ConEx Based QoE Feedback to Enhance QoS

Meral Shirazipour
Ericsson Research
Packet Technologies
Montreal, Canada

Gregory Charlot
Ecole Polytechnique
Department of Computer and Software Engineering
Montreal, Canada

Geoffrey Lefebvre
Ericsson Research
Packet Technologies
Montreal, Canada

Suresh Krishnan
Ericsson Research
Packet Technologies
Montreal, Canada

Samuel Pierre
Ecole Polytechnique
Department of Computer and Software Engineering
Montreal, Canada

ABSTRACT

Quality of service (QoS) generally represents the performance of packet networks. Quality of experience (QoE) defines the quality perceived by end-users of applications running on these networks. This paper relates these two metrics in a novel way using the newly defined congestion exposure (ConEx) mechanism. ConEx is an experimental protocol defined by the IETF that allows the sender of a flow to convey the received explicit congestion notification (ECN) information back into the network. In IPv6, ConEx is implemented in an option header with 28 unused bits. These bits can be used to convey more than ECN feedback towards the network. This paper proposes to use these bits to send real-time objective QoE information, as perceived by the end-users, into the network. Routers can leverage this information to adjust QoS mechanisms. As an example, a new queue management technique is proposed with a multi-field DiffServ classifier using the QoE metric. Simulation results show that this mechanism can help in improving the overall QoE of active flows.

Categories and Subject Descriptors

C.2 [**COMPUTER-COMMUNICATION NETWORKS**]: Miscellaneous; C.2.2 [**Network Protocols**]: Subjects—*Protocol architecture*

General Terms

Algorithms, Measurement, Performance

Keywords

ConEx, QoE, QoS, AQM

1. INTRODUCTION

Quality of service (QoS) parameters give an objective measure of the quality of the network's service, which consist of the transport of Internet protocol (IP) packets. Quality of experience (QoE) is the measure of how satisfied the end-user is of a communication service. QoE is greatly impacted by the quality of the network's service, i.e., by QoS parameters. Historically, for circuit based voice services, QoE was subjectively evaluated by a factor called the mean opinion score (MOS). The MOS factor has a value between 1 (very poor) to 5 (excellent) and was used to rate the connection quality of a telephone line. Nowadays, standard objective methods exist to relate the QoS performance of the network and estimate the equivalent MOS. These QoE computation modules often need information at the application level and need to be implemented in end-systems, however without too much loss of precision they can also be implemented in external packet processors that can monitor application flows and compute QoE information. Network based packet processors can also alleviate the imminent problem of untruthful hosts.

Notwithstanding, assuming the QoE information is available, the question raised and answered by this paper is how to dynamically relate the QoE information to QoS mechanisms in the network with the goal of fine tuning and adjusting the QoS in response to QoE degradation. The main problem for such mechanism consists of communicating the QoE information back towards the network without causing scalability and synchronization issues. This problem no longer holds as a new protocol has been proposed by the Internet engineering task force (IETF) for the congestion exposure (ConEx) mechanism. With this protocol, information can be sent towards the network in-band, i.e., on transmitted packets. This paper focuses on evaluating the feasibility of such technique. The deployment concerns for such mechanism to be realistically deployed are raised in Section 6 and are left for future studies.

This paper is structured as follows: Section 2 presents some background information on QoS, QoE and ConEx. It also presents some related work that attempt to use QoE information for network QoS adjustment, but it is arguable that these existing proposals are non-viable because routers

Table 1: E-Model's R-Factor to MOS mapping

User Perception	R-Factor	Equivalent MOS
Very Satisfied	90-100 (G.711:max 93)	4.3-5.0 (G.711:max 4.4)
Satisfied	80-90	4.0-4.3
Some users satisfied	70-80	3.6-4.0
Many users dissatisfied	60-70	3.1-3.6
Almost all users dissatisfied	50-60	2.6-3.1
Not Recommended	<50	1.0-2.6

would need to keep per-flow state information, as they usually require an out-of-band signalling protocol to convey the QoE information. Section 3 presents various QoE information formats that can be fed back towards the network. Section 4 proposes the manner ConEx can be used to convey the QoE information. Section 5 presents a differentiated services (DiffServ [4]) based queue management example, as the QoS mechanism that can be enhanced with QoE feedback information, followed with a few simulation results obtained on ns-3 [2]. Finally, Section 6 concludes and presents ongoing and future works.

