Cross-Layer Optimization for Video-streaming Transmission with QoS over Ad Hoc Networks: A Holistic Approach

Guillermo Díaz Delgado, Víctor Carrascal Frías, and Mónica Aguilar Igartua

Abstract: **In this paper we present ViStA-XL, a Cross-Layer (XL) design aiming to optimize the overall performance of video-streaming services over Mobile Ad Hoc Networks (MANETs). The idea relies on applying optimization strategies to different network layers in a holistic way. In ViStA-XL, a real-time Optimizer (XLO) periodically gathers information of the state of node and network from different layers of the stack of protocols, takes optimization decisions, and then modifies some parameters of the protocols accordingly. In addition, our proposal exploits path diversity through MM-DSR (Multipath Multimedia Dynamic Source Routing) protocol as a means to reinforce the Quality of Service (QoS) provision to multi-layer encoded video-streaming applications, by protecting the most important video information packets, balancing the load and decreasing the end-to-end delay. To show the advantages of our approach, we have developed and tested an algorithm based on ViStA-XL. Simulation results show that our proposed network design can improve the performance of video-streaming transmissions over MANETs in spite of frequent changes in network and node operating conditions.**

Index terms: **Ad Hoc networks, cross-layer design, multipath routing, QoS provision, video-streaming services**

I. INTRODUCTION

Quality of Service (QoS) provision to video-streaming applications over Mobile Ad Hoc Networks (MANETs) poses a challenging problem. On one hand, video-streaming allows the transmission of video files through a network in the form of time continuous flows of data packets (video streams), so the application at the receiver does not need to download the video file before start playing. Instead, it uses a limited-size buffer to temporally store the arriving data to be played almost instantaneously. To achieve this, the network must satisfy the stringent QoS requirements of video-streaming in order to provide a minimum level of quality to the final user.

On the other hand, a MANET is formed by a set of wireless mobile nodes that communicate with each other without any fixed infrastructure or centralized administrative support. Besides, the transmission range of the wireless network interfaces is limited, thus several intermediate nodes may be needed for one host to transfer data to another one in the network. Traditionally, MANETs have been mainly used in military and other tactical applications such as emergency rescue or exploration missions. However, civilian and commercial applications (i.e. conferences, course training, lectures, museum visits, city tours, peer-to-peer applications, e-gaming, etc.) are likely where there is a need for ubiquitous communication services.

Nevertheless, the ability of the mobile nodes of a MANET to move freely produces frequent changes in the network topology. In addition, the radio channel vagaries (e.g. interference, channel multipath effects, fading) and node's energy power limitations may also produce frequent changes in topology and connectivity. Consequently, MANETs should adapt dynamically to continue operating in spite of changes in network conditions [1].

As a result of the dynamic nature of MANETs, it is difficult to provide the QoS required for applications where a best-effort service is not enough (e.g. video-streaming). Actually, traditional QoS management techniques developed for infrastructure-based networks have shown to be inadequate, even if some IntServ and DiffServ techniques can still be applied to manage and control flows through queuing, marking and dropping packets [2]. Therefore, QoS provision in MANETs remains an open issue [3, 4].

We argue that QoS provision does not depend on any single network layer, but on the coordinated efforts of all layers. Thus, we state that, for dynamic networks as MANETs, it is best to develop dynamic solutions based on a cross-layer approach, which take into account the specific characteristics of the network [5, 6]. Moreover, a proper QoS-aware architecture for Ad Hoc networks should make sure the cooperation among all the components related to QoS provision, e.g. signaling, routing and Medium Access Control (MAC) mechanisms.

The rest of the paper is organized as follows. In Section II we present the main ideas about cross-layer design. In Section III we introduce ViStA-XL, our cross-layer design. Section IV

Manuscript received May 3, 2007, revised July 12, 2007, and October 22, 2007. This research was financially supported by the Spanish Projects SECONNET (CICYT-TSI 2005-07293-C02-01) and ITACA (CICYT -TSI 2007-65393-C02-02), as well as graduate scholarships from CONACYT (Mexico), Fundación Carolina (Spain), PROMEP-UAQ (Mexico) and UPC Recerca (Spain).

