

# A Novel Scheme for Dynamic Triggering of Packet Dispersion

N. M. Ninkovic<sup>1</sup>, Z. C. Bojovic<sup>2</sup>, S. Gajin<sup>1</sup>

<sup>1</sup>*School of Electrical Engineering, University of Belgrade,  
Bulevar Kralja Aleksandra 73, Serbia*

<sup>2</sup>*Telekom Srbija AD,  
Takovska 2, Belgrade, Serbia  
nemanja.ninkovic@amres.ac.rs*

**Abstract**—Packet burst losses are proven to be detrimental to multimedia applications. Most of the burst losses in the Internet are the result of tail drop discipline, which drops packets when queues in routers are filled to their maximum capacity. A potential way of addressing the burst losses is to use packet dispersion over multiple and disjoint paths to destination. However, packet dispersion requires packet reordering and addressing the problems concerning packet latency and jitter. We are proposing a model that dynamically triggers packet dispersion depending on the high priority queue occupancy containing high priority traffic, whereas the need for packet dispersion and the number of dispersion paths is inferred from traffic load in other queues. Proposed mechanism reduces tail drops in queues containing high priority traffic by performing packet shifting to high priority queues that belong to other output interfaces designated for packet dispersion. It delivers significant benefits regarding packet loss and packet loss distance, which are considered imperative parameters for describing the quality of multimedia and real-time traffic.

**Index Terms**—Burst packet loss, noticeable packet loss, packet dispersion, queuing analysis.

## I. INTRODUCTION

High demand and popularity of video and voice applications on the Internet imposed the necessity for performance guarantees such as bandwidth, delay, jitter and packet loss ratio (PLR). The most widespread implementation of voice application is Voice over IP (VoIP), which requires priority processing of voice packets with low latency, jitter and PLR. According to [1] targeted one-way delay of voice packet is 150 ms and packets with higher delay result in diminished voice service quality. Similar assumptions may be extended to other multimedia, time-critical or mission-critical traffic. Therefore, networks with high volume of traffic must be engineered to address potential latency problems and to introduce service class priority.

Voice applications in addition to latency issues are sensitive to packet loss. One way of dealing with burst packet loss is to use packet dispersion over multiple paths from source to destination. Different dispersion policies may be employed to spread the distance between consecutive packet losses.

Combining the dynamic triggering of packet dispersion over available paths together with shifting of excess packets from high priority queue to other dispersion path queues, we aim to analyse the impact of proposed packet dispersion mechanism on packet loss pattern.

## II. PACKET DISPERSION AND LOSS PATTERN METRICS

Burst losses are often the result of tail drop queue disciplines in routers on the path. Such drops may lead to packet losses in considerable number of separate multimedia sessions, which because of UDP as transport protocol and strict latency requirements may not recover using packet retransmission. Since voice applications are equipped to deal with separate packet losses, total or average packet loss is not a practical way to evaluate the distribution and pattern of losses. Loss pattern metrics, defined in [2], take into account distance between packet losses. For multimedia applications, the most important loss pattern metric is Noticeable Loss Rate (NLR) calculated as ratio of lost packets with distance less than  $d_{\min}$  and total number of lost packets. For instance, any packet loss with distance less than  $d_{\min}$  may be noticed by the user as a gap in speech when voice application is considered.

Packet dispersion described in [3] and [4] assume the existence of two and three disjoint paths, respectively, from source to destination. On the contrary, we argue that they do not have to be completely disjoint from source to destination. Paths have to be only disjoint for segments containing the routers, which exercise severe tail drops and have inadequately sized buffers. For multimedia applications, quality of service (QoS) constraint routing should be implemented in order to fulfil required performance guarantees for delay, jitter, PLR and bandwidth.

Several dispersion policies are defined in [3]. For our proposed scheme, we are using the round robin dispersion policy. Round robin is considered deterministic dispersion policy as it is known which path packet in session will take next. Capability of packet dispersion to spread the packet loss distance is highly correlated with the number of available dispersion paths.

There are few caveats that must be addressed when using packet dispersion. Separate paths over which packets are

dispersed may have different latency, jitter or PLR, which could potentially lead to even worse performance comparing to the scenario without packet dispersion. Packet reordering may be required on the destination as a result of different latency of dispersed packets. Consequently, for multimedia applications, play-out buffer has to be used on the receiver's end in order to compensate for latency variation adding to end-to-end latency. Some dispersion policies as a result of uneven packet dispersion among the paths may cause increased load and performance problems on some of these paths.

Reflecting the packet dispersion weaknesses, we are proposing that packet dispersion is used only when it is necessary. Accordingly, a dynamic triggering of packet dispersion effectively avoids mentioned problems, which also utilizes the advantages of spreading the packet burst losses. Triggering mechanism defined in [4] may be used for activating packet dispersion for voice traffic and it is based on probe packets and measurement of packet loss. However, this kind of probing is often unreliable since it is necessary to determine the interval for these probe packets and the validity of the measurements. In our study, we are determining the state of the path taken by packets from high priority queue by monitoring its occupancy and activating packet dispersion, if necessary. Simultaneously, the number of dispersion paths is inferred from the occupancy in multiple high priority traffic queues, adding to our scheme's awareness regarding overall network load.

### III. PROPOSED PACKET DISPERSION DYNAMIC TRIGGERING

Our proposed model is based on packet shifting from one interface output queue to one or many highest priority queues that belong to outgoing interfaces for dispersion paths. These paths are determined to be eligible for packet dispersion to specific destination. As a part of our dispersion mechanism, we are introducing the ultra-high priority queue, queue with the highest priority, which is used only for the packet dispersion purpose.

