Packet Handling and Quality Shaping for Real-Time Services in Futuristic Internet

Leif Arne Rønningen and Ameen Chilwan
Department of Telematics
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
leifarne@item.ntnu.no, chilwan@item.ntnu.no

***Abstract*—Distributed Media Plays (DMP) is a futuristic architecture for the Internet. It is specifically designed keeping in mind that a number of futuristic Internet services have real-time constraints, particularly the end-to-end delay has to be less than 20 ms. These constraints need to be fulfilled by the network and the current RTP/UDP/IP protocol stack is found to be limited in this regard. Thus an alternative network architecture is proposed and being developed with special, although compatible with IPv6, packets. The handling of these packets in DMP network is explained in this paper.**

The foundation of DMP system is laid on the quality requirements of the futuristic Internet services. Therefore, one of the features of DMP is that the content quality is gracefully degraded in case the network is overloaded and the traffic injection has to be reduced. This is provided by introducing a scheme for "Quality Shaping" which is explained in detail in this paper. Also a new "Link Flow Control (LFC)" functionality is introduced in DMP architecture to enhance quality control of contents, which is also explained herein. The LFC functionality helps the DMP network nodes to have an overview of the capacity bottlenecks in the network and hence adjust the traffic flow accordingly.

***Index Terms*—Delay systems, Internetworking, Quality of service, Real-time systems.**

I. Introduction

The quantity and nature of services being deployed on the Internet nowadays is increasing dramatically. The Internet has evolved incredibly, both in size and capacity, during the past two decades. But the nature of services which are in pipeline to be deployed in future Internet forces the research community to come up with even faster and more robust Internet. One direction which these novice deployments are taking is real-time services. The services which typically require real-time guarantees are spread across a number of fields and disciplines. These applications can be related to music; for example distributed ensemble [1], remote jazz sessions [2] and distributed orchestra, or to entertainment; for instance multi-player gaming and Multimedia Home Space (MHS) [3]. In addition, remote surgery, under-sea oil exploration and near-natural business meetings can also be considered as the services with real-time requirements which are soon to be deployed in the Internet.

Moreover, model-driven solutions for real-life systems are expected to emerge in not very distant future. Nowadays, models may control real-life processes owing to high processing power due to parallel processing. However, in order to harness such processing power, very fast and broadband connections between hardware drivers and sensors are essential. Such networks should also be equipped with mechanisms that allow for on-fly data reduction to increase effective throughput.

Cisco has recently made a leap towards providing highly function-specific systems to its users for a number of applications. Some of these systems promote the concept of "near-natural presence", a term coined by the authors to reflect that the quality of the collaboration between communicating parties is as natural as they are present at the same place. Cisco refers to such systems as Telepresence Systems and provides the entire necessary network infrastructure to setup a telepresence environment. Such systems have found applications in virtual meeting environments and also have a number of other uses as listed in [4]. In order to provide a high level of quality of service (QoS), Cisco has also developed specific protocol for such systems, an example being Telepresence Interoperatability Protocol (TIP) [5].

Although telepresence systems exist today on the Internet, they don't guarantee end-to-end delay between collaborating parties. This should not be a problem for current use of such systems, but as the real-time requirements of services will increase, the network will have to guarantee real-time constraints, and more precisely the end-to-end delay. The threshold value of the parameter for different applications is different, for example; it is found to be 14 ms for distributed ensemble [1]. An estimated value of 20 ms is considered to be acceptable for real-time services [2].

The quality requirements for such services are not limited to guaranteed end-to-end delay with a certain threshold, though it is the enabling parameter. They are extended to high temporal and spatial resolutions resulting in high data rate requirements. In addition to these two network-related parameters, a number of other parameters are also of importance in order to realize most of the example services mentioned earlier. These requirements include; auto-stereoscopic multi-view adaptive vision, surround multi-channel sound, graceful degradation of quality in case of congested network, guaranteed delivery-sequence of control packets and node-to-node security [6].