G. Díaz Delgado, V. Carrascal Frías and M. Aguilar Igartua are with Telematics Engineering Department, Universidad Politécnica de Cataluña, Jordi Girona, 1-3, Barcelona 08034, Spain. G. Díaz Delgado is also with Universidad Autónoma de Querétaro, Mexico (e-mail: gdiaz@entel.upc.edu).

1845-6421/07/7018 © 2007 CCIS

is devoted to MM-DSR routing protocol and to a QoS-provisioning algorithm based on ViStA-XL design. Some simulation results are shown in Section V. Finally, Section VI summarizes the paper, presents some conclusions and foresees the future work.

II. CROSS-LAYER DESIGN

Most modern communication systems are based on a layered network architecture design (e. g. Internet architecture). Some advantages of a layered approach are the reduced design complexity due to well-defined functional entities, the improved maintainability due to the modular nature, and the high degree of flexibility, since layers function independently of each other. Strictly layered network architecture forbids direct communication between nonadjacent layers, and communication between adjacent layers is limited to procedure calls and responses [7].

Cross-layer design, on the contrary, refers to protocol design done by exploiting the dependence between protocol layers to obtain a better system performance. In a cross-layer design approach, information can be shared among layers in both directions, upper to lower layers and lower to upper layers. This information exchange can be used to optimize the overall performance of the system in a holistic way, by adapting the protocols functionalities in the presence of changing networking conditions, for decision processes such as route selection, or as input to algorithms.

Cross-layer approach is more suitable for wireless networks, where time-varying conditions of wireless links present new problems that cannot be handled well by a strictly layered architecture [8]. Additionally, the wireless medium offers new modalities of communication that the layered architectures do not accommodate. Moreover, to deal with more challenging networking environments such as MANETs, where the mobility and energy power limitations of the nodes can produce frequent topology and connectivity changes, cross-layer design has emerged as an alternative to allow the network to adapt dynamically to maintain on-going communications in spite of these changes [9, 10].

Also, because the dynamic nature of MANETs and since QoS provision depends on the coordinated efforts from all layers, cross-layer network design must be applied to MANETs to provide the necessary adaptive QoS support to resource demanding applications, such as multimedia applications, which are sensitive to changing networking conditions.

In [11], authors classify the cross-layer design proposals in literature in four main categories, depending on the way the layers of the network architecture are coupled: (a) creation of new interfaces, (b) merging of adjacent layers, (c) design coupling without new interfaces and (d) vertical calibration across layers. The first approach consists of creating a new interface not available in the layered architecture to permit the information sharing between layers. This approach requires adding extra code to the original participating protocols and defining new headers or methods to access to cross-layer information. In the second approach, the idea is to design two or more adjacent layers together such that the service provided by the new superlayer is the union of the services provided by the constituent layers. This does not require any new interfaces to be created in the stack, because the superlayer can use the interfaces that already exist in the original architecture. The third category involves coupling two or more layers at design time without creating any interfaces for information sharing at runtime, but by designing the involved protocols with reference of each other. The problem with this approach is that it may not be possible to replace one layer without making corresponding changes to another layer.

The fourth approach refers to adjusting parameters that span across layers. This cross-layer design approach is motivated by the idea that the performance seen at the application level is a function of the parameters at all the layers below it. Thus, join tuning of parameters can help to achieve better performance than individual setting of parameters can achieve. Even if vertical calibration can be done in a static manner at design time to optimize some specific metric, it could be more advantageous if it is done dynamically at runtime, emulating a flexible protocol stack that responds to variations in the channel, traffic, and overall network conditions. This requires, however, mechanisms to retrieve and update the values of the parameters being optimized by the different layers [12, 13, 14, 15, 16]. This is the approach that we have followed in our cross-layer design (Fig. 1), due to its advantages regarding the capability of dynamic tunning of the selected parameters at all the layers, which is of major interest in MANETs.