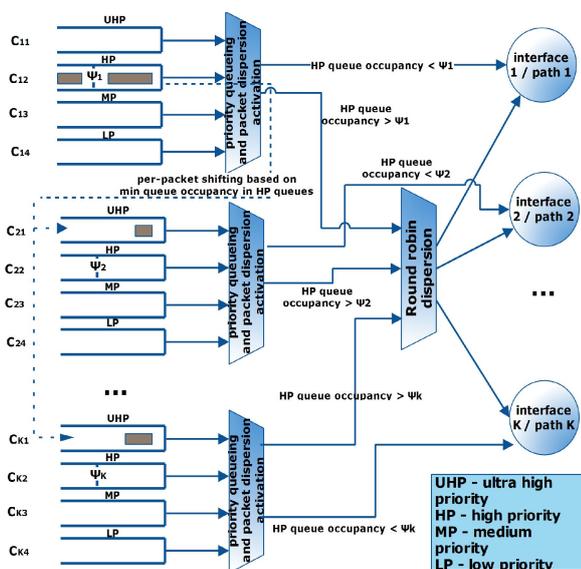


Fig. 1. High level overview of proposed model.

We assume a general model consisting of  $K$  paths to the destination (using  $K$  output interfaces). Each of these

interfaces has output queues classified according to priorities (ultra-high, high, medium and low) and for each of them, priority queuing is applied as a queue management mechanism. Proposed model is depicted in Fig. 1.

For the sake of this analysis, we are assuming that congestion happens only in high priority output queue of the *path 1*. Packets belonging to high priority traffic are placed in high priority queue. *Path 1* for packets to specific destination is considered to be the best path, although we assume there are  $K-1$  additional paths previously determined to be QoS capable for high priority traffic to specific destination. As long as the high priority queue occupancy does not reach threshold  $\psi_1$ , packets are serviced by the output *interface 1* and are sent across *path 1*. This is normal behaviour under the absence of congestion.

Once the high priority queue occupancy crosses threshold  $\psi_1$ , packets that are at the head of the line of the high priority queue are shifted to ultra-high priority queues belonging to the output interfaces determined as feasible QoS paths to desired destination. Per-packet decision is made to shift packet as opposed to flow shifting, which may cause flow disruption and affect performance [5]. Shifted packets are placed in one of the  $K-1$  additional ultra-high priority queues belonging to output interface whose high priority queue currently has minimal queue occupancy. This shifting is achieved at the cost of increasing implementation complexity of interface output queues where the shifting control is placed before the service element i.e. the output interface. Similar to [6], extensions can be made to perform monitoring and managing queue occupancy and availability. According to the dynamical change of the high priority queue occupancy, multiple ultra-high priority queues may accept shifted packets from a single high priority queue. Specific dispersion path is used for packet dispersion as long as there are shifted packets in its ultra-high priority queue, hence multiple paths may be used to reach destination.

Congestion and potential packet drop in high priority queues are reduced as a result of this packet shifting. When packet shift occurs, packet dispersion is activated and packets are no longer sent directly to output interface, but they are dispersed over multiple paths whose number depends on the number of different ultra-high priority queues occupied by these shifted packets. Priority queue management is used according to which ultra-high priority queues are serviced before the rest of the output queues belonging to a single interface. When there are no more packets in ultra-high priority queue, that interface, i.e. path, is taken out from being available for dispersion. A change in minimal high priority queue occupancy turns previously empty ultra-high priority queue to accept packets, thus activating an additional interface for use in packet dispersion process. The proposed model uses round robin dispersion policy. Packets in high priority queue that started packet shifting and ultra-high priority queues (containing only shifted packets) are serviced in a round robin fashion using currently activated output interfaces. Round robin dispersion enables dispersing packets belonging to a single session across separate paths, thus increasing the distance between lost packets in one session.

Since ultra-high priority queues are limited in capacity (in

order to prevent starvation of respective high priority queues), it is possible that there is no more available capacity for packet shifting. In that case, packets are filling the high priority queue above threshold  $\tau_j$  until the high priority queue capacity is reached. Afterwards, unless arrival rate is reduced or some of the ultra-high queues become available, packet drops will start. Probability for burst losses is lower if there are more ultra-high queues available, i.e. more paths for packet dispersion are available. Adaptable and dynamic triggering of the packet dispersion helps to deal with congestion conditions of the high priority traffic and, simultaneously, counter severe packet burst losses on the path to destination. Packet dispersion triggering based on queue occupancy results in minimizing native packet dispersion problems.

We must point out that careful queue capacity [7]–[10] and packet dispersion threshold selection must be made. As it was previously stated, it is not recommended to trigger packet dispersion frequently. Moreover, consideration is required to avoid potential tail drop due to the queue capacity. Generally, if a queue has higher priority, it should have less capacity and vice versa. Therefore, shifting the packets to ultra-high priority queues does not add excessive latency.

#### IV. PERFORMANCE ANALYSIS

In order to evaluate performance and perform numerical analysis, we apply the queuing theory. Each queue in our model before packet dispersion activation can be considered as finite M/M/1/C queue with capacity of  $C$ , Poisson arrival, exponential service time and first come first served (FCFS) discipline. Queue capacity is denoted as  $C_{ij}$ , where index  $i$  denotes output interface to which specific queue belongs and index  $j$  determines queue priority ( $j = 1$  is ultra-high,  $j = 2$  is high, etc.). We consider high priority queue belonging to *interface 1*, i.e. *path 1*. For the subsequent analysis, we assume that only *interface 1* high priority queue reaches packet dispersion threshold, while other high priority queues do not reach their threshold. Further, assuming that  $\lambda$  is an arrival rate in high priority queue and  $\mu$  is a service rate, traffic intensity is  $\rho = \lambda/\mu$ .