2. BACKGROUND AND RELATED WORK

In this section, the concepts of QoS and QoE are reviewed with a discussion on their relation. Then previous work which have attempted to relate these two parameters are presented leaving open the question of defining a mechanism that allows the QoS= function(QoE) paradigm without an extra burden on the network and the need for extra signalling protocols.

2.1 QoS and QoE overview

QoS is usually measured by five parameters: bandwidth, delay, jitter, packet loss and availability. Depending on the application, a set of these parameters will have stringent threshold values that need to be met in order to ensure end-user acceptance.

With the widespread of voice over IP (VoIP) and other IP based multimedia applications, the same service qualification as with the MOS factor needed to be measured. For this reason, ITU-T standard G. 107 proposes the E-Model as a way to estimate an R-Factor for VoIP calls [8]. There is also a direct mapping of R-Factor to MOS as shown in Table 1. The E-Model is therefore a standard method to measure the quality of a VoIP call perceived by the end-user. The measure is an estimate based on various system level factors as well as network factors including delay, packet loss, etc.

Similar to VoIP applications, objective QoE evaluation techniques have been proposed for video and web browsing applications. This paper uses the E-Model in the discussed examples, however depending on the application, the corresponding QoE evaluation technique should be used.

In an attempt to enhance QoE, network operators usually concentrate on QoS, which is easier to monitor and control. Typical QoS enhancing mechanisms are either traffic conditioning or related to traffic queuing and scheduling. Traffic conditioning covers metering, classification, marking, shaping and policing. Queuing and scheduling algorithms cover round-robin, first in first out, priority queuing, etc.

Classification is often performed using DiffServ mechanisms [4] at the edge of the network, followed by a differentiated service per hop behavior (PHB) in core network routers. All these mechanisms are implemented to intervene for or avoid network congestion. While over-provisioning may seem like a good solution to alleviate congestion, it is not always possible and does not produce guaranteed results. Furthermore, sometimes physical resources are scarce (e.g. radio resources) and prevent over-provisioning.

2.2 Related work

Reference [3] proposes an adaptive routing algorithm using the QoE of voice calls obtained with a pseudo subjective quality assessment algorithm. The main contribution is a new routing algorithm, with the abstraction of how to convey the QoE information in a viable way. Then, Ref. [19] proposes a method for QoS control where QoE feedback is used to vary the codec's bitrate in a voice call. This work was further improved by [13]. Here the QoE information is used by end-systems and not by the network. In Ref. [9], another new voice and video codec adaptation technique is proposed based on QoE feedback. The mechanism is new with QoE impacting both sender and receiver, affecting the bandwidth used and the jitter buffer respectively. Here, QoE information is also only used by end-systems.

The concept of QoE feedback is actually possible via the control protocol of the real-time transport protocol (RTP) flows. RTP control protocol (RTCP) is an application layer protocol which controls the RTP stream [17]. It periodically transmits information about the characteristics of an RTP session and the QoS. For each RTP session there is an RTCP session. Using RTCP extended reports (RTCP XR), the protocol can also be used to feedback the QoE as estimated by the R-Factor [7]. RTCP XR based QoE feedback is also only usable by end-systems, for codec adjustment purposes, etc.

Despite all the work around QoS and traffic engineering mechanisms, there has been no way to directly relate and operate these measures together using the monitoring of QoE to readjust the QoS in individual nodes in the network. The main reason why this has not yet been achieved is the difficulty of implementation of such mechanism. Using out-of-band signalling brings scalability and synchronization challenges on existing signalling networks or on a newly proposed signalling network. Moreover, existing QoE feedback solutions are at layers above the IP layer and thus mostly usable by end-nodes.

Designing a new IP layer protocol for QoE feedback is another possibility but the work and consensus gathering process for such effort has prevented this approach. However, with the recently proposed ConEx protocol, which is intended for explicit congestion notification (ECN) re-injection in the network, QoE feedback with in-band signalling becomes a possibility. This is the method exploited by this article. Before describing the proposed method, Section 2.3 below gives a brief overview of the ConEx protocol.