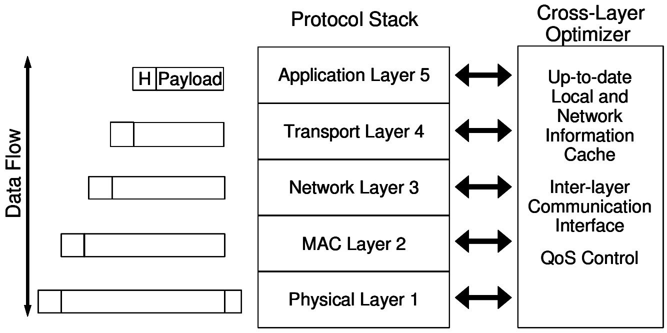


Fig. 1. Vertical calibration approach to cross-layer network design

It is important to remark that most of the research on MANETs turns around the capital problem of providing reliable paths for data transmission from sources to destinations in the hostile environment posed by a multihop scenario, with unreliable wireless links and changing topology and connectivity [1, 9, 17]. Because of this, cross-layer design has been mainly applied taking into account just two or the three lowest layers of the protocol stack shown in Fig. 1 (i.e. Physical, MAC and Network layers) [1, 2, 6, 17]. In general, those proposals provide Best Effort (BE) data delivery service that supports most of the non-interactive services and sometimes provide limited QoS to allow some interactive services (e.g. chatting and very low quality voice and video streaming applications).

For QoS provision to more demanding applications over wireless networks, where the BE data delivery service does not suffice (e.g. video-streaming, audio and video conference, video-gaming), some cross-layer design proposals consider also the Transport and Application layers [12, 13, 14, 15, 16, 17, 18, 19, 20]. Nevertheless, several proposals only take into account two or three layers of the network architecture [1, 2, 5, 6, 17]. Also, some cross-layer design proposals that considers application adaptability to network and node conditions [5, 15, 16, 21] are designed for less complex infrastructure one-hop wireless networks [16].

The cross-layer design proposed in this paper, in contrast with some other cross-layer designs [12, 13, 14], gathers information coming from all the layers of the network architecture and considers application adaptability. This way, the real-time cross-layer optimizer is able to know the actual node and network states for the decision-taking process. In addition, the network takes into account the specific characteristics of video-streaming applications and it is able to exploit them in order to provide a better Quality of Perception (QoP) through the use of unequal error protection techniques.

In order to show the advantages of our proposal, we have developed an implementation of the Cross-Layer Optimizer defined in our proposed cross-layer design (Fig. 1). Indeed, the optimizer selects the highest quality paths that meet the video-streaming communications requirements, changes packet marking policy at the Application/Transmission layers interface, and selects the appropriate scheduling scheme at the Network/MAC layers interface dynamically, based on the information about the state of the network and the node gathered from Physical to Transport layers (Fig. 1).

III. VISTA-XL

ViStA-XL (Cross-Layer design for Video Streaming over Ad Hoc networks) has been developed to provide soft-QoS (i.e. with no strict QoS guarantees) to multi-layer encoded video in MANETs. In ViStA-XL, all the network architecture layers (Physical, MAC, Network, Transport and Application) cooperate with each other to fulfill the task of QoS provision to video-streaming applications by service differentiation (Fig. 2).