In order to provide a resilient network platform for deployment of real-time services, the RTP/UDP/IP protocol stack was initially considered and novel mechanisms for queuing, traffic

978-0-7695-4927-9/12 $26.00 © 2012 IEEE
DOI 10.1109/FIT.2012.9

balancing and traffic shaping were developed and studied. Some of these studies are presented in [7]–[9]. But the real-time requirements were found to be hardly achievable, because part of the latency was contributed by processing which is quite complex in TCP/IP protocol stack due to high level of header encapsulation. Another European Union (EU) research project, 3DPresence [10], estimates that the end-to-end time delay is much higher than the required values. Therefore, a futuristic network protocol stack, termed as Distributed Media Plays (DMP), was proposed with a single header for all the network-related functionalities [3]. The DMP network architecture is presented in Figure 1.

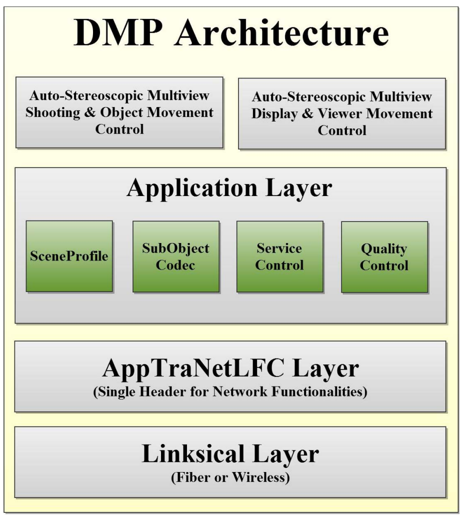

Fig. 1. DMP Network Architecture for Real-Time Services

DMP network architecture uses the notion of graceful degradation of quality, not only to provide acceptable level of user perception of the service quality, but also to control the network footprint of the service. It is done by constantly monitoring the network performance and shaping the service traffic accordingly. But, it is not done in the manner as the traditional traffic shaping is done where the amount of traffic generated by a source is a matter of concern [11], [12]. Instead, the main focus is on the quality aspect of the service and hence the dropping is done by considering the final quality to be presented to the viewer. This proposes the name "Quality Shaping" for such controlled dropping of packets [3]. It is found crucial in futuristic Internet services that content-driven dropping shall be done to keep the amount of traffic injected in the network within specific limits, at the same time maintaining the quality requirements of the users.

In this paper, the theoretical basis for packet handling and quality shaping schemes proposed for DMP-based systems are presented. It is of highest concern that the basic requirements for end-to-end delay and delivering acceptable perceived quality levels are not compromised. The philosophical background and practical layout is explained without performing, at this stage, optimization and evaluation of the mechanisms.

The rest of the paper is structured in the following pattern. Section II gives an overview of the technology on which the quality shaping scheme is designed to work, i.e. DMP. Section III deals with the nature of packets which exist in DMP, their forwarding in DMP network nodes, and eventually their classification in order to provide basis for their controlled dropping using quality shaping scheme. Sections IV and V detail the proposals for Link Flow Control (LFC) and quality shaping scheme respectively, as proposed by the authors. The paper is concluded in Section VI with a note on future directions in Section VII.

II. Background Information

A. Distributed Media Plays (DMP)

Futuristic services have aroused the need for a novel protocol stack, other than the current TCP/IP, for catering the real-time requirements associated with some of these services. A futuristic stack, termed as DMP, is proposed and is currently under continuous study and hence rapidly taking shape. This protocol stack was first published in [3] and it was a shift from the earlier work for fulfilling real-time requirements of the services using RTP/UDP/IP protocol stack. Although a number of methods were employed for enhancing the functionalities of the latter [9], but the results were not as required for deploying real-time service. Hence DMP was proposed.

The network architecture proposed in DMP is depicted in Figure 1 and explained in detail in [3]. In this paper, only a very brief introduction and highlights of the most important features is given. It is three-layer architecture and there exists a layer on top of these layers. The top-most layer is typically divided into two distinct control groups and collectively they are a part of a **Collaboration Space**. A *collaboration space* is defined as the environement for DMP deployment at the user-premises. It includes all the displays, audio systems, Camera Cluster Arrays (CCA) and all the other equipment required for shooting, displaying and performing even more sophisticated functions, such as tracking the viewer's eye or object's movement for providing a better 3D experience. This is shown as blocks in Figure 1, a block for the viewing functionality of a space and the other for the shooting functionalities.

The remaining three-layers from the network protocol stack are collectively called Real-Time Internet (RealInt). This term has recently been adopted by the authors in order to differentiate between the more generic term DMP which is used for the whole system and more specific network-related part of it. As this segment of the DMP system is more important to our work, we will dig a little deeper in its details.

