects is given in **Figure 17**. It shows the comparison of E2ED maximum, minimum,
750   and average values in ms units as experienced by the traffic originated in different nodes. The delay values

751   in these figures are not inclusive of the propagation delay. N1 to N4 are at a distance of three hops from
752   destination and N5 and N6 both are at a distance of two hops. It can be observed that the traffic experiences
753   least delay in CNC with the TIPS algorithm where the maximum value of the delay experienced is limited
754   to $4ms$ and average values are in fractions of ms. The worst case can be observed for FQ where E2ED is
755   beyond $50ms$ and the average delay remains around $20ms$. This comparison shows that the performance
756   of all these schemes degrades in CNC state in comparison to the TIPS algorithm where performance is
757   observed efficiently for the E2ED.

C. Comparison of Throughput in CNC

759   The comparison of packet throughput attained with all the algorithms is presented here. The throughput is
760   calculated as the fraction of packets sent by the source and packet received by the respective sink nodes.
761   The throughput relationship follows a relationship as shown in the Equation **(10)** where P_t represents the
762   packet throughput, R_i represents the packets received by node j sent from node i and S_{ij} represents the
763   packets sent by node i to node j.

$$P_t = \left(\frac{\sum\limits_{i=1}^{n} R_{ij}}{\sum\limits_{i=1}^{n} S_{ij}} \right) * 100 \qquad (\,10\,)$$

764   The **Table V** shows that the TIPS algorithm is generally better to give 100% data delivery. This fact
765   depends on packets sent by the source nodes and packets received by the sink node. Since there was no
766   packet loss as we saw in previous sections, the TIPS algorithm gave significantly better throughput while
767   delivering the data in CNC conditions. These outcomes have a direct relationship to packet loss in the
768   network. The worst instance of packet loss was with the FQ approach that has the lowest throughput of
769   effective delivery rate. Figure **18** illustrates the analysis of the data delivery with different algorithms in
770   comparison to the FOBPB and **Figure 19** shows throughput with different algorithms normalized to
771   TIPS. Furthermore, the comparison of the actual delivered amount of data is shown in **Figure 20**. It is
772   evident from these results that the TIPS algorithm provides the best data delivery rate in comparison to all
773   other scheduling techniques and the max target data volume determined by the FOB process. It provided
774   96.5% successful data delivery ratio and if adding 3.6% MAP overhead it achieved approx. 100% data
775   delivery ratio.

D. Comparison of Delay Variance

777   The comparison of delay variance seen by packets from different nodes during the experiment is shown in
778   Table **VI**. The maximum variance faced by packets with the TIPS algorithm is $2.09ms$ with a maximum
779   distance of 3 hops. The average value of variance is $0.35ms$ and minimum variance is observed around the
780   value of $0.11ms$. With the TIPS algorithm the maximum jitter is minimized by thirty to forty-five percent
781   in comparison to other scheduling algorithms. It shows significant improvement to the data delivery
782   achieved with the TIPS algorithm.

Other Algorithms

784   There are some other scheduling schemes such as class-based queuing(CBQ), Class based Weighted Fair
785   Queuing(CBWFQ) and Priority Queuing(PQ) that provide some additional approaches to assign the
786   weights or priority to the different packets based on some user criteria. These algorithms work fine for
787   prioritizing traffic classes or groups, but these suffer from issues if traffic characteristics are changed.
788   Additionally, the limited number of classification methods puts limits on their performance. Furthermore,
789   these algorithms support only static configurations and cannot cope with the changes in the traffic
790   characteristics dynamically.

791   ## Applications

792   We have seen in previous sections; the TIPS algorithm takes advantage of diversity in traffic characteristics
793   across nodes. The typical application scenario is the outgoing interfaces on aggregation and core nodes of
794   networks. The underlying reason is that one such application involves a number of POPs. With the
795   traditional approaches the rule configuration may become complex.   Another case is to utilize it in
796   aggregation and core nodes or regional and national core nodes in telecommunication IP/Optical Transport
797   Networks. Similar logic applies here as above as there is diversity in traffic originating from different. In
798   web edge hubs, TIPS can essentially make distinction in deferral, jitter and parcel drop difficulties. Other
799   potential applications incorporate WAN connections in big business organizations, where TIPS can give
800   better execution and organization usage because of lower deferrals, dormancy, and bundle drops. Higher
801   these qualities the lower network execution and organization usage as end clients would retransmit the lost
802   or dropped packets or breaks because of enormous measures of delay.

803   ## Future Research Directions

804   With the advancements in artificial intelligence and service requirements, the networking domain is also
805   required to use the machine learning based techniques to help the autonomous and cheaper and faster
806   decision-making in managing the network functionalities. There are various ML based approaches that can
807   predict the traffic forecast based on the analysis of the previous historical usage of the data. These
808   algorithms can extract the Spatio-temporal insights from the data and make predictions for the future.
809   Furthermore, there are ML based techniques that can provide the efficient traffic classification considering
810   the various aspects of previous network usage. With the machine learning based approaches the TIPS shall
811   be able to work efficiently. Extreme learning, the Bayesian network and decision tree can classify the
812   internet traffic with the focus on peer-to-peer applications (Sena & Belzarena, 2012). The traffic
813   classification algorithms must provide classification information to other systems as soon as possible to
814   make necessary decisions. Support vector machine based Early Traffic classification models can address
815   this issue (L. He et al., 2016). The ML models like Multi-Layer-Perception, Radial basis function, decision
816   tree based C4.5, Bayesian networks can also provide online or offline traffic classification. These models
817   can also classify the traffic using only the partial captures, i.e., only few packets of flow and allows the
818   reduction of data volume to be analyzed (Jin et al., 2012; Luo et al., 2008; Nguyen et al., 2012; Singh &
819   Agrawal, 2011; C. Zhang et al., 2019). AdaBoost based ML models can classify the traffic based on flows
820   rather than per packet classification. It detects the flows or transactions in the captured traffic and associates
821   the related packets to flow as per the flow definition (Kong et al., 2018). Recurrent Neural Networks has
822   the specific capability to detect the patterns from the time series data. In this regard, the long-short-term-
823   memory based traffic model is proposed by (Z. He et al., 2019; Reddy & Hota, 2013; C. Zhang et al., 2019).
824   These models can predict the traffic forecast on a daily basis to long term periods where the first kind of
825   forecast is useful for the day-to-day optimizations and resource allocation and second can be used for long-

826 term planning for the network. The convolutional neural network-based model has the capability to consider
827 over-subscription of resources, SLA violations and can also detect and incorporate the spatio-temporal
828 dependencies for the long-term forecast (Bega et al., 2019; Reddy & Hota, 2013). A progressive transfer
829 learning model can provide short term forecast and long-term traffic forecasting for individual POP
830 locations in the operator's network (Z. He et al., 2019). By using above ML models, traffic profiles can be
831 dynamically learnt, and scheduling decisions are efficiently made in order to improve the QoS and network
832 utilization. In addition to intelligent scheduling decisions, TIPS need to evaluate wireless networks with a
833 focus on cellular networks. The traffic intensity signaling between nodes needs further research and it needs
834 to be evaluated with other traditional algorithms also.

Conclusions

836 In a detailed discussion on simulation results, we saw the TIPS approach can significantly enhance
837 performance of a network, its utilization, and quality of services. It provides opportunities to maximize
838 network utilization to 100% as compared to 95% to 97% with other approaches. We have also seen that
839 networks exhibit better performance in regard to minimizing delay caused by queuing in CNC. Maximum
840 enhancements to delay are several hundred percent reductions as compared to FQ, DRR, RR and SFQ. This
841 is an excellent improvement in response time especially for real-time services and interactive services that
842 have strict bounds for response time. We also saw TIPS provide 45% jitter reduction as compared to other
843 approaches. We also saw that TIPS can ensure required throughput targets with the best performance and
844 optimal network utilization. In addition to this there was no packet drop in CNC which significantly reduces
845 wastes of network resources used to retransmission of same packets. TIPS make use of the difference in
846 traffic intensities in different time, origin, variations to make efficient packet scheduling decisions. In CNC
847 cases, where services performance is degraded with other existing approaches, TIPS can effectively
848 maintain services performance, it also helps in managing congestion by optimizing packet drop
849 probabilities and end-to-end delay for packets which is also a factor of time-out of sessions or
850 retransmissions. Thus, the need for minimal delay and jitter becomes very important as network diameter
851 increases; it can dramatically provide opportunities to optimize network utilization.

References

853 Angus, I. A. N. (2000). An Introduction To Erlang-B and Erlang-C. *Telemanagment No.187*.

854 Bega, D., Gramaglia, M., Fiore, M., Banchs, A., & Costa-Perez, X. (2019). DeepCog- Cognitive Network
855 Management in Sliced 5G Networks with Deep Learning. *IEEE INFOCOM 2019 - IEEE Conference on
856 Computer Communications*, 280–288.

857 Choi, Y. (2008). A Network Dimensioning Framework for QoS-guaranteed IP Networks. *NOMS
858 Workshops 2008 - IEEE Network Operations and Management Symposium Workshops*, 66–70.

859 Dahmouni, H., Girard, A., & Sansò, B. (2012). An analytical model for jitter in IP networks. *Annals of
860 Telecommunications - Annales Des Télécommunications*, 67(1), 81–90.

861 Davies, G., Hardt, M., & Kelly, F. (2004). Come the revolution—Network dimensioning, service costing
862 and pricing in a packet switched environment. *Telecommunications Policy*, 28(5), 391–412.

863 Deb, P., Podder, D., Deb, S., & Majumder, A. (2016). Study of fairness and delay of packet scheduling in
864 IEEE 802.11 based Wireless Mesh Network. *3rd International Conference on Recent Advances in
865 Information Technology*, 101–106.

866 Deshmukh, A., & Vaze, R. (2016). Online Energy-Efficient Packet Scheduling for a Common Deadline
867 With and Without Energy Harvesting. *IEEE Journal on Selected Areas in Communications*, 34(12), 3661–
868 3674.

869 Dockers. (2012). Docker Containers. Docker Website.

870 Glabowski, M., & Stasiak, M. D. (2012). Internal blocking probability calculation in switching networks
871 with additional inter-stage links and engset traffic. *8th International Symposium on Communication*
872 *Systems, Networks Digital Signal Processing*, 1–6.