2.3 ConEx overview

ConEx [1, 12] is an IETF defined experimental protocol to allow a sender to inform the network about congestion encountered by previous packets of the same flow. The protocol is currently defined for the transmission control protocol (TCP) and ECN feedback; it consists in re-sending the ECN information received via subsequent packets being sent. The TCP based ConEx protocol is called Re-ECN [5]. Re-ECN for IPv6 has been defined by [10].

At the basis of ConEx, ECN [15], is a mechanism proposed beforehand by the IETF to inform network senders of congestion along the network path between the sender and receiver. Routers experiencing congestion mark packets selectively instead of dropping them, using known queuing techniques like random early detection (RED). Upon reception, the receiver witnessing the marking, relays this information back to the sender using a transport protocol specific mechanism like TCP [15] . ECN has recently been extended for RTP [18]. In TCP, upon reception of ECN marked packets, a well behaved (and ECN enabled) sender will throttle itself to alleviate the network congestion by adjusting its send window. The overall behavior is similar to how TCP reacts to dropped packets, except that with ECN no packets need to be dropped. ECN uses the last two remaining bits in the outdated type of service (TOS) field of the IPv4 header to indicate if ECN is supported or not by the end-points, and to mark packets during congestion. In IPv6 ECN uses the last two bits of the traffic class field.

ConEx extends ECN with the aim of:

1) providing a mechanism for routers to know how much congestion is expected downstream (between the said router and the receiver);

2) providing accountability of senders that take part in congested routes.

More precisely, ConEx works as in the following: similar to ECN, routers mark packets when experiencing congestion. Routers can estimate how much congestion a packet will experience downstream by subtracting the rate at which they mark packets from the rate of marked packets they received (for a given flow, source. etc.). This information represent the accountability of a sender with regards to the congestion caused. Moreover, the information can be used by routers to enhance QoS mechanisms in the presence of congestion. The use cases of ConEx [6, 11] are out of the scope of this work and left for future work.

In IPv4, ConEx is supported by Re-ECN and uses the last unused bit in the IPv4 header in addition to the two ECN bits. Since there are not enough free bits in the IPv6 header to support ConEx (Re-ECN), the IETF proposes a header extension [10] in support of ConEx. This option provides a 32-bit field from which 4 bits are currently specified for ConEx and the remaining bits are reserved for future use. This paper proposes to use these reserved bits both as a feedback mechanism from receiver to sender, and as request mechanism for QoE enhancement from the senders to routers along the way.

3. FEEDBACK OF QOE FOR QOS ENHANCEMENT

Figure 1 describes the proposed mechanism on top of the ConEx procedure. The ECN/ConEx procedure steps are marked with '*' and are also optional to the QoE feedback mechanism. However, both mechanisms can easily co-exist. Each arrow on Figure 1 is describe below:

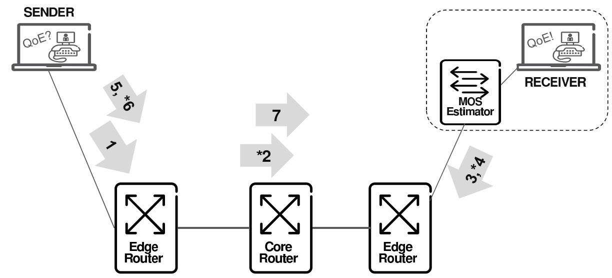

Figure 1: Feedback mechanism using ConEx

Table 2: QoE feedback formats

QoE Data	Description
Plain value of QoE	The actual QoE metric (e.g. R-Factor).
QoE Variation	The variation between two QoE computation time intervals.
Speed of variation of QoE	The rate of change of the variation in a given time frame.
Single-bit for signalling	A single bit to signal that QoE enhancement would be required. (note: only option with IPv4).

1) The sender sends packets belonging to a flow.

*2) Optionally, routers along the path can congestion mark packets (assuming ECN is on).

3) Receiver, or edge node monitoring the flow, will calculate and feedback QoE metric experienced by the end-user for that flow. Feedback is made in-band, in IP packets belonging to the flow (e.g. in ACKs messages).

*4) Optionally, receiver will feedback congestion experienced (assuming ECN is ongoing).

5) Sender re-feeds back the QoE information to routers along the path to enhance their QoS mechanism (and this will results for overall a better QoE).

*6) Optionally, sender will mark RE congestion experienced back towards the network (assuming ConEx is ongoing).