B. Real-Time Internet (RealInt)

As already stated, RealInt is the part of DMP system dealing with network related functions. The proposed system is designed to have two distinct types of nodes; **AccessNodes** and **CoreNodes**, in addition to a number of specialized servers. The former are closer to the user premises while the latter comprise the required back-bone network for the DMP system. Depending upon the specific functions, different nodes have different blocks in their typical configuration as shown in Figure 2.

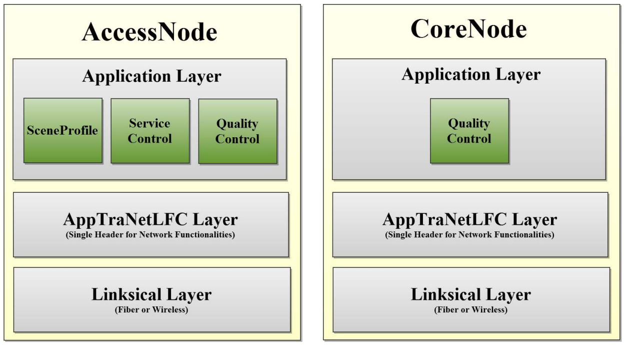

Fig. 2. Network Nodes for Real-Time Internet (RealInt)

1) Linksical Layer: The lower-most layer in RealInt is the **Linksical Layer** which is a blend of physical and data link layers of the OSI protocol stack. There is a combined header for the functionalities of the two layers. It is perceived that as the processing complexity is simplified, the processing time for a packet will be reduced compared to the case when there are distinct physical and datalink layers.

The RealInt proposal for Linksical layer proposes wireless and fiber connections between the nodes. This proposal is made because wireless connections provide lowest propagation delays and hence can play an important role in adhering to the latency budget for real-time requirements. But optical fibers are being spread very fast as preferred access network technology due to their robustness and high capacity. Conversion between optical and electrical signals is made at each end of the optical link. In lieu of current advancements in optical switching and routing technologies [13], it can be considered to change the whole backbone part to optical-only providing fast traverse paths.

Another aspect of this layer is that it is proposed to be implemented as firmware directly on hardware [6], thus removing the delays incurred due to software implementation of protocols. The proposal for implementation on PCI Express (PCIe) is developed and detailed in [14]. This implementation approach also gives an opportunity for parallel processing of headers which mainly contain preamble for recognizing bit start and clock synchronization.

2) AppTraNetLFC Layer: This layer comprises the functionalities of upper OSI layers, namely the application, presentation, session, transport, and network layer. It aggregates all the required functionalities in a single header and in addition, also provides Link Flow Control (LFC) to the packets. It uses IPv6 protocol for forwarding and routing and IPsec security mechanism for authentication and authorization, with some modifications [15]. This implementation scenario gives a possibility for DMP system to be hosted in the existing Internet. The details of compatibility between the two systems and the precise definition of AppTraNetLFC packets is given in [6].

A new functionality, LFC, is subsequently added to this protocol after its initial design and it will be described later in Section IV. Again, in order to capitalize on the faster processing times due to hardware implementation of protocols, the protocol is proposed to be implemented on hardware using FPGAs or customized ASICs.

3) Application Layer: The top-most layer of RealInt is the **Application Layer**. DMP application layer is different from its OSI counter-part because it has DMP-specific functions to perform. The functions include establishing a DMP connection, negotiating initial QoS guarantees of the collaboration, implementing quality shaping scheme for graceful quality variability and so on. These functions are shown as blocks in Figures 1 and 2.

III. AppTraNetLFC Packets

A. Classification of Packets

Two distinct types of packets are defined in DMP, these are Audio-Visual (AV) packets and control packets. It is clear from the nomenclature that AV packets carry the data while control packets are used for collaboration control purposes.

1) Audio-Visual (AV) Packet: It traverses through nodes experiencing delays but the processing delays in user equipment can be guaranteed lower than a specified value. Except for output link queues in nodes, there is negligible waiting. Sufficient information is included in the packets, so that each AV packet can be interpreted independent of the contents of other packets. Thus, the contents of a packet can be used to represent parts immediately, at the right place and with the right quality (variable, however) without waiting for other parts of the object or other objects. Pre-stored (negotiated) configuring data such as SceneProfiles, are used (without delay) in the rendering process.