873 Han, D., You, K., Xie, L., Wu, J., & Shi, L. (2015). Stochastic packet scheduling for optimal parameter
874 estimation. *54th IEEE Conference on Decision and Control*, 3057–3062.

875 He, L., Xu, C., & Luo, Y. (2016). VTC- Machine Learning Based Traffic Classification as a Virtual
876 Network Function. *In proceedings of the ACM International Workshop on Security in Software Defined*
877 *Networks and Network Function Virtualization*, 53–56.

878 He, Z., Chow, C.-Y., & Zhang, J.-D. (2019). STCNN- A Spatio-Temporal Convolutional Neural Network
879 for Long-Term Traffic Prediction. *20th IEEE International Conference on Mobile Data Management*, 226–
880 233.

881 Huang, S., Izquierdo, E., & Hao, P. (2016). Adaptive packet scheduling for scalable video streaming with
882 network coding. *IEEE International Conference on Communications*, 1–6.

883 Ian Berry, Tony Roman, Larry Adams, Pasnak, J. P., Jimmy Conner, Reinhard Scheck, & Andreas Braun.
884 (2012). *Cacti Manual, The Complete RRDTool-based Graphing Solution*. Cacti Group.

885 Iqbal, M. F., Holt, J., Ryoo, J. H., Veciana, G. de, & John, L. K. (2016). Dynamic Core Allocation and
886 Packet Scheduling in Multicore Network Processors. *IEEE Transactions on Computers*, 65(12), 3646–
887 3660.

888 ITU-T-G.8261.1/Y.1361.1. (2012). Packet delay variation network limits applicable to packet-based
889 methods. *ITU-T Spec. G.8261.1/Y.1361.1*.

890 Jin, Y., Duffield, N., Erman, J., Haffner, P., Sen, S., & Zhang, Z.-L. (2012). A Modular Machine Learning
891 System for Flow-Level Traffic Classification in Large Networks. *ACM Transactions on Knowledge*
892 *Discovery from Data,* 6(1), 4:1-4:34.

893 Kaur, G., & Singh, T. P. (2015). A Weighted Fair Queue based SBPN (WFQ-SBPN) Algorithm to Improve
894 Qos for Multimedia Application in Mobile Ad Hoc Networks. *International Journal of Computer*
895 *Applications*, 112(2), 13–17.

896 Lee, S. J., & Choi, Y. S. (2015). GMPS(Group based multi-level packet scheduling) method in multi-beam
897 based mobile communication system. *International Conference on Information and Communication*
898 *Technology Convergence* , 1050–1053.

899 Luo, Y., Xiang, K., & Li, S. (2008). Acceleration of decision tree searching for IP traffic classification.
900 *Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications*
901 *Systems*, 40–49.

902 Miao, W., Min, G., Wu, Y., & Wang, H. (2015). Performance Modelling of Preemption-Based Packet
903 Scheduling for Data Plane in Software Defined Networks. *IEEE International Conference on Smart*
904 *City/SocialCom/SustainCom*, 60–65.

905 Mishra, A., & Venkitasubramaniam, P. (2016). Anonymity and Fairness in Packet Scheduling: A
906 Quantitative Tradeoff. *IEEE/ACM Transactions on Networking*, 24(2), 688–702.

907 MRTG. (2012). The Multi Router Traffic GrapherMRTG. *MRTG Website*

908   Nguyen, T. T. T., Armitage, G., Branch, P., & Zander, S. (2012). Timely and Continuous Machine-
909   Learning-Based Classification for Interactive IP Traffic. *IEEE/ACM Transactions on Networking*, 20(6),
910   1880–1894.

911   NS2. (2012). The Network Simulator—NS2. *NS2 Website*.

912   Patel, Z., & Dalal, U. (2016). Implementation and evaluation of dynamically weighted low complexity fair
913   queuing (DWLC-FQ) algorithm for packet scheduling in WiMAX networks. *China Communications*,
914   13(5), 128–140.

915   Pavithira, S. D., & Prabakaran, N. (2016). Downlink packet scheduling mechanism in long term evolution
916   technology. *International Conference on Circuit, Power and Computing Technologies*, 1–4.

917   Reddy, J. M., & Hota, C. (2013). P2P traffic classification using ensemble learning. *Proceedings of the 5th*
918   *IBM Collaborative Academia Research Exchange Workshop*, 1–4.

919   RFC-3339. (2012). IP Packet Delay Variation Metric for IP Performance Metrics. *IETF RFC3339*.

920   Sharifian, A., Schoenen, R., & Yanikomeroglu, H. (2016). Joint Realtime and Nonrealtime Flows Packet
921   Scheduling and Resource Block Allocation in Wireless OFDMA Networks. *IEEE Transactions on*
922   *Vehicular Technology*, 65(4), 2589–2607.

923   Singh, K. & Agrawal. (2011). Feature extraction based IP traffic classification using machine learning.
924   *Proceedings of the International Conference on Advances in Computing and Artificial Intelligence*, 208–
925   212.

926   Striegel, A., & Manimaran, G. (2002). Packet scheduling with delay and loss differentiation. *Computer*
927   *Communications*, 25(1), 21–31.

928   Sungjoo Park, Kim, D., Chang Mo Yang, & Chai-Jong Song. (2016). A packet scheduling scheme for
929   seamless transmission of life media contents. *Eighth International Conference on Ubiquitous and Future*
930   *Networks*, 478–480.

931   Xu, S., Chang, T., Lin, S., Shen, C., & Zhu, G. (2016). Energy-Efficient Packet Scheduling With Finite
932   Blocklength Codes: Convexity Analysis and Efficient Algorithms. *IEEE Transactions on Wireless*
933   *Communications*, 15(8), 5527–5540.

934   Yu, Q., Znati, T., & Yang, W. (2015). Energy-efficient, Delay-aware packet scheduling in high-speed
935   networks. *IEEE 34th International Performance Computing and Communications Conference*, 1–8.

936   Zakariyya, I., & Rahman, M. N. A. (2015). Bandwidth Guarantee Using Class Based Weighted Fair Queue
937   (CBWFQ) Scheduling Algorithm. *International Journal of Digital Information and Wireless*
938   *Communications*, 5(3), 152–157.

939   Zhang, C., Zhang, H., Qiao, J., Yuan, D., & Zhang, M. (2019). Deep Transfer Learning for Intelligent
940   Cellular Traffic Prediction Based on Cross-Domain Big Data. *IEEE Journal on Selected Areas in*
941   *Communications*, 37(6), 1389–1401.

942   Zhang, J., Qi, H., Guo, D., Li, K., Li, W., & Jin, Y. (2015). ATFQ: A Fair and Efficient Packet Scheduling
943   Method in Multi-Resource Environments. *IEEE Transactions on Network and Service Management*, 12(4),
944   605–617.

945   Zhang, L., Maple, J., & Guo, C. (2003). Coverage and Capacity Calculations for 3G Mobile Network
946   Planning. *University of Luton*

947   Zhang, Z.-L., Duan, Z., & Hou, Y. T. (2001). Fundamental trade-offs in aggregate packet scheduling.
948   *Scalability and Traffic Control in IP Networks*, 4526, 299–308.