7) Routers perform QoS adjustment (e.g. new queuing scheme, etc.).

3.1 QoE feedback formats

The QoE information value may be returned in various formats as presented in Table 2. First, for *plain value of QoE*, each packet will carry the most up to date QoE value associated with the flow. This can be the E-Model's R-Factor value for example. *QoE variation* is usually calculated from a sample window of two or more packets forming the QoE time interval. This value can be positive or negative, to signify respectively an improvement or degradation in QoE. *QoE speed of variation* is the rate at which the QoE variation changes in a given time frame window. It is the speed of change and could carry a lot of insight in how the

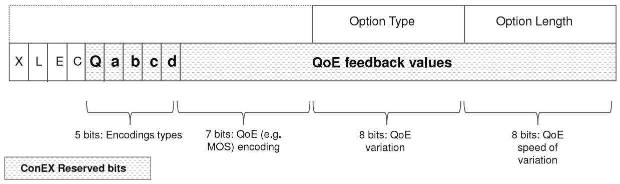

Figure 2: ConEx destination header for QoE feedback

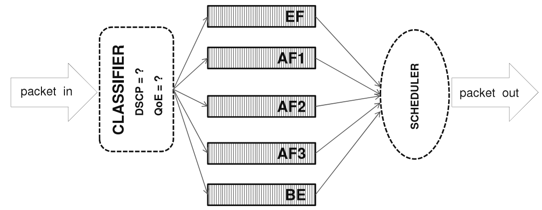

Figure 3: Multi-field DiffServ classifier using QoE information

QoE is fluctuating during the flow's lifetime. At last, it is possible to use a single bit to convey the information of QoE degradation with need for QoS upgrade. However, the nodes along the path will not have more information to base their decision on, as it is the case with the other QoE formats. However, as it will be noted in Section 4 below, this format is currently the only format available for IPv4 ConEx based feedback.

4. QOE FEEDBACK USING CONEX

The proposed mechanism uses the ConEx feedback option to signal in-band the need for QoE enhancement by QoS adjustment of subsequent packets. The QoE information value may be returned in various forms as explained in Section 3.1 above. Using IPv4, only single bit signalling is possible. ConEx's RE control flag is positioned where the 'reserved' control flag was, i.e., at bit 48 of the IPv4 header (starting from bit zero). It is possible to use the currently unused Re-ECN encoding, with ECN field = 10 and RE flag = 1 for signalling QoE degradation (see Table 1 of Ref. [5]). Using the same RE control flag bit to signal QoE enhancement requests prevents the use of ConEx itself. Moreover, reverse flow packets carrying QoE information from receiver towards the sender will not be differentiable from sender to receiver packets. This means that the reverse direction packet (e.g. ACKs) will also signal QoE enhancement request to routers.

The rest of this paper assumes IPv6 ConEx. Figure 2 shows how the 28 reserved bits in the ConEx option header for IPv6 can be used to carry a feedback of QoE information. First, if bit **Q** is set, it signifies that QoE feedback mechanism is on and that routers need to process the packet using the transported QoE encoding. Any router along the path that supports this feature will then use the QoE information, when applicable, to enhance QoS mechanisms. Bits **a**, **b**, **c** signify that the option header carries QoE, QoE variation and QoE speed of variation information. Their value allows for up to eight types of QoE information (e.g. R-Factor, MOS equivalent for live video, MOS equivalent for streaming, etc.). Bit **d**, if set, signifies that QoE must be enhanced (equivalent to the IPv4 implementation). Then, QoE (7 bits), QoE variation (8 bits) and QoE speed of variation (8 bits) information are possibly carried as unsigned integers in the remaining bits, as shown in Figure 2. Other encodings of these 28 bits are possible but this is out of the scope of this paper and is better tackled by standardization bodies like IETF.

As mentioned previously, each packet carries the most up to date QoE information related to the application flow to which they belong. Any QoS or traffic engineering mechanism can leverage on this information to further optimized overall performance. QoE-feedback based QoS mechanisms can be implemented in access nodes or in all core router nodes. Section 5 below gives the example usage of QoE feedback by a new queue management mechanism.