2) Control Packets: They are used for setup of collaborations, adaptive control during collaborations (Quality Shaping), and teardown of collaborations. Such packets are defined to have various priority classes, seen from their application point of view. These classes along with the classification criteria are shown in Table I. The table is sorted in decreasing order of priority.

TABLE I
Classification of Control Packets

Packet Class	Real-Time Requirement	Delivery Requirement
Class A	Guaranteed	very low loss probability
Class B	Moderate	sequential delivery
Class C	None	sequential delivery
Class D	None	no losses any delivery sequence
Class E	None	packets can be lost

B. Packet Handling in Network Nodes

It is already stated that control packets differ from each other depending upon their respective applications. Some control packets are used in typical request-response sequences. It is up to the application to decide how to handle the situation

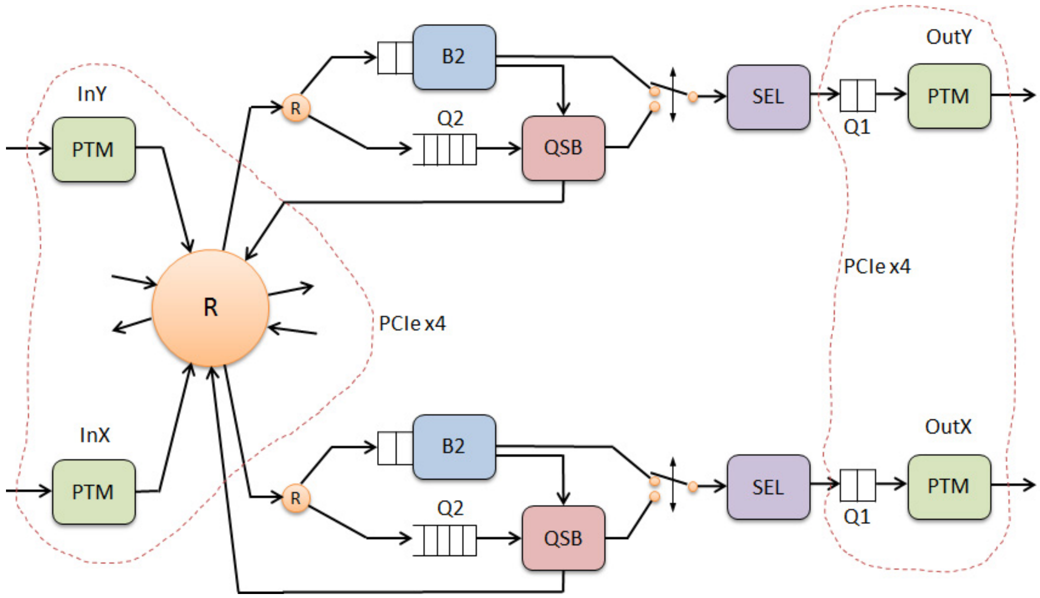

Fig. 3. Packet Forwarding and Quality Shaping mechanisms in RealInt

if packets that need acknowledgement are not acknowledged. Since such requests do not (normally) have real-time requirements and shall have an extremely low probability for being lost, the output link queuing system, shown in Figure 3, is introduced as part of the AppTraNetLFC protocol in all nodes after routing by block **R**. Control packets enter queue **Q2**, which holds packets in a very large queue. If selected by **Sel**, with probability **p1**, the control packet enters **Q1** for link output. The module **OutX/OutY** is an abstraction of the output link and physical layer, sending the packets out at the maximum packet rate given by the link capacity. Control packet transport delays are highly variable, depending on traffic patterns. The maximum length of **Q2** should be so large that reaching the maximum should have an extremely low probability. Before this happens, the admission control shall decrease the traffic injection at the sources.

Class A packets transporting critical multimedia content with a guaranteed maximum end-to-end delay are sent to buffer **B2**. The selector **Sel** fetches packets from **B2** with probability **1-p1**, and forwards them to **Q1**. **Q1** has a limited length and this determines maximum jitter of a transfer through the node. **B2** is a very short buffer used for dropping according to the sequence number of the packet. If **Q1** is full, packets start queuing in **B2** where they are dropped according to algorithm given in [3].