Figure 1

Algorithm Initialization

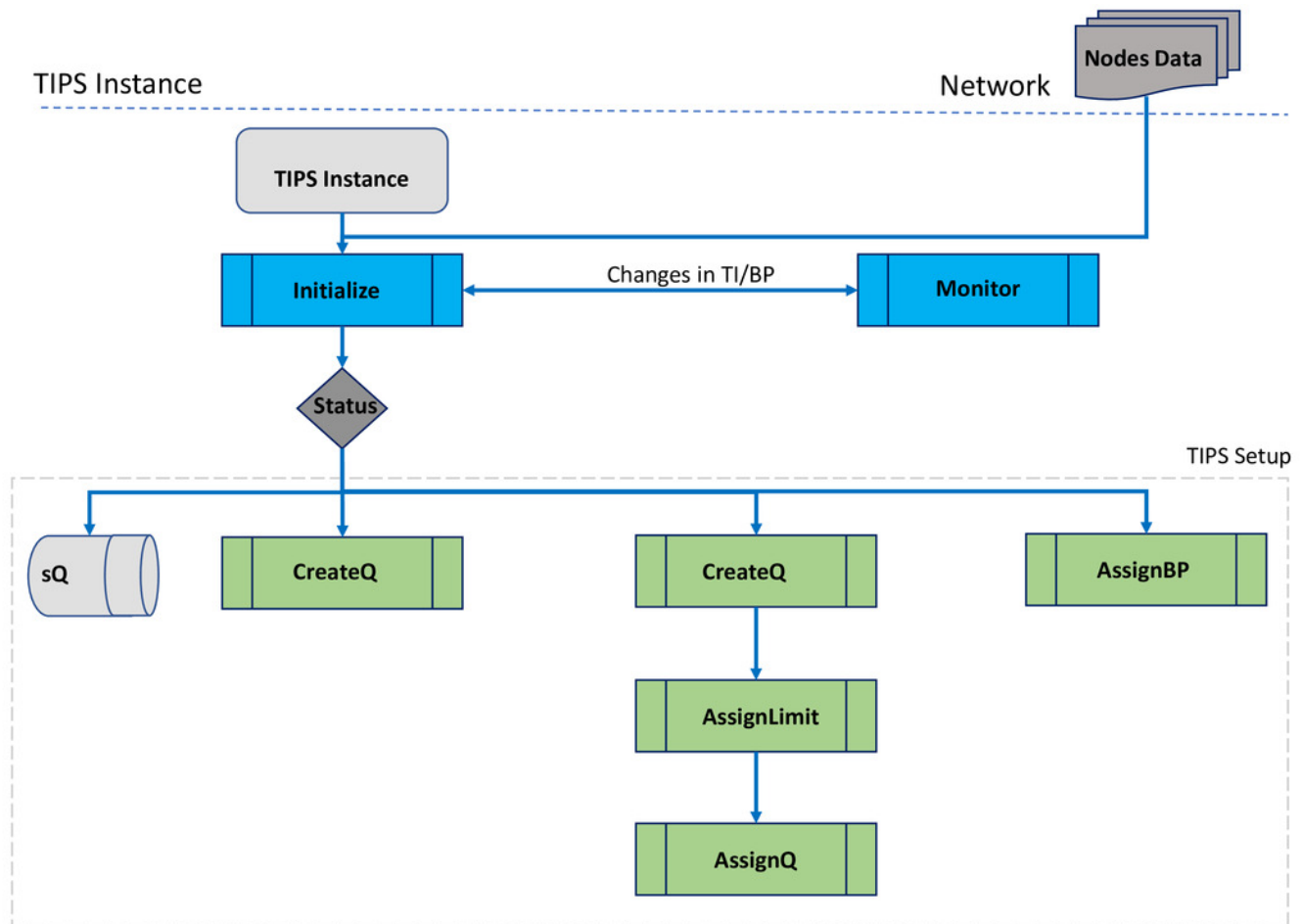



Figure 2

Processing on Received Side

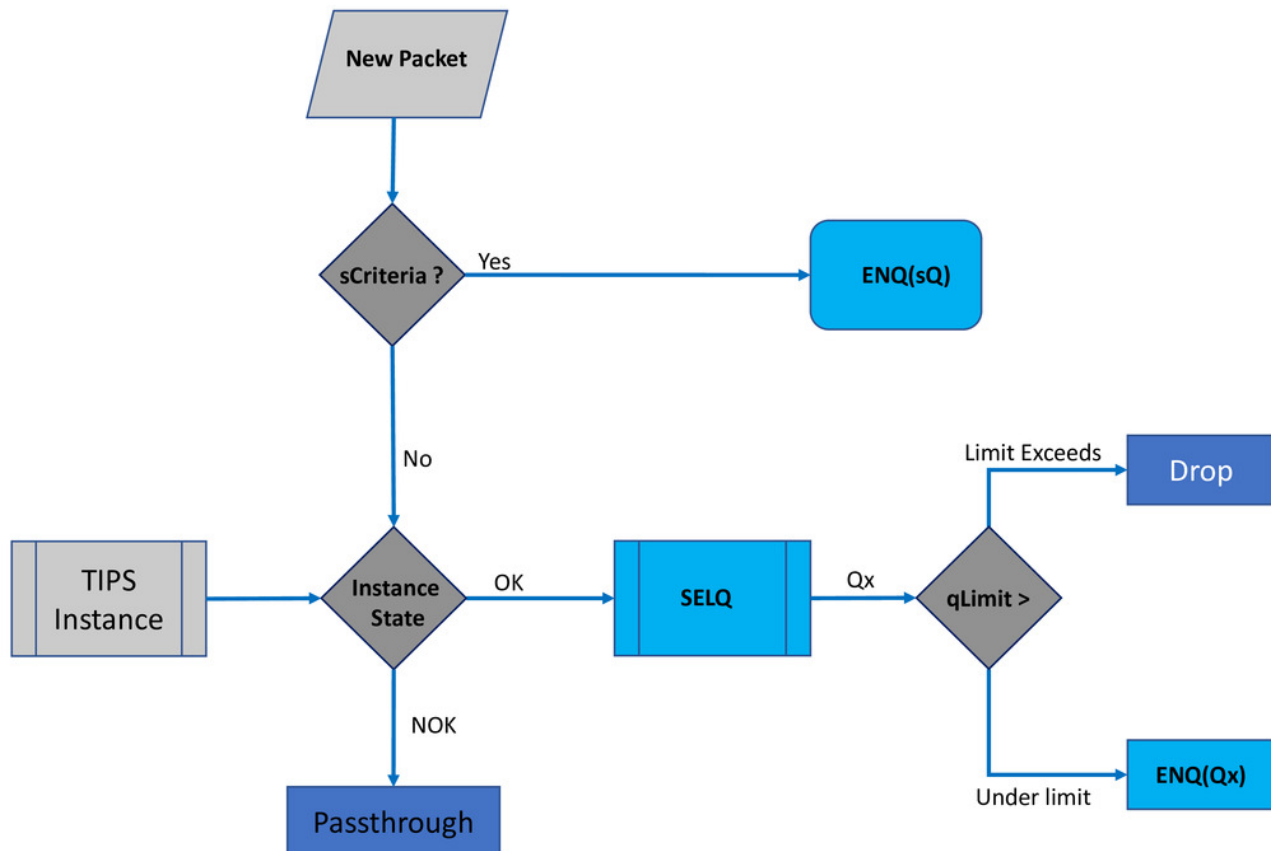



Figure 3

Processing on Transmit Side

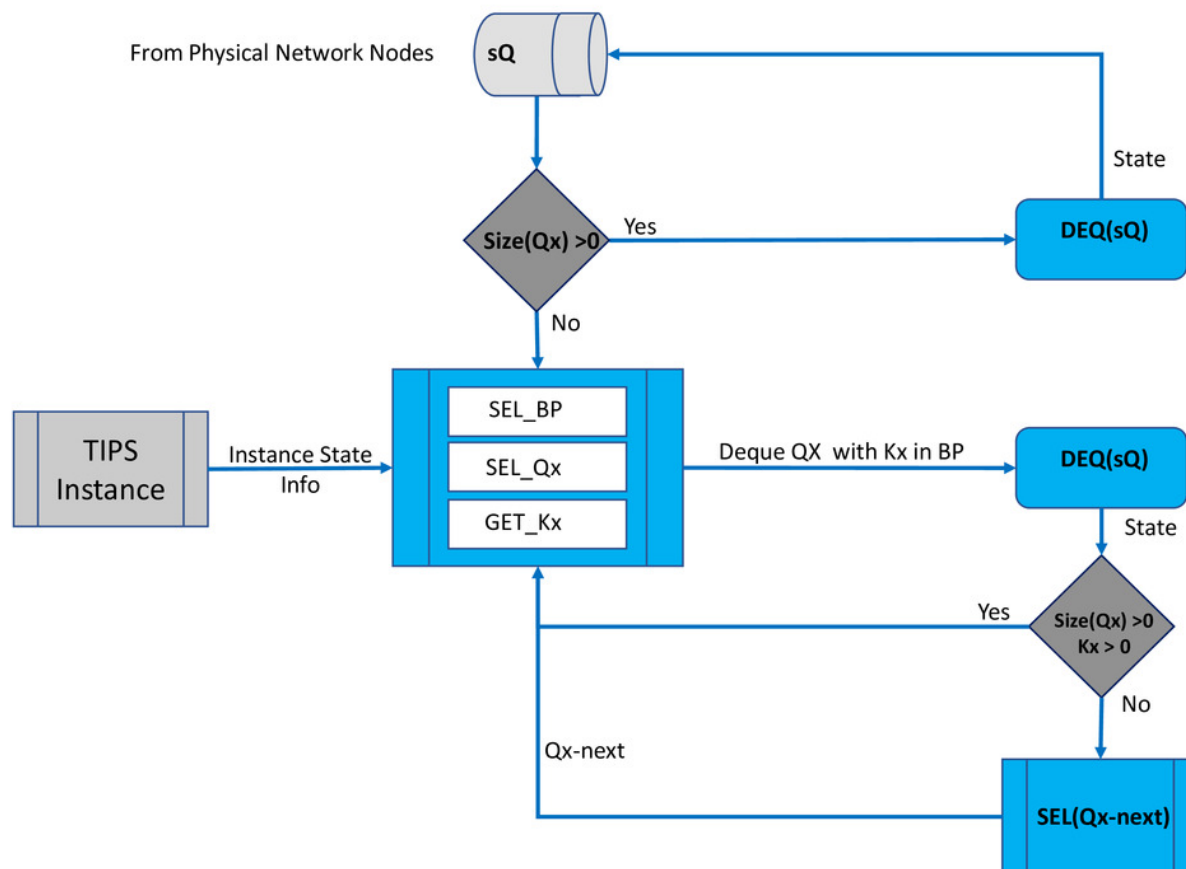



Figure 4

Process Steps for HNUB Traffic Profiling

Figure 5

ASB Traffic Profiling

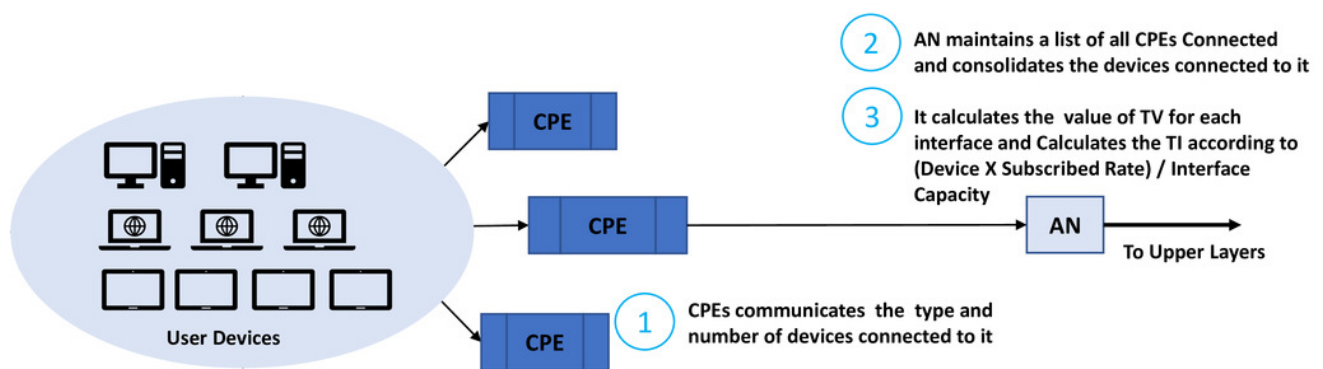