Before the *interface 1* high priority queue occupancy reaches threshold  $\tau$ , this queue may be observed as finite M/M/1/ queue with capacity  $C$ . According to the queuing theory, probability that high-priority queue occupancy will reach its threshold  $\tau$  is

$$P_{\tau} = \rho^{\tau} \cdot \frac{1 - \rho}{1 - \rho^{\tau+1}}, \quad (1)$$

and probability that high priority queue is empty is

$$P_0 = \frac{1 - \rho}{1 - \rho^{\tau+1}}. \quad (2)$$

Expected number of packets in the system (including serviced packets) and packets in queue is provided respectively:

$$L_{s,\tau} = \sum_{n=1}^{\tau} n P_n = \frac{\rho \dots (1 - \rho^{\tau}) (\tau + 1) + \rho^{\tau+1}}{(1 - \rho) (1 - \rho^{\tau+1})}, \quad (3)$$

$$L_{q,\tau} = \sum_{n=1}^{\tau} (n - 1) \cdot P_n = L_{s,\tau} - (1 - P_0). \quad (4)$$

Average packet waiting time in high priority queue before activation of packet dispersion equals

$$W_{q,\tau} = \frac{L_{q,\tau}}{\lambda_{\tau}} = \frac{L_{q,\tau}}{(1 - P_{\tau})}, \quad (5)$$

where  $\lambda_{\tau}$  is considered to be effective arrival rate before packet dispersion activation or more accurately, before reaching threshold  $\tau$ .

When queue occupancy in high priority queue reaches threshold  $\tau$ , arrival rate in high priority queue is changed and packets are shifted from the head of the line to ultra-high priority queue with minimal high priority queue occupancy. Once the packet dispersion is activated, round robin scheduling of involved queues (high priority that activated it and ultra-high priority queues containing shifted packets) is performed. We are modelling round robin scheduling as M/G/1/C, which is serviced using processor sharing (PS) discipline [11]. Probability mass function of M/M/1/C and M/G/1/C/PS is the same. Consequently, mean delay is calculated as in (5), with different service rate since it has general distribution in M/G/1/C/PS queues. This equivalence was effectively employed in web server modelling [12] and performance analysis of the PC-based router [13]. Service rate  $\mu_{dispersion}$  now depends on the packet size distribution and the capacity of the interface servicing it. Since interfaces may have different capacities and subsequent packets from one queue may be serviced by interfaces with different capacities, average capacity is taken into account, so the service rate once packet dispersion is activated is

$$\mu_{dispersion} = \frac{1}{\frac{1}{K} \sum_{k=1}^K \frac{L_k}{B}}, \quad (6)$$

where  $L_k$  is packet size, which has general distribution and  $\bar{B} = \frac{1}{K} \sum_{i=1}^K B_i$  is the average bandwidth capacity of interfaces used for dispersion. This approximation is appropriate to model the round robin scheduling when queue servicing is performed by interfaces with similar capacities.

Packet arrival rates before packet dispersion activation are depicted in Fig. 2(a). If high priority queue occupancy crosses threshold  $\tau$ , packets will be shifted to ultra-high priority queues with the rate of  $P$  and packet arrival to separate ultra-high priority queues will depend on queue occupancy of respective high priority queues. Therefore, arrival of shifted packets to ultra-high priority queue is

$$\lambda_{i1} = \lambda_{\tau} P_{i1}, \quad (7)$$

where  $\lambda_{i1}$  is an arrival rate in ultra-high priority queue

whereas  $P_{i1}$  is the probability that specific high priority queue has minimal queue occupancy comparing to the other high priority queue occupancies belonging to interfaces eligible for packet dispersion. This scenario is depicted in Fig. 2(b).

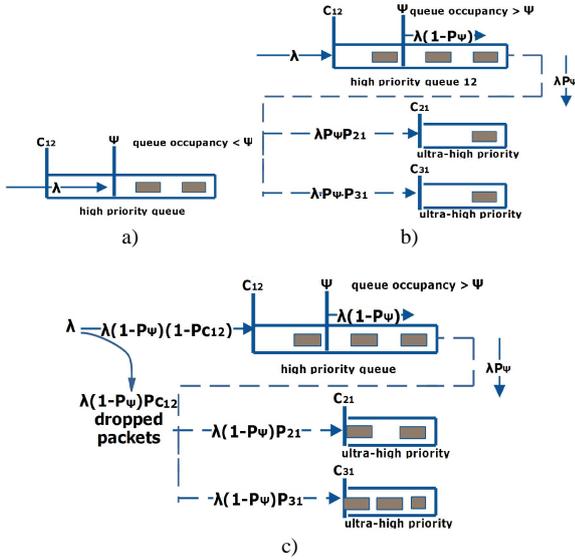


Fig. 2. Packet arrival rates in respective queues: (a) high priority queue occupancy  $< \Psi$ ; (b) high priority queue occupancy  $> \Psi$ ; (c) high priority queue occupancy  $> \Psi$  and packets dropping.