Since a path through the network includes nodes in series, as shown in Figure 3, the maximum end-to-end time delay of multimedia content packets can clearly be guaranteed. If the processing times in user equipment and switches (nodes) are constant, the end-to-end packet delay is a sum of propagation times in the path, the constant processing times in path nodes, and the waiting times in **Q1** in path nodes (the waiting time in **B2** can be neglected compared to the waiting time in **Q1**).

The only undefined block in Figure 3 remains **QSB** which is discussed in Section IV.

IV. Link Flow Control (LFC)

The flow control mechanism introduced in this paper is applied to control packets going through **Q2**, that is control packets with priority B, C and D, since for class E no quality requirements need to be fulfilled, these packets may become lost and may be delivered out of sequence. Class A control packets are dealt with as NOC packets that shall not be dropped, and go through **B2**. Negative Acknowledgement Flow Control (NakFC) and Acknowledgement Flow Control (AckFC) packets are prepared by the Quality Shaping Block (QSB), which reads the current SeqNo1, SeqNo2 or Credit values, and sent via router **R** to **B2**, Figure 3. The **QSB** also makes measurements on all queues of the output link and forwards them to the next node via **R** and **B2**. The measurements are carried by Quality Shaping packets that are treated in the same manner as NOC packets and are not dropped.

The **QSB** needs to store control packets of classes B, C and D that are sent out but not yet acknowledged. When an AckFC packet is received via **B2**, all stored packets with sequence number up to MaxCredit (the last packet acknowledged) are discarded. If a NakFC packet is received, packets with sequence numbers between SeqNo2 and SeqNo1 are re-sent. The receiving **QSB** in the other end of the link has to queue a number of received packets if a CRC of a packet fails, and send NakFC (SeqNo2, SeqNo1). When all packets between SeqNo2 and SeqNo1 are correctly received they are put in the right place in the sequence and can be sent out. It is expected that SeqNo2 most of the time is SeqNo1+1, that is, burst errors are rare. Since the AckFC and NakFC delay and delivery can

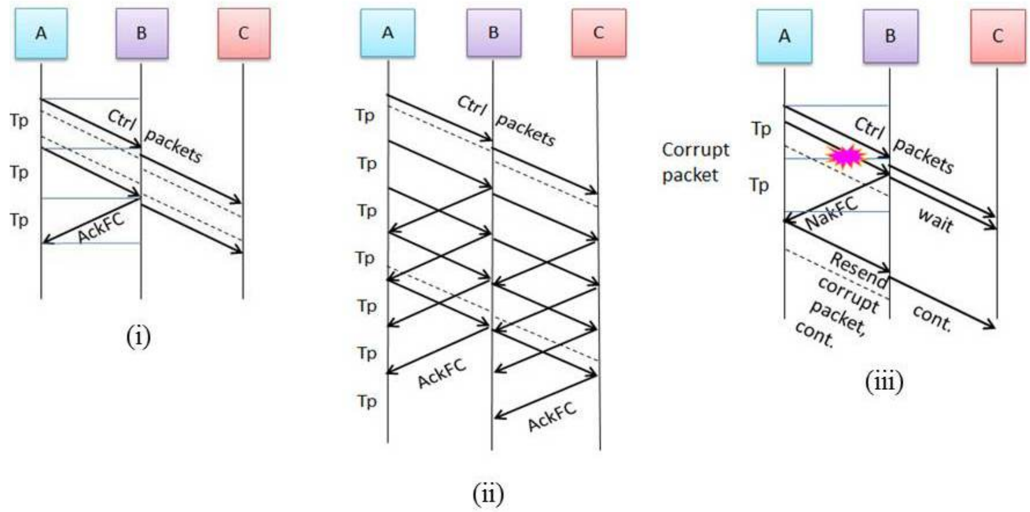

Fig. 4. Scenarios of Link Flow Control (LFC)

be guaranteed, the retry delay can be calculated. When packets of class D require retry, the receiving block does not have to wait to maintain the sequence of outgoing packets.