Figure 6

Integration with TIPS

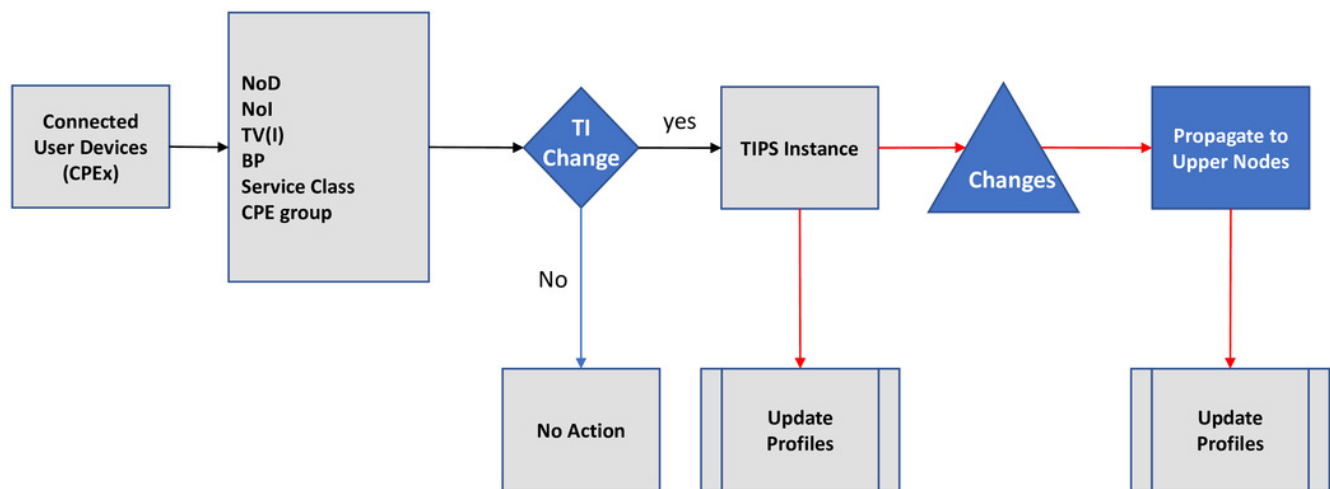



Figure 7

Network Topology for Evaluation and Comparison

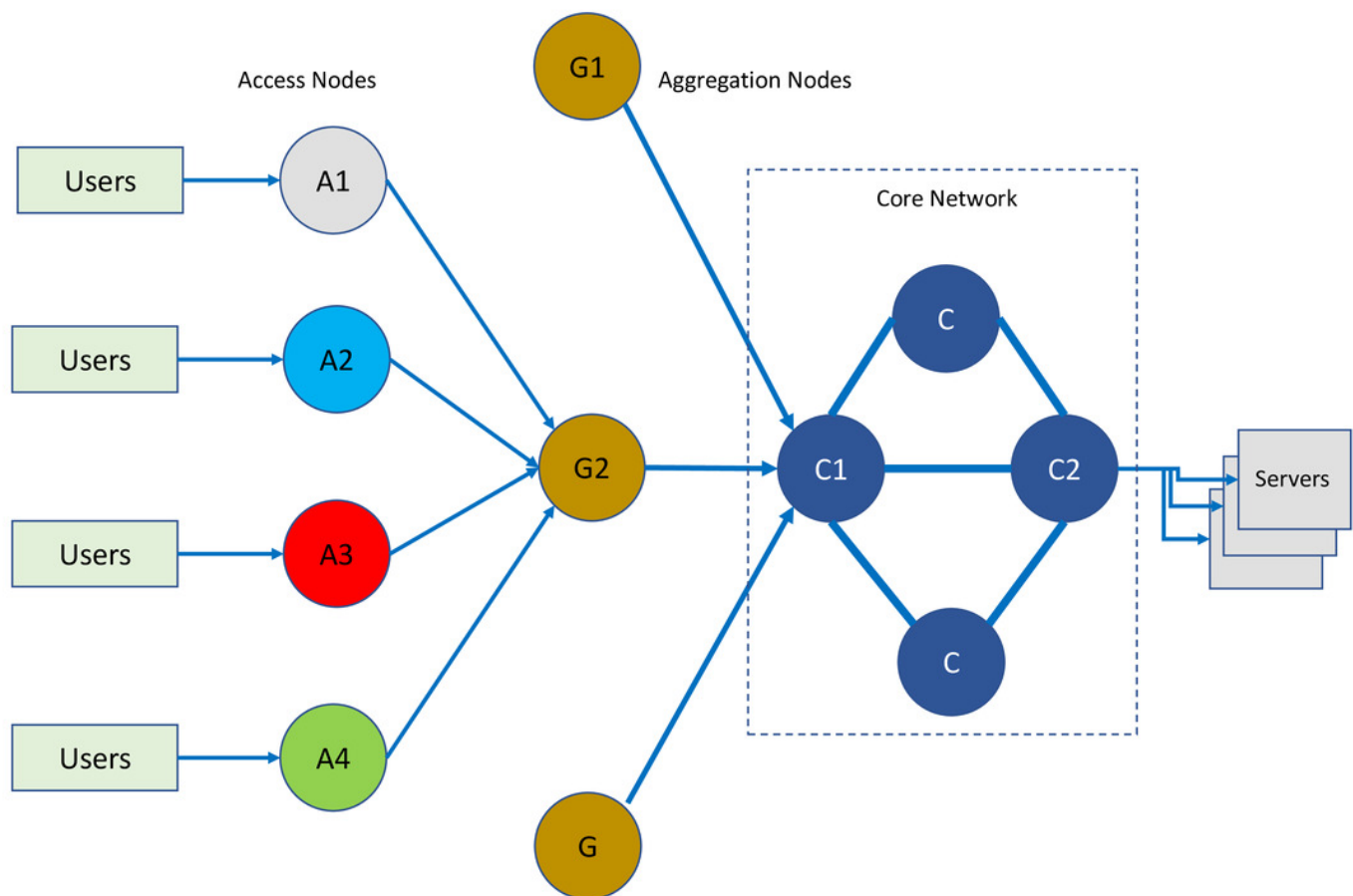



Figure 8

End-to-End Delay for CNC

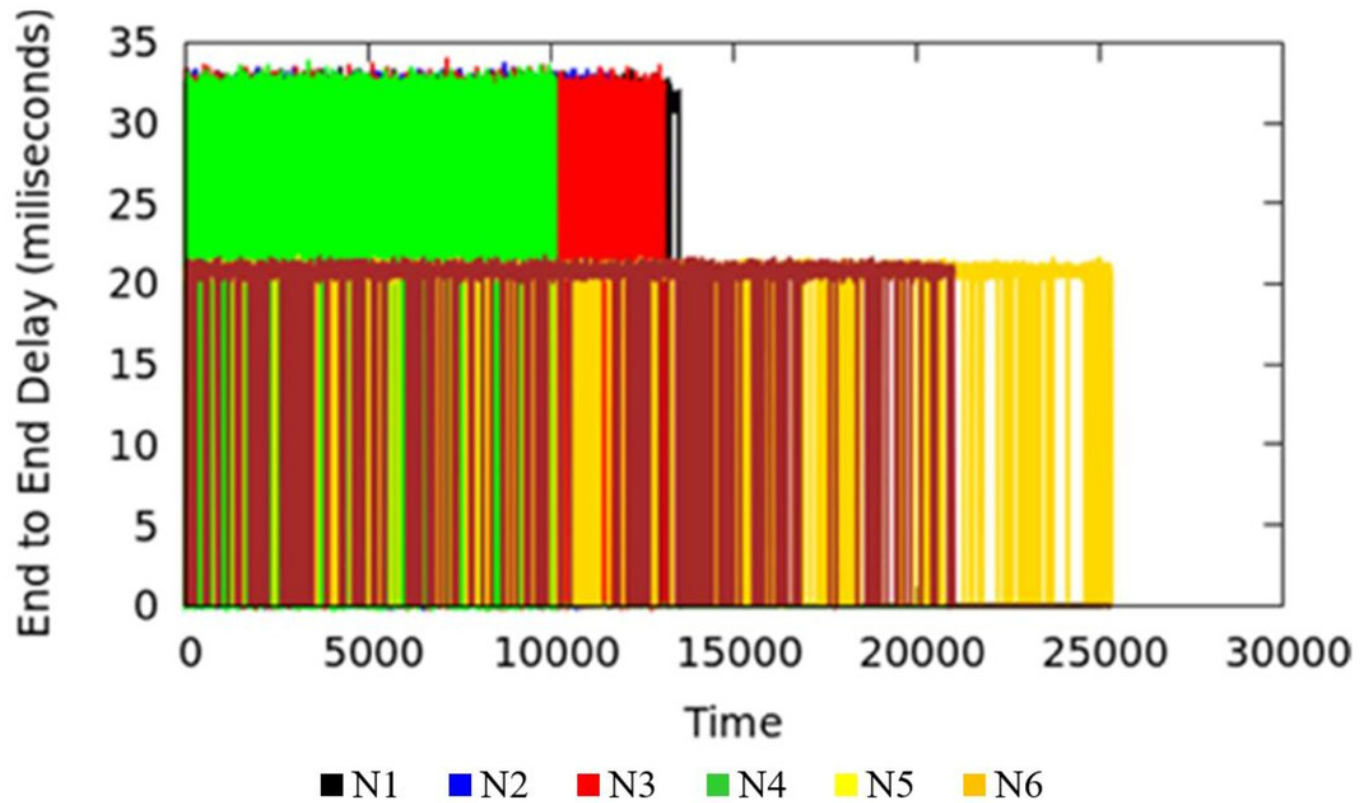



Figure 9

CNC Throughput

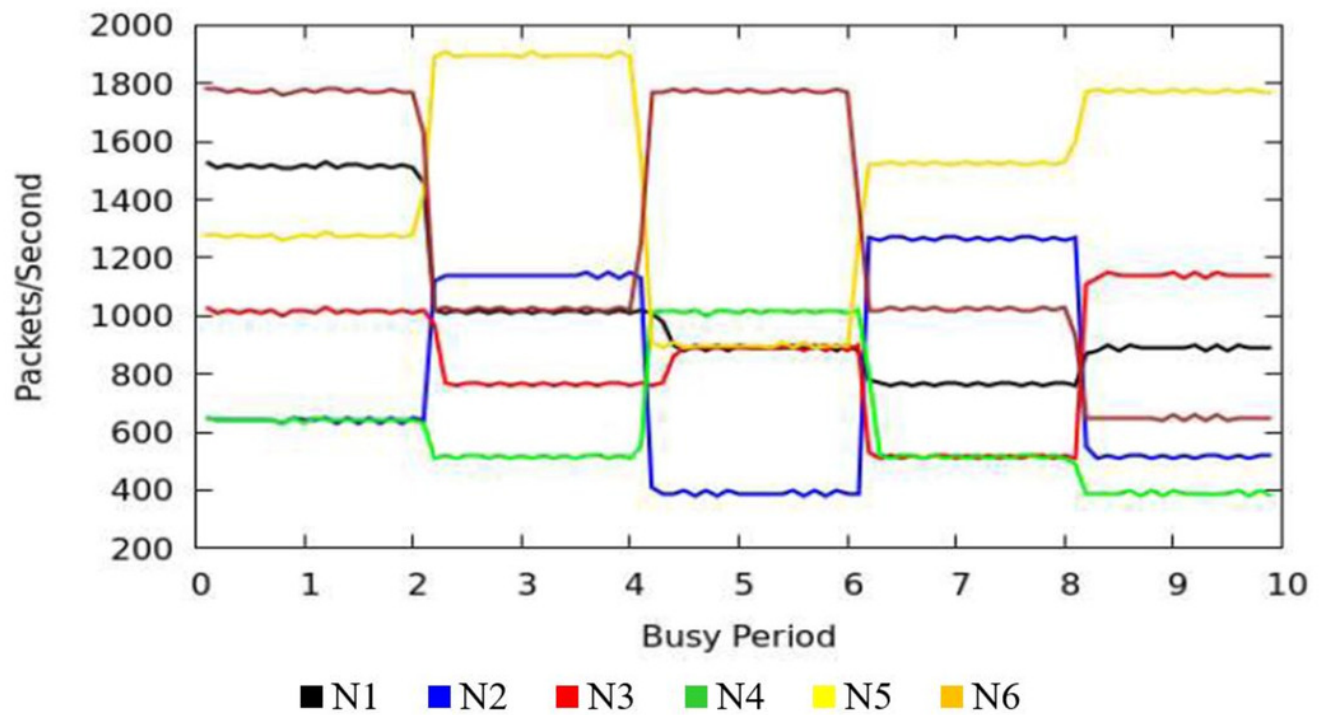



Figure 10