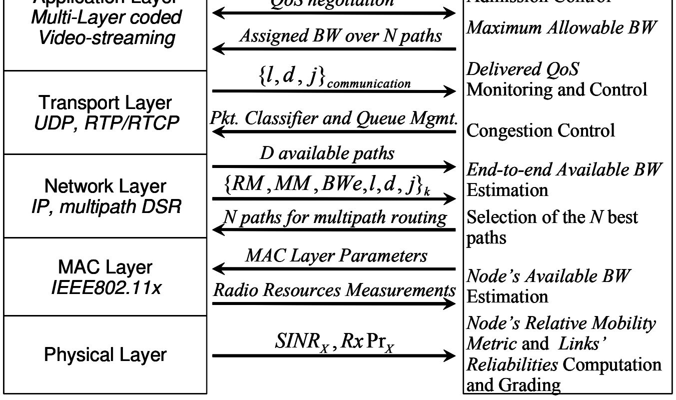


Fig. 2. ViStA-XL architecture overview

Even if our system has been designed having in mind the transmission of hierarchical multi-layer encoded video-streaming, it can actually manage any kind of multi-layer encoded video-streaming, e.g. hierarchical [22], Multiple Description Coding (MDC) [23, 24], Fine Granularity Scalability (FGS) [25, 26], as well as other media-streaming applications. Our framework could work with any of these layered coding based schemes, the only requirement is a scheme capable of manage several substreams, each one of them contributing to a better level of video quality. Those substreams will be transmitted through the different available paths according to the assigned priorities which have been assigned depending on their importance in the decoding process of the video flow in the receiver side.

A. Description of ViStA-XL

The main element of ViStA-XL design is the Cross-Layer Optimizer (XLO). By exploiting the periodically obtained information, the XLO module is in charge of doing the necessary functions to optimize the protocol stack in a global way. To do this, XLO adjusts dynamically several parameters at different protocol layers. These adjustments are done in real-time, so the protocol stack can adapt quickly to changes in network, in environment (presence of obstacles, interference) and in nodes (mobility, available resources).

As it can be seen in Fig. 2, all network architecture layers send information to the XLO module. Thus, in our design, the *Physical Layer* of a node informs XLO about the received signal power ($RxPr_X$) and the signal to interference-plus-noise ratio ($SINR_X$) from each one of its neighbors. *MAC Layer* sends information related to the radio channel usage, such as quality of links to its neighbors, interference level in channel, sent-to-received MAC frames ratio, channel utilization and hidden nodes. *Network Layer* informs to the XLO module about the number (D) of available paths between source and destination nodes, which the node maintains in a cache memory. Network Layer also informs about the quality of each one of those D paths by periodically sending probe messages which return with the following information relative to each path: reliability (RM, Reliability Metric), mobility index of nodes (MM, Mobility Metric), end-to-end available bandwidth (BW_e), percentage of packet losses (l), average packet delay (d) and average delay jitter (j). With that information, the XLO module decides dynamically which N best paths will be used to route the data packets from source to destination until next arrival of information about the quality of the D available paths. By means of RTCP (Real-Time Control Protocol) [27] generated reports, *Transport Layer* informs about the quality metrics for each end-to-end communication: percentage of packet losses, mean packet delay and mean delay jitter. This information helps the XLO module to ask the application to adjust its QoS requirements according to network and node's conditions, if possible. The *Application Layer* sends information to the XLO module about the QoS requirements from the specific application (e.g. multi-layer encoded video-streaming): required bandwidth (BW_{REQ}), minimum user acceptable bandwidth (BW_{MIN}), and maxima packet losses, delay and delay jitter (l_{MAX}, d_{MAX} and j_{MAX}, respectively) that can be accepted by the application. With this information, together with an end-to-end available bandwidth estimation, the XLO module performs the Call Admission Control (CAC) of new communication requests.

One of the main characteristics of ViStA-XL is that it has been conceived for flexible applications that can adapt to dynamic conditions of MANETs and heterogeneity of network nodes. Transmission of multi-layer encoded video allows light nodes, with scarce resources and low profile features (e.g. PDA), as well as to more powerful nodes with more resources (e.g. laptop computers), to be able to access to video-streaming services (e.g. video-on-demand, VoD). Moreover, the flexibility of multi-layer encoded video (see Section III.B) makes it possible to applications to keep alive on-going video communications even in low performance network conditions. This could be done by lowering the quality of the transmitted video, instead of just cutting the service off.

At *Network Layer* we propose a routing algorithm that allows the framework to find and manage multiple paths between a source and a destination. Our multipath algorithm is based on the DSR (Dynamic Source Routing) [28] protocol. By using several paths for video packets transmission, it is possible to obtain the necessary bandwidth to let a video-streaming communication be admitted by the CAC mechanism with a certain level of end-to-end QoS. Also, path diversity allows to unequally protect video information packets (depending on their importance) and to perform load balancing. It is important to note that it is not the multipath algorithm which performs the routes selection to forward packets but the XLO module, based on the knowledge of node and network states (e.g. end-to-end available bandwidth, percentage of packet losses, end-to-end mean packet delay, index of mobility of nodes in a path, path reliability, etc.).