5. FEEDBACK OF QOE AND QUEUE MANAGEMENT

As an example of QoE feedback QoS mechanism, this paper takes the case of a DiffServ based network where packets are marked based on policy (e.g. depending on the user and the application). Each packet thus carries a fixed DiffServ code point (DSCP) and dynamic QoE information as presented in Sections 3.1 and 4. Usually packets belonging to real time or multimedia flows will carry QoE information. The QoE information is computed by the receiver, or edge node near the receiver, and encoded in each header as depicted in Figure 1. Each router in the DiffServ domain typically implements a queuing mechanism with DiffServ based priority queuing and possibly a variant of weighted fair queue (WFQ) scheduling. The QoE information can be a key ingredient to the queue management system. It is assumed up to five priority queues are implemented for each of the DiffServ classes: expedited forwarding (EF), assured forwarding (AF1, AF2, and AF3), and best effort (BE). When the packet arrives at the node, based on its DSCP it is usually assigned to one of the queues.

With the proposed mechanism however, as depicted by Figure 3, the multi-field (MF) classifier first examines the DSCP as usual, then it examined the QoE information. If QoE enhancement is not required, then the packet is assigned to the queue associated to its DSCP. If QoE enhancement is required (e.g. QoE value is lower than a threshold or the variation of QoE is negative) then the classifier can assign the packet to a higher queue class. The classifier will need to consider the overall queue lengths and estimate the gain before doing the move. It is important to point out that the DSCP is left unchanged allowing the classifier in each node along the path to perform a similar decision. Figure 4 shows a simplified flow diagram of when the packet enters the node until it leaves the node. First the router node must extract the *Q bit* from the ConEx IPv6 header (point 1 in Figure 4). If the *Q bit is set* and the *QoE information* shows a need for QoS improvement, the mechanism continues (point 2a). Otherwise (point 2b), the packet is sent to the normal *DiffServ scheduler* before leaving the node (point 5).

For sake of simplicity, other traffic engineering mechanisms are not shown and are assumed as part of the *DS scheduler/packet out* step (point 5).

As the mechanism continues, the *DSCP*, *QoE informa-*

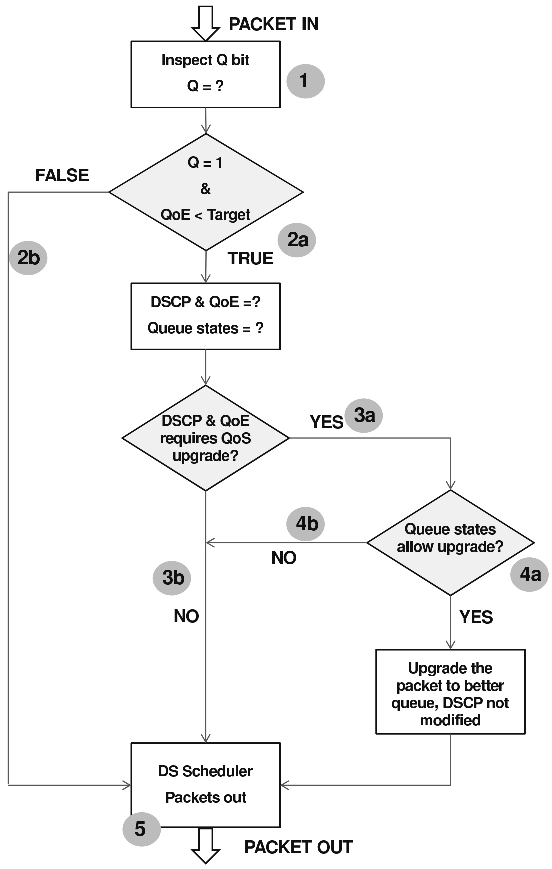

Figure 4: Example algorithm for multi-field DiffServ classifier and scheduler

tion and *queue states* are evaluated (point 3a,b). Further analysis of QoE and DSCP information determines if a QoS upgrade is required. If not (point 3b), the packet is sent to the *DiffServ scheduler*(point 5). If yes (point 3a), the packet continues to further determine if upgrade is allowed based on current conditions (point 4a,b).

Queue models are then used to estimate the effect of QoS upgrade by assigning the packet to a higher class queue than its DSCP alone would have allowed (point 4a,b). If the *queue states* and evaluation allow the QoS upgrade (point 4a), the packet is sent to the *DiffServ scheduler* (point 5) but to a higher level queue. Moreover, inside the same queue, the packet could be assigned to a better level drop precedence, e.g., from AF3 to AF1. This upgrade is done without re-marking the DSCP (point 4a). If upgrade is not allowed (point 4b), the packet is not offered enhanced QoS and just continues to the *DiffServ scheduler* (point 5).