Data Volume Delivery of TIPS with FOB

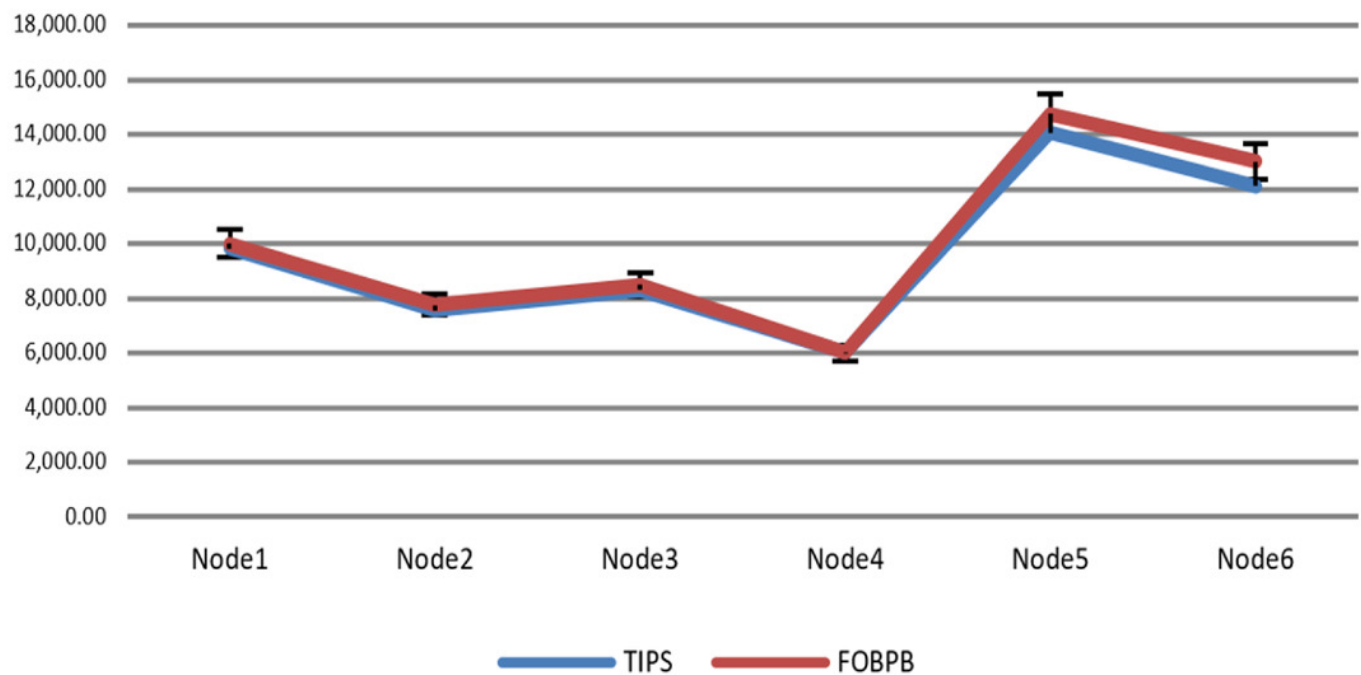



Figure 11

Jitter Results in CNC

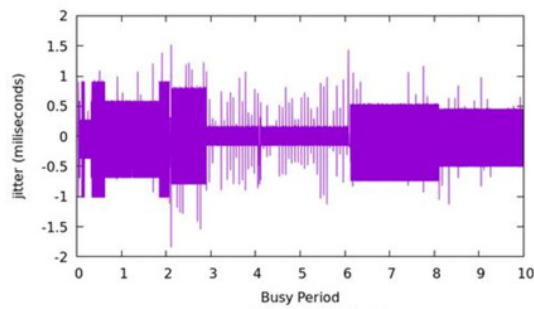
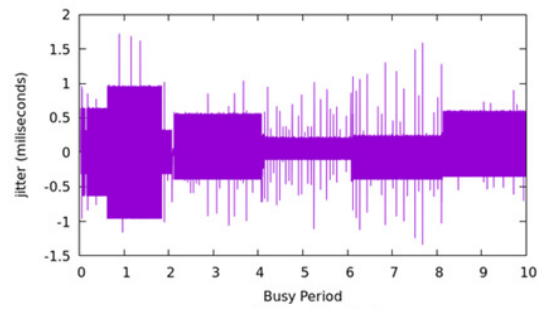
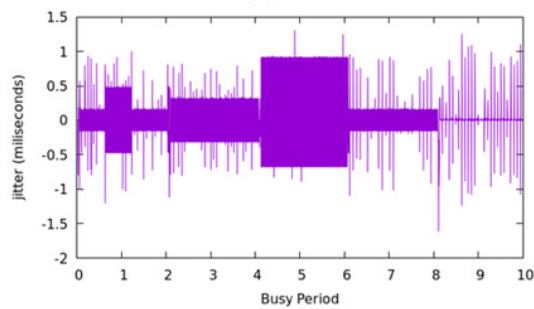
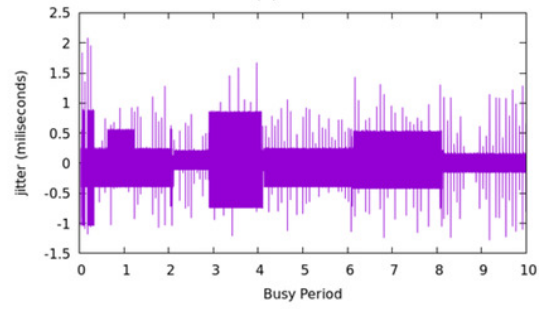
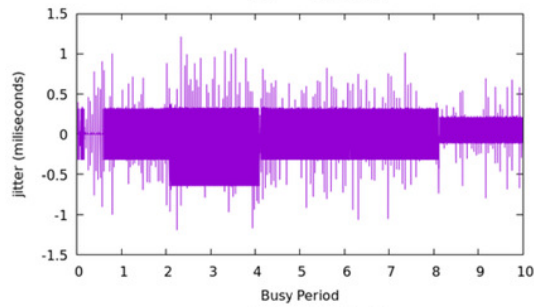
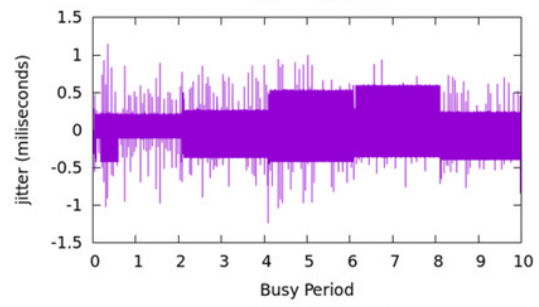