One of the main functions of the XLO module consists of making possible interactions between different layers of the network architecture (interfacing). Thus, for example, packet classification, queuing and scheduling performed at Network and MAC Layers are based on packet marking done at Application Layer. Furthermore, these interactions depend on the assigned bandwidth to each communication and on the number of selected paths.

The design of ViStA-XL is based on the well known and widely used IP (Internet Protocol) [29] at the Network Layer, and UDP (User Datagram Protocol) [30] and RTP/RTCP (Real-Time Protocol/Real-Time Transmission Control Protocol) [27] at the Transport Layer, as well as on the IEEE 802.11 standard at the MAC and Physical Layers [31]. We have also based our design on some radio channel measurement ideas taken from the IEEE 802.11 TGk work [32].

B. Hierarchical multi-layer encoded video

In this work we have considered MPEG-2 hierarchical temporal scalable multi-layer encoded video [33]. MPEG-2 encoded video is formed by GoPs (Groups of Pictures). A GoP is composed by a fixed number of encoded frames and has a defined structure. There are basically three types of frames: I frames (intrapicture), P frames (predicted picture) and B frames (bidirectional predicted picture). I frames can be thought of as a reference frame; they are self-contained and thus carry the most important information of the pictures. P and B frames are not self-contained; they specify relative differences from some reference frame or frames. Actually, P frame specifies the differences from the immediately previous I or P frame, while B frames gives an interpolation between the immediately previous and subsequent I or P frames. As such, there is only one I frame in a GoP and there could be no or more P and B frames. In general, the size of P frames is about 20% the size of I frames, while the size of B frames is only about 10% the size of I frames.

It is important to say that I frames are absolutely necessary to decode the video sequence, and an entire GoP would be lost if we don't have the corresponding I frame at decoding time, even if we have all the P and B frames of that GoP. In the same way, B frames are useless if preceding and following I or P frames are not present at decoding time. On the other hand, GoPs can be decoded even if just I frames are present. Thus, I frames contain the most important video information, while information carried by B frames is the least important one for the decoding process at the receiving side when recovering the video sequence.

Although we know that unequal error protection for IPB frames is not a novel technique, we wanted to consider previous experiences developed by other authors in the area of video coding [26] and include them in our ViStA-XL framework. Also, if in the future we want to upgrade our architecture to offer QoS to another video compression format such as H.264, we will only have to establish a new mapping between the different coded frames and the priorities available in the system [20].

IV. MM-DSR AND QOS-PROVISIONING ALGORITHM

QoS-provision in networks requires the existence of a path with relative reliability from source to destination, in order to maintain packet losses and delays within a predictable range. In addition, some real-time and multimedia applications (e.g. video-streaming), require a minimum end-to-end bandwidth availability. However, in MANETs, a path usually consists of multiple highly unstable wireless links that sometimes are not able to provide the required bandwidth. On the other hand, the broadcast transmission nature of nodes in a MANET makes possible the existence of several simultaneous paths between a source and a destination. Thus, the use of path diversity has been proposed as a solution to QoS-provision in MANETs [34]. Even more, some approaches [35, 36, 37] propose to select totally disjoint paths to avoid congestion in common nodes and links, and to maximize the available bandwidth, but it is not always possible to obtain such variety of paths. Also, a wrong path selection from the available paths can reduce the effectiveness of the path diversity technique [38]. Alternatively, it is possible to look for all the paths from source to destination, disjoint and non-disjoint paths (with one or more intermediate common node), and to select the best routes to forward the packets, according to the QoS required and to the quality of the paths.