Average waiting time of the shifted packets consists of two components, the average time spent in high priority queue before shifting and the average time spent in ultra-high priority queue before servicing. Traffic intensity in high priority queue that activated packet dispersion is  $i_{i2} = \lambda / \mu_{dispersion}$  and traffic intensity in ultra-high priority  $i_{i1} = \lambda / \mu_{dispersion}$ . Therefore, the average waiting time of the packets that remained to be serviced from high priority queue  $W_{q,C_{i2},dispersion}$  and shifted packets  $W_{q,C_{i1},shifted}$  is respectively provided in the following:

$$W_{q,C_{i2},dispersion} = \frac{L_{q,C_{i2},dispersion}}{\lambda_{\text{eff}}(1-P_{C_{i2}})}, \quad (8)$$

$$W_{q,C_{i1},shifted} = \frac{L_{q,C_{i2},dispersion}}{\lambda_{\text{eff}}(1-P_{C_{i2}})} + \frac{L_{q,C_{i1},shifted}}{\lambda_{i1}(1-P_{C_{i1}})}, \quad (9)$$

where  $L_{q,C_{i2},dispersion}$  and  $L_{q,C_{i1},shifted}$  are queue lengths as described in (4) when traffic intensity is  $i_{i2}$  and  $i_{i1}$ , respectively, whereas  $P_{C_{i1}}$  and  $P_{C_{i2}}$  is probability that ultra-high and high priority queue, respectively, has reached its capacity. Our scheme prevents packet shifting to ultra-high priority queues with filled capacities.

Packet dropping in the high priority queue may happen if all ultra-high priority queues are filled and packet shifting is not possible. In that case, high priority queue is filled to its capacity after which packet dropping occurs (Fig. 2(c)). Probability that high priority queue will start dropping packets is

$$P_{C_{i2}} = \dots_{i2} C_{i2} \times \frac{1 - \dots_{i2}}{1 - \dots_{i2}^{C_{i2}+1}}, \quad (10)$$

where  $C_{i2}$  is the capacity of high priority queue that started

dispersion. Additionally, the dropping rate  $d$  is

$$d = \lambda_{\text{eff}} \cdot P_{C_{i2}} = \lambda \cdot (1 - P_{\text{eff}}) \cdot P_{C_{i2}}. \quad (11)$$

Previous analysis assumes that only one high priority queue activates packet dispersion and considering that our model implements priority queue management, high priority queue may be pre-empted by ultra-high priority queue. Consequently, high priority queue service rate when other high priority queues are able to activate packet dispersion is

$$\lambda_{\text{eff}} = \begin{cases} \lambda \cdot P_{UHP0}, & \text{queue occupancy} < \Psi, \\ \lambda \cdot dispersion \cdot P_{UHP0}, & \text{queue occupancy} > \Psi, \end{cases} \quad (12)$$

where  $P_{UHP0}$  is the probability that respective ultra-high priority queue is empty.

As long as there is packet in multiple ultra-high priority queues, more paths will be used and packet dispersion will counter burst losses more effectively. However, packet latency and jitter during this process must be minimized. In order to prevent queue starvation due to the priority queuing management, ultra-high priority queues are small-sized, hence the additional latency and jitter between packets that are shifted and the ones that are not is the result of time spent in ultra-high priority queue. Further analysis shows our model's influence on severe packet losses on the path.

## V. MODELLING BURST LOSSES USING GILBERT MODEL

Packet burst losses, as a result of tail drop queue discipline, are modelled using the Gilbert model. According to it, path is considered to be in “good” state, when there is low probability of packet loss, or “bad”, when there is high probability of packet loss.

Two-state Gilbert model is specified using Markov chain that consists of two states with different probabilities of packet loss. Transition between the “good” and “bad” state and vice versa, is described with probabilities  $p$  and  $q$ , respectively, whereas  $P_g$  and  $P_b$  is used to describe packet loss in respective states (Fig. 3).

Transitions between the two states are provided using the transition matrix given in the following

$$A = \begin{bmatrix} 1-p & p \\ q & 1-q \end{bmatrix}, \quad (13)$$

whereas error matrix provides packet loss probabilities in these two states

$$B = \begin{bmatrix} P_g & P_b \end{bmatrix}. \quad (14)$$

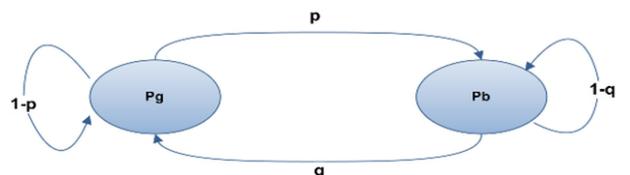


Fig. 3. Gilbert two-state model.

In order to model the burst losses using Gilbert model,

Markov chain, described using transition matrix in (13), has to be irreducible and regular. For regular Markov chain, steady state probabilities are:

$$\begin{cases} f_g = \frac{q}{p+q}, \\ f_b = \frac{p}{p+q}, \end{cases} \quad (15)$$

where  $f_g$  and  $f_b$  are steady state probabilities for “good” and “bad” state, respectively. Since we are using Gilbert model to perform packet loss modelling, steady state probabilities refer to the probability of separate and burst packet loss.