Figure 12

CNC Packet Drop for All Nodes

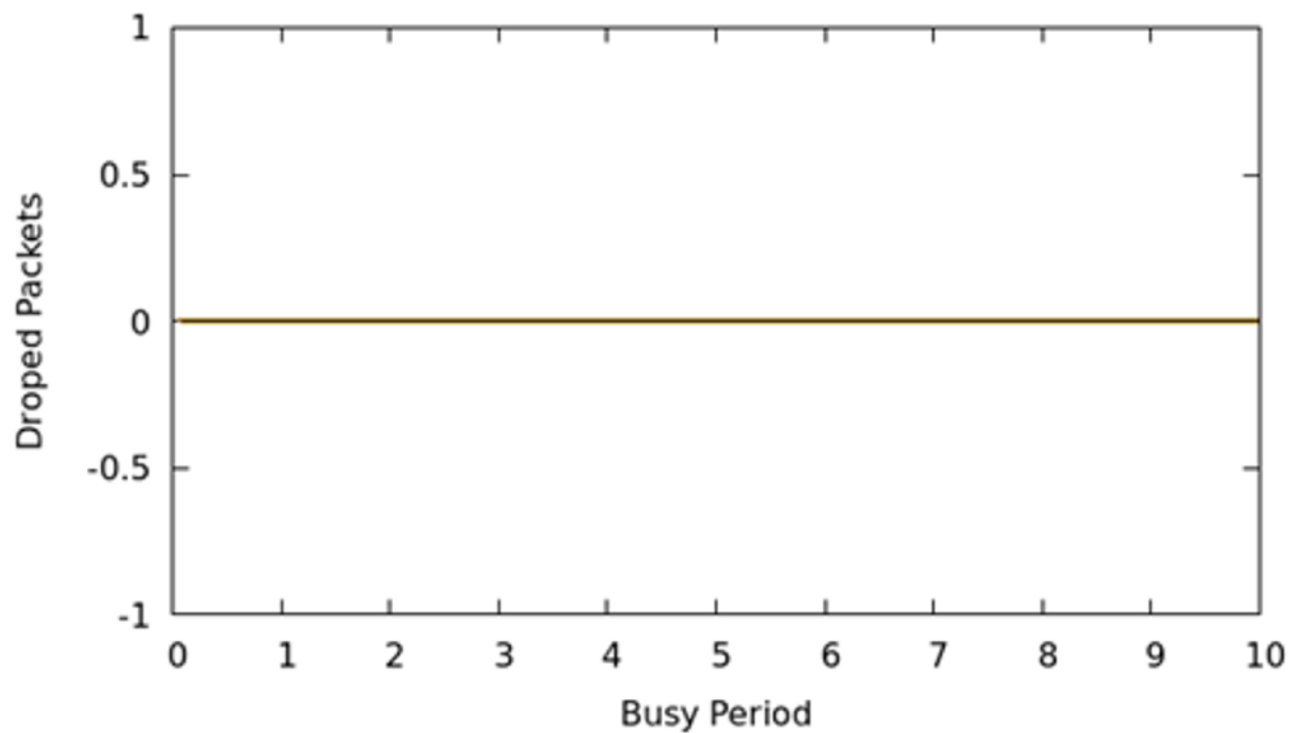



Figure 13

End to End delay in MNC

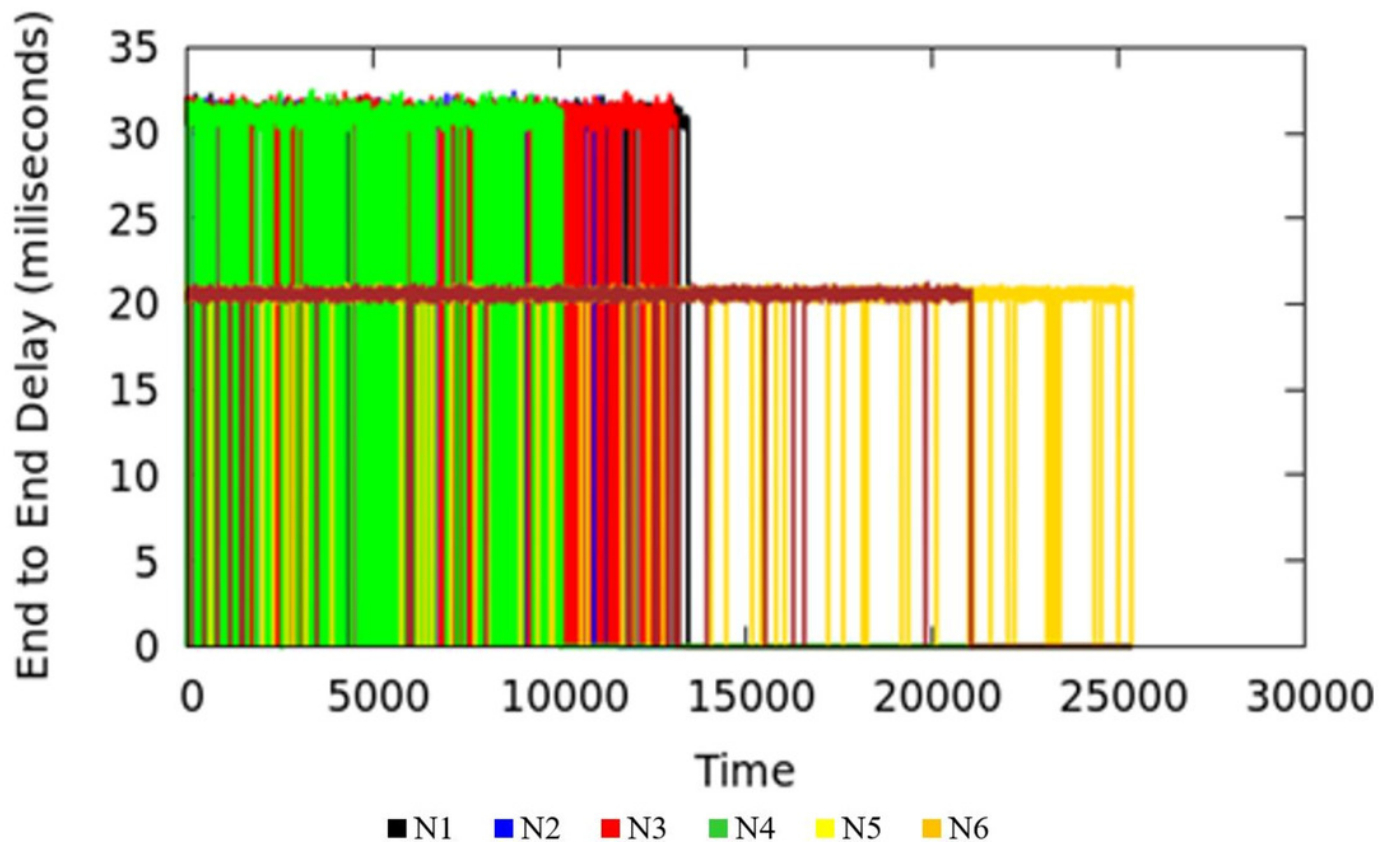



Figure 14

Throughput in MNC

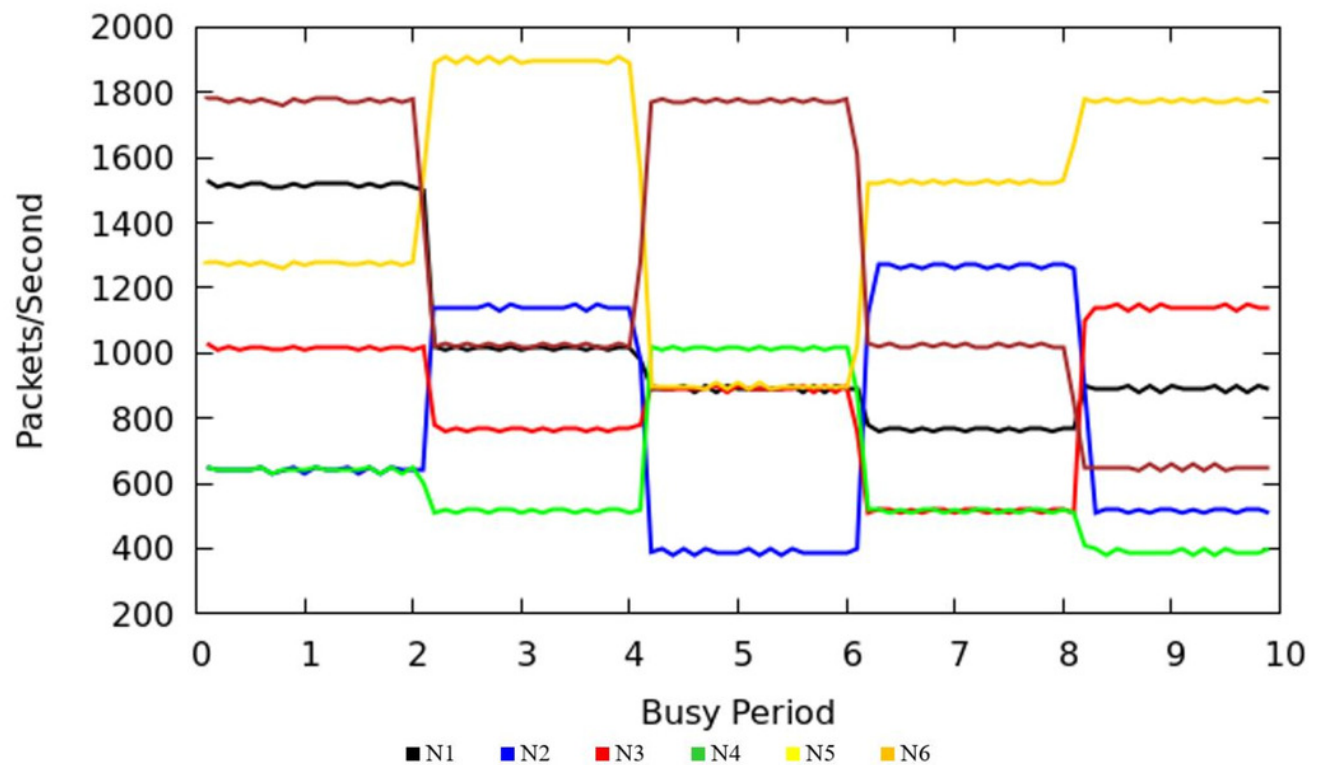



Figure 15

End to End Jitter in MNC

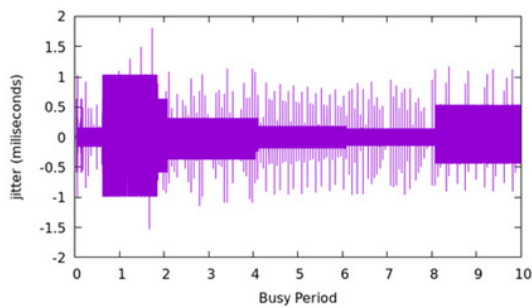
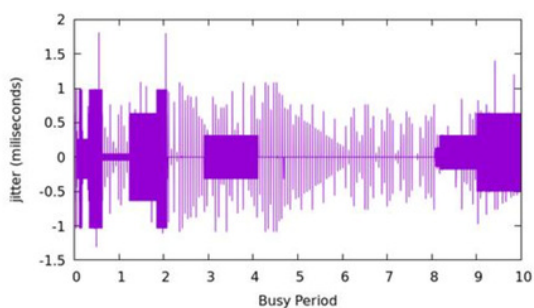
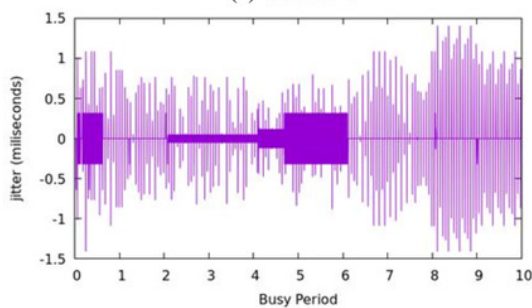
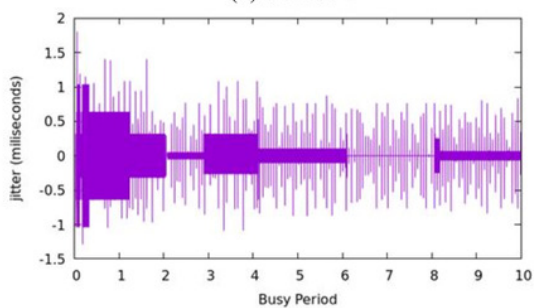
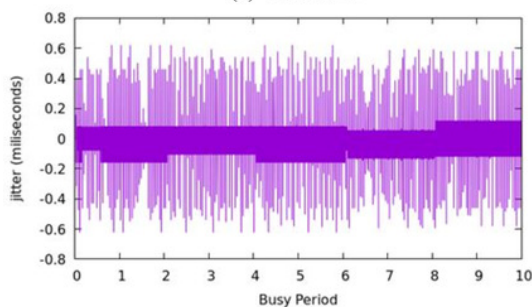
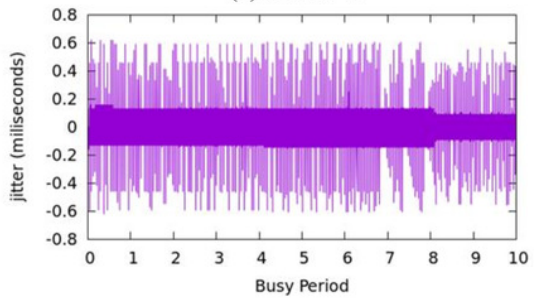