5.1 Simulation Results

This proposal's validation is initially performed using simulations on ns-3.13 platform [2]. The DiffServ module of Ref. [16] is used after integrating it in ns-3.13. The topology shown in Figure 5 is simulated with ten VoIP flows at an average rate of 28kbps. These flows are divided between DiffServ AF1 and AF3 classes. WFQ is configured with AF1 having 45% of the weight, AF2 having 32%, AF3 having 18%, and BE having the remaining 4%. The background traffic is composed of file transfer protocol (FTP) and user datagram protocol (UDP) traffic mix; the simulation scenarios presented in this section do not deteriorate the overall throughput of the BE traffic.

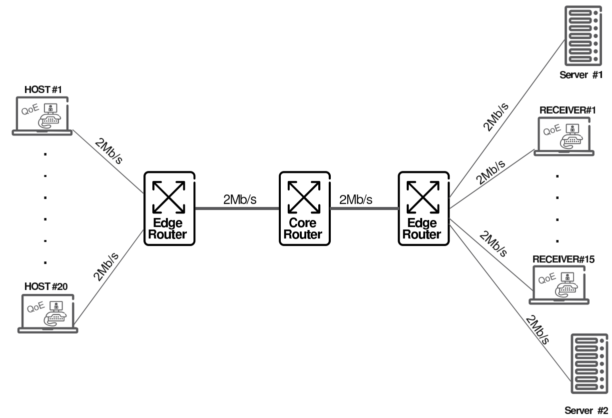

Figure 5: Simulated topology

Table 3: Simulation scenarios

Scenario	Number of Flows	Percentage of VoIP traffic
1	19	52.6
2	21	47.6
3	22	45.5
4	24	41.7

Table 4: QoE feedback QoS results- QoE gain

Scenario	average gain	MIN gain	MAX gain
1	0.98%	10.14%	-0.92%
2	1.45%	7.84%	-0.95%
3	5.40%	24.53%	-2.03%
4	2.29%	11.23%	-0.95%

Table 5: Measured MOS values

Scenario	MIN	MAX	AVG
1	2.1	4.1	3.2
2	2.2	4.1	3.3
3	1.8	4.1	3.1
4	2.1	4.1	3.2

The MOS value is computed using a simplified E-Model implementation. The MOS value carried consists of the average MOS from a time interval of about five seconds. The MOS speed of variation is computed using a linear regression model in the same time frame window. MOS, variation and speed of variation are all considered when making the decision about the queue class upgrade as shown in Figure 4.

Table 3 presents the traffic used for each simulation scenario. The simulation ran for many scenarios of which four were selected for this paper. For each scenario, the number of VoIP flows remained constant while the background traffic was varied.

The tables above present some results which compare the value of the average MOS for all the VoIP flows when the proposed mechanism is not used versus when it is used (mechanism is used only for the VoIP flows). Table 4 shows the comparison for various scenarios of the mechanism with the absence of the mechanism.

Table 5 shows the measured MOS values per scenario. These are presented to give more insight into the gain values shown in Table 4. The results are the lowest value observed irrespective of the presence or absence of the proposed mechanism.

These preliminary results show that there is in fact a direct effect in QoE measurements when queue management is based on fed-back QoE. However, more tuning and trials are necessary to propose guidelines on configuration and threshold values to use. One possibility can be the self-adjusting configuration of the thresholds based on the QoE variation and speed of variation results. Future studies are outlined in Section 6 below.

6. CONCLUSIONS

This paper presented a novel method to enhance QoE using a loopback mechanism where QoE is fed back to QoS mechanisms allowing the latter to readjust and to subsequently improve the QoE. The feedback is made possible in a viable way using the ConEx mechanism, as an in-band IP layer signalling protocol. The proposed QoE feedback mechanism was defined and applied for a DiffServ based queuing system where packets belonging to one flow could temporarily experience enhanced QoS by queue class upgrade.