PLR generated using Gilbert model with mentioned parameters is provided as

$$P_l = f_g \cdot P_g + f_b \cdot P_b. \quad (16)$$

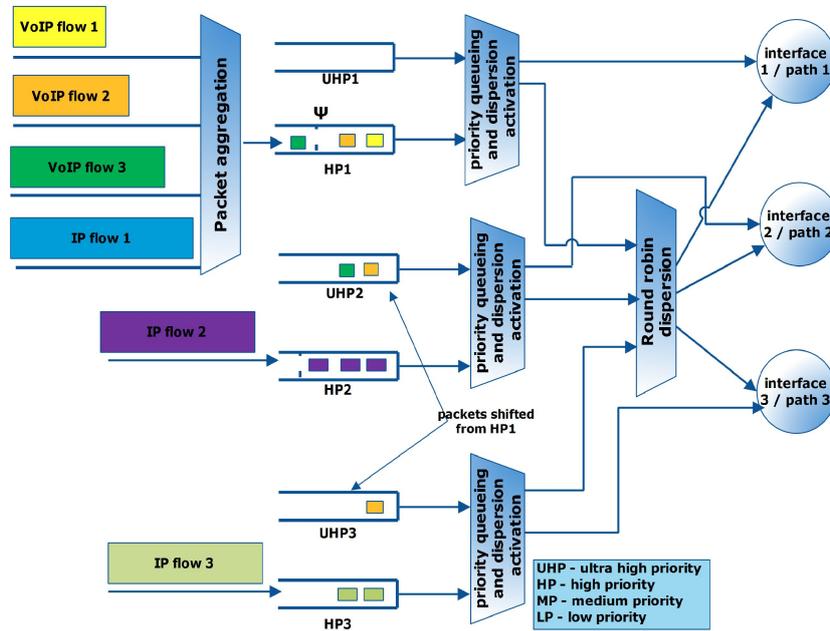


Fig. 4. High level overview of simulation framework.

High priority traffic consists of three G.711 voice sessions for packetization interval of 20 ms. *VoIP flow 1* session lasts for first 20 seconds, *VoIP flow 2* lasts for the first 30 seconds and the *VoIP flow 3* lasts for 50 seconds. These three VoIP flows are sent to the first HP queue. High priority traffic serviced by other two interfaces consists of two IP flows that have same packet length as voice traffic with exponential intergeneration time. These IP flows are meant to model the load in other two HP queues and to cause queue occupancy change, so the packet dispersion would activate different paths according to our model. *IP flow 1* is used to generate HP queue occupancy above packet dispersion threshold. Same packet length was chosen, so that it would be appropriate to express queue occupancy in number of packets. During the course of this analysis, we let only *HP1* queue to activate packet dispersion whereas the other two HP queues do not reach packet dispersion threshold.

In this simulation, we are assuming that *path 1* is congested, which is reasonable to assume since there will be higher occupancy in *HP1* queue. Therefore, we are using

It is argued in [14] that more accurate modelling is achieved using the extended Gilbert model containing higher order Markov chain adding to computational complexity. However, for our study, specification of burst losses on the paths is not available and we are observing the worst case scenario using two-state Gilbert model, which is able to accurately show the effectiveness of our proposed model.

## VI. SIMULATION AND RESULTS

Simulation and testing environment were developed using the extensive MATLAB model (Fig. 4). It consists of three output interfaces (three paths) each containing ultra-high priority (UHP) and high priority (HP) queues. During the course of simulation, queue capacity  $C_{12}$  and threshold of the *HP1* queue are changed to analyse their effect on packet dispersion triggering and consequently, packet losses experienced on the path to the destination.

Gilbert loss model to account for any voice packet tail drops on *path 1* to the destination. Burst loss scenario that we are applying in this simulation is extreme and can be considered worst case scenario since the *path 1* with this kind of burst packet loss is not usable for voice application. Parameters for burst losses are provided in (17) and Table I:

$$\begin{cases} A = \begin{bmatrix} 0.95 & 0.05 \\ 0.45 & 0.55 \end{bmatrix}, \\ B = \begin{bmatrix} 0.001 & 0.8 \end{bmatrix}. \end{cases} \quad (17)$$

TABLE I. BURST LOSS PARAMETERS.

$g$	0.9
$b$	0.1
$P_l$	0.0809

Capacity of *HP2* and *HP3* queues is  $C_{22} = 300$  packets and  $C_{32} = 600$  packets, respectively. We change the values of threshold, which affects the triggering of the packet dispersion and consequently, burst losses occurring on *path*

1. PLR and NLR are evaluated and compared to the case when there is no packet dispersion implemented. Since we are performing tests on voice applications, which have Packet Loss Concealment (PLC) implemented, every packet loss with loss distance less than three packets is considered noticeable. Therefore, NLR is calculated for  $\min \geq 3$ .

We performed two experiments for this study. First experiment encompasses the change of queue occupancy threshold of HP1 queue whose capacity  $C_{12}$  is 500 packets, whereas in second experiment  $C_{12}$  is 300 packets. In Fig. 5–Fig. 8 we provide the PLR and NLR values, when our model is put to use, which are normalized to the PLR and NLR values without proposed model (no dispersion whatsoever). Threshold in Fig. 5–Fig. 8 is expressed as a percentage of HP1 queue capacity.

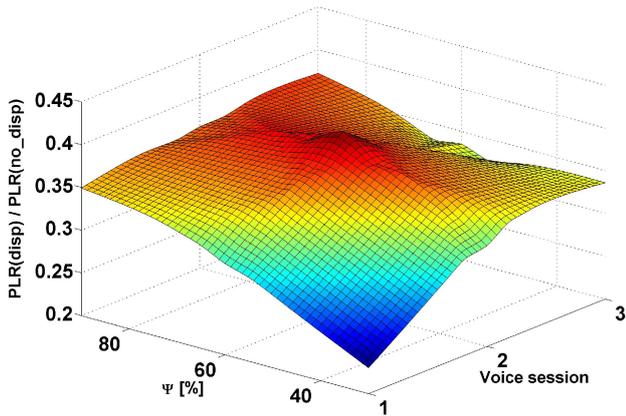


Fig. 5. Normalized PLR ( $C_{12} = 500$  packets).