Figure 16

Data Rate in MNC

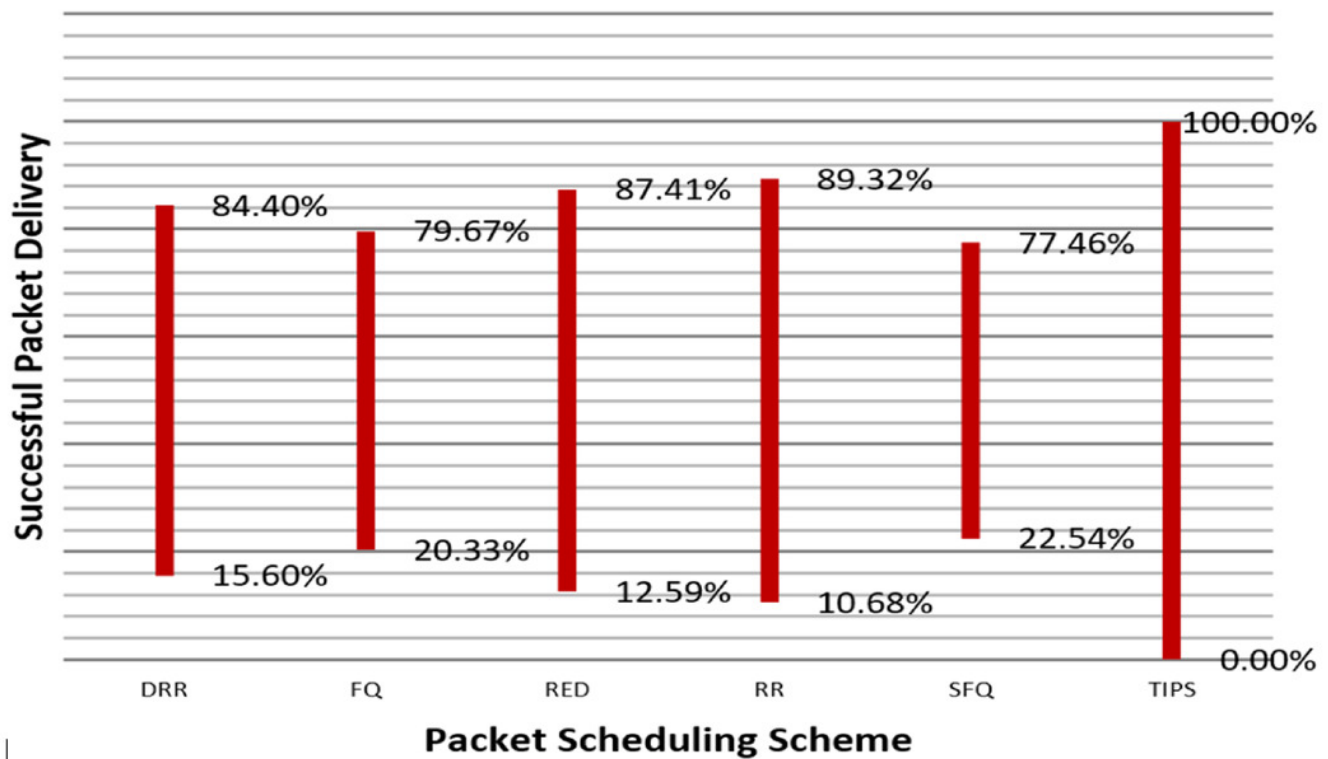



Figure 17

Comparison of delay in milliseconds experience by the traffic

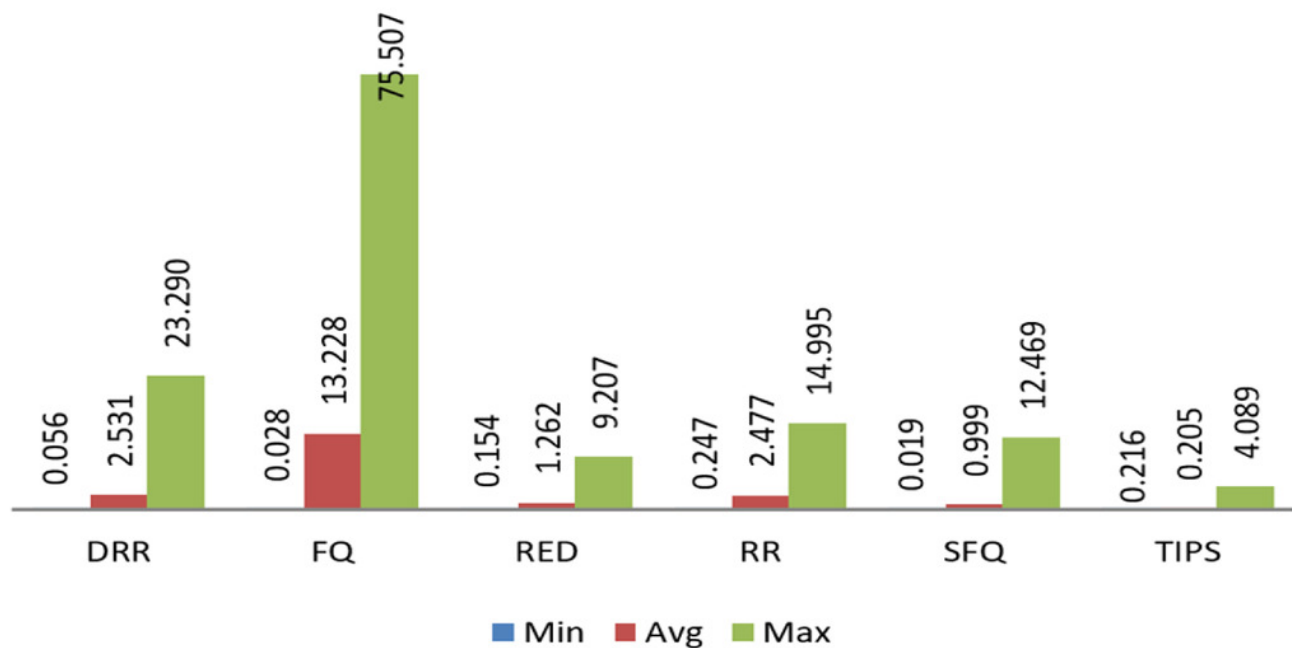



Figure 18

Traffic Volume Comparison with TIPS

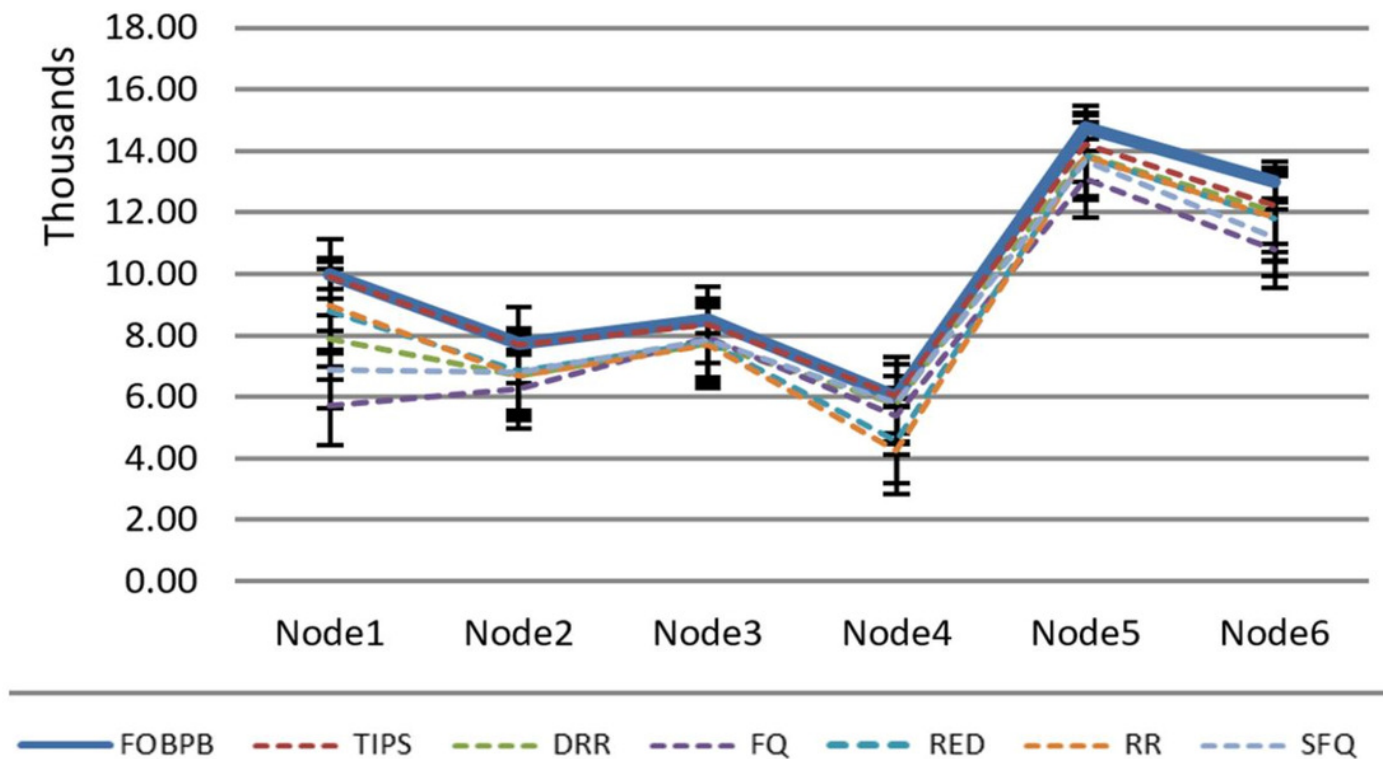



Figure 19

Comparison of throughput Normalized to TIPS

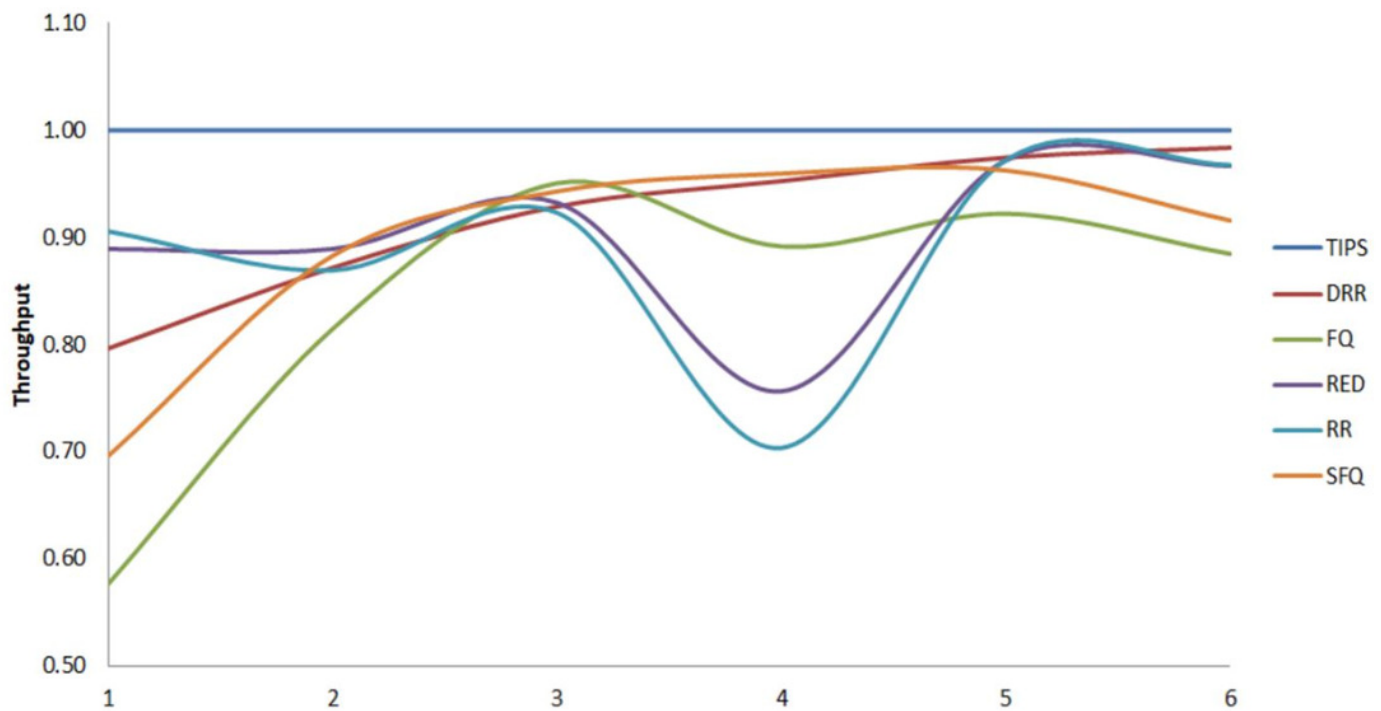
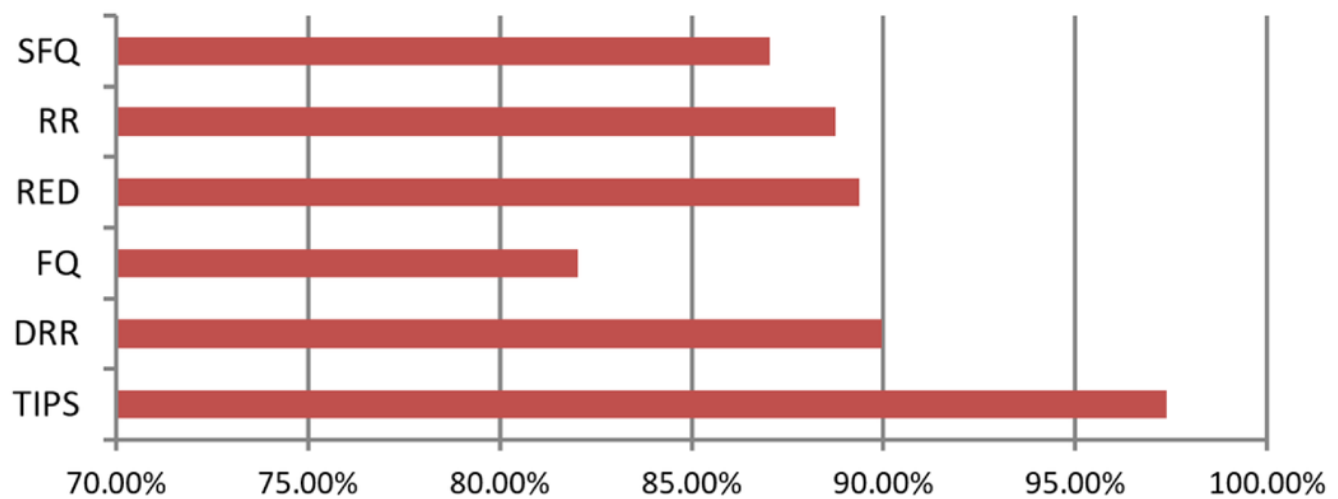



Figure 20

Delivery Percentage of Traffic Volumes

Link Capacities

(a) - For CNC Scenario

#	Links	Max Bandwidth (Kbps)
1	A1 . G2	25000
2	A2 . G2	25000
3	A3 . G2	25000
4	A4 . G2	25000
5	G1 —ÀCN1	25000
6	G3 —ÀCN1	25000
7	G2 —ÀCN1	25000
8	CN1 —ÀCN2	50000

(b) - For MNC Scenario

#	Links	Max Bandwidth(Kbps)
1	A1 . G2	25000
2	A2 . G2	25000
3	A3 . G2	25000
4	A4 . G2	25000
5	G1 —ÀCN1	50000
6	G3 —ÀCN1	25000
7	G2 —ÀCN1	25000
8	CN1 —ÀCN2	75000

Distribution of Traffic Volumes

(a) - Distribution of Traffic Volumes   (b) – FOB BI Distribution

BI	N_1	N_2	N_3	N_4	N_5	N_6
BI_1	2,985	404	1,992	746	2,522	3,511
BI_2	1,950	1,840	1,538	1,008	3,733	2,124
BI_3	1,406	877	1,504	1,971	1,885	3,480
BI_4	1,543	2,418	817	942	3,006	2,101
BI_5	171	1,674	1,082	2,081	756	3,349