As future work, the proposed queuing management system needs to be further investigated to propose clear configuration guidelines or even a self-configuring method. Moreover, more work is needed to study the effect of QoE feedback for enhancing other QoS mechanisms and in particular other queuing mechanisms like CoDel [14]. Furthermore, the QoE information can possibly enhance existing ConEx use case. The reverse can also apply, where the ConEx information can enhance the router decision making in deciding if it is the cause of QoE deterioration. This is also left as subject for future investigation. Finally, the question of untruthful hosts needs to be addressed.

7. ACKNOWLEDGMENTS

This work was partially funded by the Natural Sciences and Engineering Research Council of Canada (NSERC). The authors also thank the anonymous reviewers who's comments improved the understanding of this paper and raised interesting concerns to be addressed in ongoing and future works.

8. REFERENCES

[1] Congestion Exposure (ConEx). IETF Working group, http://datatracker.ietf.org/wg/conex/charter.

[2] Network Simulator. NS-3, http://www.nsnam.org/.

[3] T. H. Anh, A. Mellouk, and S. Hoceini. User to user adaptive routing based on qoe. In *ISPS'11, 10th International Symposium on Programming and Systems*, pages 39–45, April 2011.

[4] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Service. RFC 2475 (Informational), December 1998. Updated by RFC 3260.

[5] B. Briscoe, A. Jacquet, T. Moncaster, and A. Smith. *Re-ECN: Adding Accountability for Causing Congestion to TCP/IP*. IETF, April 2012. Status: Individual Draft: draft-briscoe-conex-re-ecn-tcp-00, Work in progress.

[6] B. Briscoe, R. Woundy, and A. Cooper. *ConEx Concepts and Use Cases*. IETF, July 2012. Status: WG Draft: draft-ietf-conex-concepts-uses-05, Work in progress.

[7] T. Friedman, R. Caceres, and A. Clark. RTP Control Protocol Extended Reports (RTCP XR). RFC 3611 (Standards Track), November 2003.

[8] ITU-T. The e-model, a computational model for use in transmission planning. Recommendation G.107, International Telecommunication Union, Geneva, April 2009.

[9] E. Jammeh, I.-H. Mkwawa, A. Khan, M. Goudarzi, L. Sun, and E. Ifeachor. Quality of experience (qoe) driven adaptation scheme for voice/video over ip. *Telecommunication Systems*, 49(3):99–111, January 2012.

[10] S. Krishnan, M. Kuehlewind, and C. Ucendo. *IPv6 Destination Option for ConEx*. IETF, March 2012. Status: WG Draft: draft-ietf-conex-destopt-02, Work in progress.

[11] D. Kutscher, F. Mir, R. Winter, S. Krishnan, Y. Zhang, and C. Bernardos. *Mobile Communication Congestion Exposure Scenario*. IETF, July 2012. Status: WG Draft: draft-ietf-conex-mobile-00, Work in progress.

[12] M. Mathis and B. Briscoe. *Congestion Exposure (ConEx) Concepts and Abstract Mechanism*. IETF, July 2012. Status: WG Draft: draft-ietf-conex-abstract-mech-05, Work in progress.

[13] N. T. Moura, B. A. Vianna, C. V. N. Albuquerque, V. E. F. Rebello, and C. Boeres. Mos-based rate adaption for voip sources. In *ICC'07, IEEE International Conference on Communications*, pages 628–633, June 2007.

[14] K. Nichols and V. Jacobson. Controlling queue delay. *Communications of the ACM Magazine*, 55(7):42–50, July 2012.

[15] K. Ramakrishnan, S. Floyd, and D. Black. The Addition of Explicit Congestion Notification (ECN) to IP. RFC 3168 (Standards Track), September 2001.

[16] S. Ramroop. A DiffServ model for the NS-3 simulator, University of West Indies. http://www.eng.uwi.tt/depts/elec/staff/rvadams/sramroop/index.htm.

[17] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550 (Standards Track), July 2003.

[18] M. Westerlund, I. Johansson, C. Perkins, P. O'Hanlon, and K. Carlberg. Explicit Congestion Notification (ECN) for RTP over UDP. RFC 6679 (Standards Track), August 2012.

[19] Q. Zizhi, S. Lingfen, N. Heilemann, and E. Ifeachor. A new method for voip quality of service control use combined adaptive sender rate and priority marking. *ICC'04, IEEE International Conference on Communications*, 3:1473–1477, June 2004.