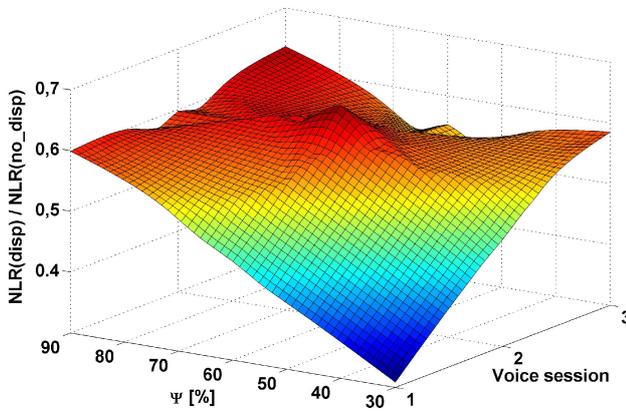


Fig. 6. Normalized NLR ( $C_{12} = 500$  packets).

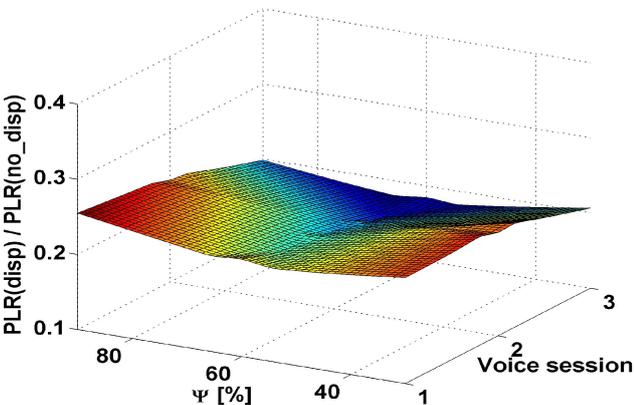


Fig. 7. Normalized PLR ( $C_{12} = 300$  packets).

Second experiment, when  $C_{12} = 300$  packets, provides lower packet losses and NLR. This is due to the fact that

packet dispersion was triggered faster and more packets were subjected to packet dispersion over multiple paths. In the first experiment (when  $C_{12} = 500$  packets and  $\Psi = 300$  packets) 373 packets were dropped for all VoIP sessions, whereas in the second experiment (when  $C_{12} = 300$  packets and  $\Psi = 180$  packets) 268 packets were dropped as a result of burst losses on the *path 1*. The main reason for lower packet loss and NLR in the second experiment is lower queue capacity threshold. Based on the previous experiments, we can deduce that for low capacity queues, application of our model provides superior performance comparing to the scenario without packet dispersion. On the other hand, first experiment results in less probability of dropped packets in HP1 queue as a result of higher available queue capacity.

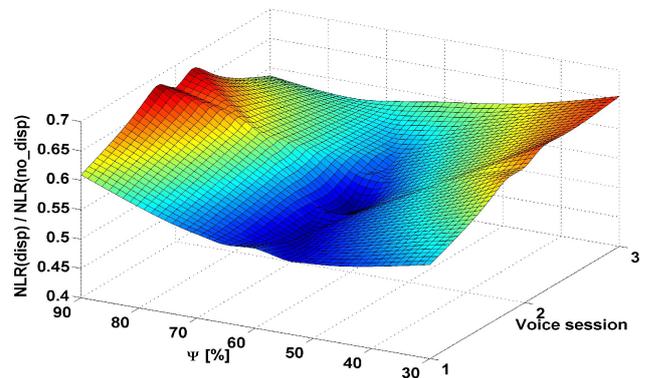


Fig. 8. Normalized NLR ( $C_{12} = 300$  packets).

Observing the PLR and NLR parameters, it can be noted that for both experiments, packet losses are significantly lower in comparison to the scenario without packet dispersion. Generally, it is a good practice to trigger packet dispersion earlier, for lower queue occupancy, so that fewer packets take the path with burst losses. However, packet dispersion does not always include all available paths and may, at some point, affect performance of traffic in other HP queues belonging to other output interfaces. We recommend that the threshold, which triggers packet dispersion, is chosen so that packet dispersion is not activated frequently, but only in cases leading to serious packet drop.

Packet dispersion triggering and the number of paths in our model are highly correlated with HP queue occupancy. Queue occupancy in HP2 and HP3 is depicted in Fig. 9(b) and Fig. 9(c), respectively. HP1 queue occupancy in first and second experiment is depicted in Fig. 9(a) and Fig. 9(d), respectively.

Frequent change of the HP queue with minimal occupancy will activate more paths for packet dispersion. The number of activated dispersion paths is depicted in Fig. 10 for different queue capacities and dispersion thresholds.

Most of the packet dispersion during the simulation is performed over two paths whereas in less frequent cases, all three dispersion paths are active. Simultaneous use of three paths would be more frequent if shifted packets spent more time in UHP queues, which have small capacity and occupancy (UHP queue capacity for both experiments is 20 packets). Consequently, shifted packets are serviced quickly after which specific path is taken out from dispersion

process. Packet distribution of shifted packets to UHP queues depends on queue occupancy in HP queues (Fig. 9(b) and Fig. 9(c)). This fact shows the adaptability of our model to overall traffic load in other queues and prevents

significant performance degradation of other traffic flows. Hence, our model adapts to network state and packet load in output queues.