Sum	8,055	7,213	6,933	6,748	11,902	14,565.00

BI	N_1	N_2	N_3	N_4	N_5	N_6
1	0	0	0	0	0	0
2	2	2	2	2	2	2
3	4	4	4	4	4	4
4	6	6	6	6	6	6
5	8	8	8	8	8	8

Equivalent TI Distribution

BI	N_1	N_2	N_3	N_4	N_5	N_6
BI1	12	5	8	5	10	14
BI2	8	9	6	4	15	8
BI3	7	3	7	8	7	14
BI4	6	10	4	4	12	8
BI5	7	4	9	3	14	5

Packet Loss Results and Comparison with TIPS

	% Packet Drop (Total)					
Algorithm	N1	N2	N3	N4	N5	N6
DRR	18.6	7.86	4.44	3.79	2.25	1.4
FQ	49.93	24.42	5.07	14.08	6.46	12.66
RED	11.39	8.22	7.31	9.97	4.39	6.25
RR	10.57	8.29	7.94	15.27	2.57	3.73
SFQ	26.3	7.05	2.07	3.38	6.47	12.44
TIPS	0	0	0	0	0	0

Maxmimum Throughput

Algorithm	Percentage Throughput						
	N1	N2	N3	N4	N5	N6	Sum
DRR	86.84	96.05	98.42	99.6	99.78	99.9	97.3
FQ	65.14	86.3	97.17	92.16	99.77	99.9	88.4
RED	95.12	95.34	95.86	93.28	99.78	99.9	97.4
RR	95.82	95.12	95.49	90.51	99.77	99.91	97.1
SFQ	84.72	95.92	98.18	97.54	99.78	99.91	96.7
TIPS	100	100	100	100	100	100	100

Jitter Comparison

Packet Jitter (milliseconds)					
Maximum	**Method**	**Average**	**Method**	**Minimum**	**Method**
2.09	TIPS	0.35	TIPS	0.11	TIPS
3.73	DRR	0.4	DRR	0.11	DRR
2.74	FQ	0.42	FQ	0.11	FQ
3.01	RED	0.36	RED	0.16	RED
3.59	RR	0.42	RR	0.32	RR
3.77	SFQ	0.4	SFQ	0.31	SFQ