The propagation delay between two nodes, A and B, is denoted as Tp. During the propagation delay Tp, A sends out Np packets such that, Np = C.Tp, where C is the capacity of the link. After delay Tp, the first packet arrives at B. If all packets received during the next Tp are successful, B sends an AckFC packet back to A. This takes another Tp, meaning that from start of the burst until reception of AckFC it takes a total time of 3Tp, as depicted in Figure 4(i). If, in the worst case, the last packet of the burst from A to B is corrupted, it takes a time 3Tp before A knows this, which means that A has to store 3Np = 3C.Tp packets to be able to retransmit the lost packet(s). If the packet burst is sent further from B to C, the same sequence repeats, Figure 4(ii) shows such a case.

If a corrupt packet is received, a NakFC (SeqNo2, SeqNo1) is sent, asking for retransmission of packets with sequence numbers between SeqNo2, SeqNo1. Figure 4(iii) illustrates this when a single packet in a burst fails. It takes a time 2Tp before a corrupt packet is resent, and 3Tp before it is received successfully. The consequence of this is that the SeqNo parameter needs a max value of at least 3Np before it wraps.

V. Quality Shaping

A. Background Traffic Shaping

Traffic Shaping, the concept that Quality Shaping builds upon, was first proposed in [11], where variance reduction of traffic streams by queuing and dropping were evaluated. Later, traffic shaping was further developed and the Leaky Bucket [12] and Token Bucket flow-control schemes were proposed. A large number of traffic control algorithms have been proposed since.

B. Quality Shaping Profiles

The quality of a scene (a frame in a video clip) can be changed in many ways. Possible parameters that can be varied are referred to as **QualityShapingProfiles**, Table II. All nodes, AccessNodes included, perform selective packet dropping (from sub-objects) based on agreed QualityShapingProfiles. Only the user can change the content within packets. The QualityShapingProfiles are based on parameters describing temporal and spatial scene resolution and composition. These parameters are listed in Table II.

TABLE II
Parameter List of QualityShapingProfiles

S.No.	Parameter	Remarks
1	Number of sub-scenes in a scene	
2	Number of objects per sub-scene	
3	Number of stereoscopic views per object	
4	Number of sub-objects per object	
5	Update rate of each sub-object	
6	Components representing the sub-object	e.g. RGB
7	The Alpha channel depth	e.g. RGBA
8	The sample depth of each component	e.g. RGBA - [16,16,16,16]
9	Sampling rate of each component	e.g. YCrC 4:2:2
10	Shape and size of sub-object	Shape Mask
11	Compression and coding scheme for sub-object	NOC or Mod. JPEG2000

C. *Quality Shaping Mechanism*

Quality Shaping is not based on individual streams, but on the aggregated packet streams from each source AccessNode loading a network node. Many AccessNodes normally contribute to the packet stream into an output link of a network node. The leading 80 bits of the user's IP address uniquely identify the AccessNode he is connected to. AccessNodes initiate QualityShaping packets at certain intervals (say 5 ms, an important configuration parameter). An AccessNode receiving packets from other AccessNodes sends QualityShaping packets to those AccessNodes, but not to others.

After a collaboration has been set up, the users start shooting scenes and sending AppTraNetLFC packets. The packet header parameter PacketRate indicates the scene sub-object's packet rate and defines the time when the rate change takes place. The rate change is indicated by sending a packet of type, e.g., PT = 'Visual, change of rate' and a Timestamp value that defines the time until the rate change takes place. The QualityControl of the AccessNode measures the overall packet rate of each scene, and takes action if it is different from that indicated by the scene's sub-object packets.

Nodes regularly forward QualityShaping packets to source AccessNodes that reports packet arrival count, drop count, departure count, output link capacity, mean length of control packet queue and other measurements in the interval. All nodes in the established routes check the Payload type of all packets, and if it is a QualityShaping packet, it is delayed shortly allowing the nodes to write their measurements into the packet body. When the packet arrives at the source AccessNode, the QualityControl analyses the measurements, and takes the decision to start dropping user packets or not. Normally several users contribute to the overload in the remote node, and the QualityControl/ServiceControl sends new QualityShaping packets to all of them. It is up to the users to ignore, or alter parameters according to the QualityShapeProfile.

Each AccessNode includes a QualityShaper that receives QualityShaping packets from network nodes, and that actually scales the packet rates from users by controlled dropping from sub-objects.