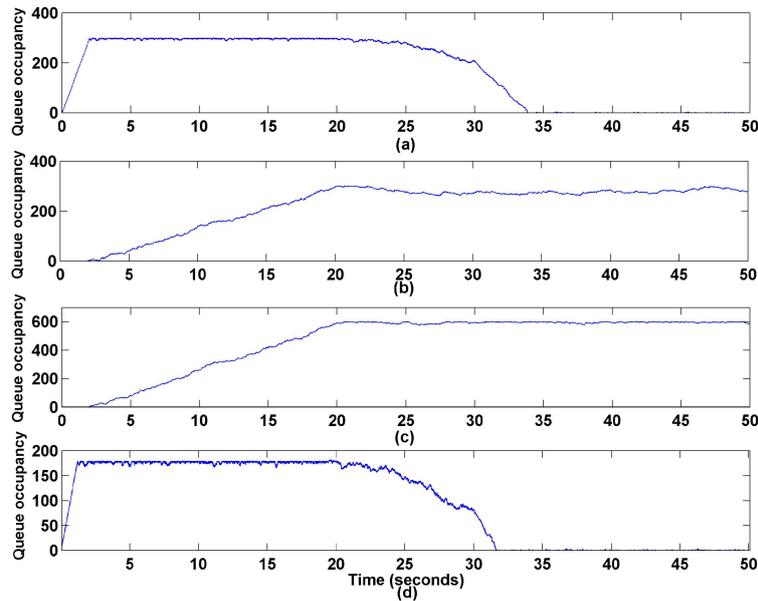


Fig. 9. Number of packets in queues: (a) *HP1* ( $C_{12} = 500$  packets,  $C_{22} = 300$  packets); (b) *HP2* ( $C_{22} = 300$  packets); (c) *HP3* ( $C_{32} = 600$  packets); (d) *HP1* ( $C_{12} = 300$  packets,  $C_{22} = 180$  packets).

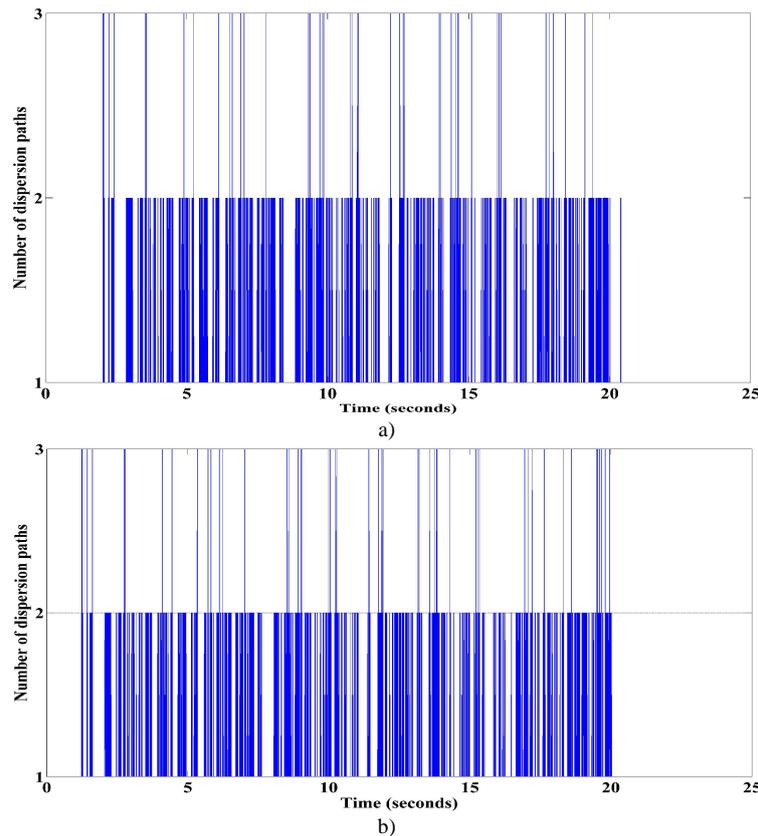


Fig. 10. Dispersion path triggering: (a) *HP1* ( $C_{12} = 500$  packets,  $C_{22} = 300$  packets); (b) *HP1* ( $C_{12} = 300$  packets,  $C_{22} = 180$  packets).

As jitter presents one of the main difficulties that needs to be addressed when implementing packet dispersion, it is important to consider the way our packet dispersion scheme is affected by it. Shifted packets to UHP queues may have different latencies comparing to packets that remained in HP queues. Therefore, difference in latency of these packets, i.e. jitter, is actually the average time that shifted packets spend

in UHP queues. More accurately, second part of the (9) presents the jitter value.

Impact of active dispersion paths and the UHP queue sizes on jitter is shown in Fig. 11. Assuming that the probability of minimal queue occupancies is equal for every HP queue of each available output interface (equal interface capacities), the increase of activated dispersion paths

number leads to smaller packet jitter as there are fewer packets in UHP queues (Fig. 11(a)).

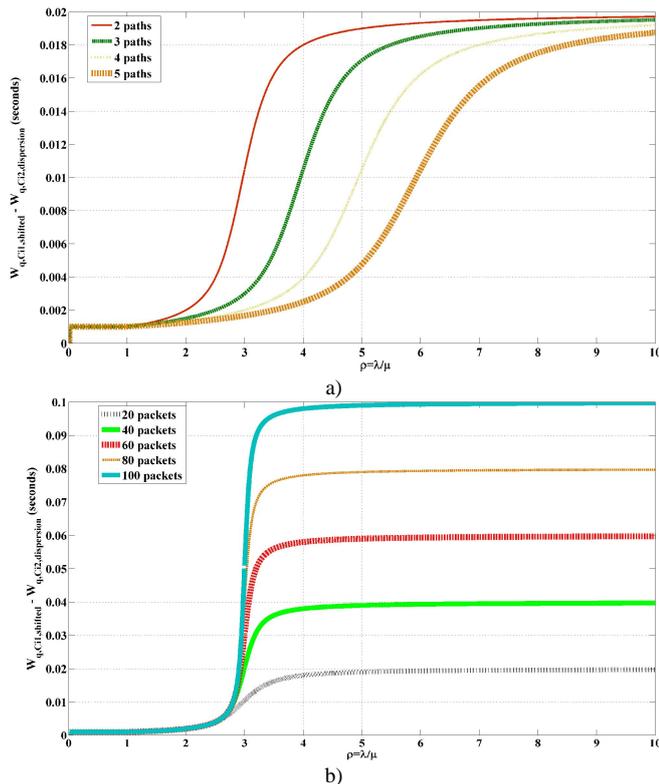


Fig. 11. Packet shifting jitter: (a) UHP capacity = 20 packets,  $\rho = 180$ ; (b) 2 dispersion paths activated,  $\rho = 180$ .

Therefore, our proposal does not add excessive jitter when number of dispersion paths is increased, which further addresses packet dispersion problem previously explained. Additionally, as the size of the UHP queue grows, jitter increases since the average time spent in UHP queue is now longer (Fig. 11(b)). Therefore, trade-off is required when attempting to optimize jitter value with our packet dispersion mechanism applied. We generally recommend that UHP queues should be small-sized since the large queue capacities may result in higher packet latency and jitter.

## VII. CONCLUSIONS

The proposed packet dispersion scheme successfully copes with packet loss and tail drops. We have shown that using our model that combines the dynamically triggered packet dispersion over variable number of paths and packet

shifting from higher priority to lower priority queues, it is possible to significantly reduce packet losses and to increase the distance between them. Further flexibility of this model may be achieved in the capability of classifying the traffic we wish to apply packet dispersion to. It is particularly applicable for addressing high packet loss and burst losses as a result of high traffic loads, inconsistent QoS mechanisms or bad network engineering.

## REFERENCES

- [1] ITU-T Recommendation G.114, *One-way transmission time*. 2003.
- [2] R Koodli, R Ravikanth, *One-way loss pattern sample metrics*, RFC 3357, 2002.
- [3] H. Levy, H. Zlatokrilov, "The effect of packet dispersion on voice applications in IP Networks", *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 277–288, 2006. [Online]. Available: <http://dx.doi.org/10.1109/TNET.2006.872543>
- [4] J. F. Banu, V. Ramachandran, "Multipath adaptive packet dispersion for voice applications", *Journal of Computer Science*, vol. 8, no. 4, pp. 454–459, 2012. [Online]. Available: <http://dx.doi.org/10.3844/jcssp.2012.454.459>
- [5] W. Shi, M. H. MacGrefor, P. Gburzynski, "Load balancing for parallel forwarding", *IEEE/ACM Trans. on Networking*, vol. 13, no. 4, pp. 790–801, 2005. [Online]. Available: <http://dx.doi.org/10.1109/TNET.2005.852881>
- [6] G. Cincikas, S. Japertas, "Router resource management research", *Elektronika ir Elektrotechnika*, no. 6, 2012. [Online]. Available: <http://dx.doi.org/10.5755/j01.eee.122.6.1825>
- [7] A. Vishwanath, V. Sivaraman, M. Thottan, "Perspectives on router buffer sizing", *SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 2, pp. 34–39, 2009. [Online]. Available: <http://dx.doi.org/10.1145/1517480.1517487>
- [8] L. Sequeira, J. Fernandez-Navajas, J. Saldana, L. Casadesus, "Empirically characterizing the buffer behaviour of real devices", *Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2012, pp. 1–6.
- [9] Y. Zang, J. YanTuning, "Self-similar traffic to improve loss performance in small buffer routers", in *Proc. the Tenth Int. Conf. Networks*, 2011.
- [10] Y. S. Hanay, T. Wolf, "Practical packet pacing in small-buffer networks", in *Proc. IEEE Int. Conf. Communications (ICC 2009)*, 2009, pp. 1–6.
- [11] A. R. Ward, W. Whitt, "Predicting response times in processor-sharing queues", in *Proc. of the Fields Institute Conf. on Comm. Networks*, 2000, pp. 1–29.
- [12] J. Cao, M. Andersson, C. Nyberg, M. Kihl, "Web server performance modeling using an M/G/1/K\*PS queue", in *Proc. of the 10th Int. Conf. on Telecomm. (ICT)*, 2003, pp. 1501–1506.
- [13] S. Seo, I. Kong, "A performance analysis model of PC-based software router supporting IPv6-IPv4 translation for residential gateway", in *Proc. of the Fourth Annual ACIS Int. Conf. on Comp. and Inf. Sci.*, 2005, pp. 144–150.
- [14] Y. Xunqi, J. W. Modestino, T. Xusheng, "The accuracy of Gilbert models in predicting packet-loss statistics for a single-multiplexer network model", *INFOCOM 2005, 24th Proc. IEEE*, 2005, vol. 4, pp. 2602–2612.