VI. Conclusion

In the current work, the need for having a network architecture which can cater for real-time constraints of futuristic services is motivated. The constraint is already been quantified and is found to have the threshold value of 20 ms for musical performances. In this paper two important advancements towards the realization of DMP system, a system based on futuristic Internet architecture with emphasis on distributed arts performances, is presented. First of them, the packet forwarding mechanism, is detailed with emphasis on technical aspects of it. For the second advancement which deals with quality control of contents, two mechanisms working on different layers in DMP architecture in order to account for quality-oriented traffic handling are explained. These features are Link Flow Control (LFC) and quality shaping. The former is essential for preventing the network from getting congested while the latter serves the task of gracefully degrading the perceived scene quality by dropping packets in controlled fashion.

VII. Future Directions

In this paper, the theoretical basis for packet handling, quality shaping and Link Flow Control (LFC) for DMP systems is laid. These mechanisms are now to be realized, preferably starting with simulations, and accounted for their functionality and performance. The iterative method for the refinement of these mechanisms can then be used to give them a final definite shape.

Hardware implementation of these schemes is to be carried out so that they can be analysed and evaluated. A platform has already been developed which runs on FPGA and provides interface for the deployment of DMP network nodes [16]. This platform makes stepping stone for hardware realization of schemes presented in this paper.

References

[1] C. Chafe, M. Gurevich, G. Leslie, and S. Tyan, "Effect of time delay on ensemble accuracy," in *Proceedings of the International Symposium on Musical Acoustics*, vol. 31. Citeseer, 2004.

[2] L. Rønningen and A. Lie, "Distributed Multimedia Plays with QoS guaranties over IP," *Wedelmusic, Leeds*, 2003.

[3] L. Rønningen, "A protocol stack for futuristic multimedia," in *Proc. International Conference on Signal Processing and Communications Systems*, 2007.

[4] "Beyond the Boardroom: Innovative Applications of Telepresence," Web Report, Cisco. [Online]. Available: www.cisco.com/web/telepresence/collateral/InnovativeApplicationsOf-Telepresence.pdf

[5] "Telepresence Interoperability Protocol (TIP)," Web Report, Cisco. [Online]. Available: http://www.slideshare.net/imtcorg/4-tip-imtc-tp-workshop-061510

[6] L. Ronningen, M. Panggabean, and O. Tamer, "Toward futuristic near-natural collaborations on Distributed Multimedia Plays architecture," in *Signal Processing and Information Technology (ISSPIT), 2010 IEEE International Symposium on*. IEEE, 2010, pp. 102–107.

[7] L. Rønningen and A. Lie, "Transient Behaviour of an Adaptive Traffic Control Scheme," in *EUNICE workshop, Trondheim*, 2002.

[8] A. Lie, "P-AQM: low-delay max-min fairness streaming of scalable real-time CBR and VBR media," in *Internet and Multimedia Systems and Applications*. ACTA Press, 2007.

[9] Lie, A., "Enhancing Rate Adaptive IP Streaming Media Performance with the use of Active Queue Management," Ph.D. dissertation, Norwegian University of Science and Technology, 2008.

[10] "3DPresence EU Project," Web page. [Online]. Available: http://www.3dpresence.eu/

[11] L. Rønningen, "Input Traffic Shaping," in *Proc. International Teletraffic Congress*, 1982.

[12] J. Turner, "New directions in communications(or which way to the information age?)," *IEEE communications Magazine*, vol. 24, no. 10, pp. 8–15, 1986.

[13] S. Bjornstad, M. Nord, T. Olsen, D. Hjelme, and N. Stol, "Burst, packet and hybrid switching in the optical core network," *TELEKTRONIKK*, vol. 101, no. 2, p. 148, 2005.

[14] L. Rønningen, "DMP Processing Architectures based on FPGA and PCIe," Technical Report, 2011. [Online]. Available: http://www.item.ntnu.no/people/personalpages/fac/leifarne/start

[15] K. Dai, "Performance analysis of the dropping scheme of DMP network nodes," Master's thesis, Norwegian University of Science and Technology, 2008. [Online]. Available: http://ntnu.diva-portal.org/smash/record.jsf?pid=diva2:347710

[16] M. Wielgosz, M. Panggabean, A. Chilwan, and L. Rønningen, "FPGA-Based Platform for Real-Time Internet," in *The Third International Conference on Emerging Security Technologies (EST-2012)*, 2012.