There are also some proposals that use multipath routing for QoS-provision together with load balancing [2, 5, 24], but most of them focus only on a single QoS parameter (e.g. bandwidth or delay). Here, we propose a cross-layer algorithm that takes into account several QoS parameters referred to an individual path, such as available bandwidth, delay, delay jitter and packet losses, and heuristically seeks for a set of paths that provides the required level of QoS in a flexible and dynamic way. Besides, our algorithm involves two new path quality parameters, say *Reliability Metric* and *Mobility Metric*, which allow it to select the most suitable set of paths to perform a multipath scheme taking into account the inherent characteristics of the Ad Hoc networks [39]. These two parameters are closely related with the quality of the paths according to the quality of the links which form those paths, specifically their levels of mobility and reliability. Thus, a path whose links are more reliable may be preferable than a shorter path whose links have a lower reliability. In the same way, our scheme will prefer a path whose nodes move less as its duration probably will be longer, instead of a path with briskly nodes which will produce frequent breaks. We have also modified the DSR (Dynamic Source Routing) protocol. The resulting MM-DSR (Multipath Multimedia DSR) looks for all the available paths from source to destination, it is able to manage multiple paths, and applies a dynamic load-balancing scheme. Finally, MM-DSR sends the packets through the paths selected by the cross-layer QoS-provisioning algorithm.

Our cross-layer QoS-provisioning algorithm involves several parameters. First of all, we have the customer's requirements, established by means of a Service Level Agreement (SLA). Such SLA specifies the network's QoS parameters and their necessary values to deliver the committed image quality. These QoS parameters are the minimum expected bandwidth (say BW_{MIN}), the maximum percentage of data losses (say l_{MAX}), the maximum delay (d_{MAX}) and the maximum delay jitter (j_{MAX}):

$$customer\_req \equiv \left\{ BW_{MIN}, l_{MAX}, d_{MAX}, j_{MAX} \right\} \tag{1}$$

The main idea of our algorithm is this: nodes are continuously querying their neighbors in order to get information related to the signal quality of the link and the relative movement of the nodes, as described below. Besides, a *Probe Message* (PM) is sent periodically along all the available paths between source and destination. Each PM collects network information from each one of the nodes belonging to that path. This information is composed of various parameters called the "*path–state*" of a path k:

$$path\text{-}state_k^i \equiv \{BWe_k^i, l_k^i, d_k^i, j_k^i, RM_k^i, MM_k^i\} \tag{2}$$

where i stands for the iteration number of the algorithm and k for the path number. RM_k^i and MM_k^i stand for *Reliability Metric* and *Mobility Metric* respectively, and they are explained in Sections IV.A and IV.B. Once the destination node receives a PM packet, it will wait 2 seconds for all the PM packets sent by the source to arrive. The total number of PM packets that have been sent is specified into the headers of each one of the PM packets, so the destination node knows how many of them have been sent. Those PM packets that do not arrive into that time slot are discarded. Once the destination has all the PM packets that arrive successfully, a *Probe Message Reply* is generated and sent back to the source. The information collected from all the paths is processed by the source, so the system is able to choose the best paths by means of various thresholds for each one of the parameters (as described in Section IV.C). We assume that the network topology remains barely the same between two successive iterations, in order to reach a consistent solution [3]. The values for BWe_k^i are the bottleneck bandwidth values of each path k (i.e. the minimal residual bandwidth of all the nodes in that path) estimated at the i-th iteration.

To compute and update the actual values for l_k^i, d_k^i and j_k^i (i.e. the packet losses, delay and delay jitter of each path k estimated at the i-th iteration) in a continuous way, an Exponentially Weighted Moving Average (EWMA) filter is applied to the $l_k^{i\_sample}, d_k^{i\_sample}, j_k^{i\_sample}$ (i.e. the packet losses, delay and delay jitter sample values gathered by the PM for each path k at the i-th iteration) as follows:

$$v_k^i = (1-\alpha) \cdot v_k^{i-1} + \alpha \cdot v_k^{i\_sample} \tag{3}$$

where v_k^i stands for l_k^i, d_k^i, j_k^i. A low value for the α coefficient (i.e. $\alpha = 0.25$) has been chosen to let l_k^i, d_k^i and j_k^i parameters evolve smoothly. Then, we get a mark for the path by comparing these values with the values of the customer requirements (1), following these equations:

$$\text{IF} \begin{pmatrix} BWe_k^i > 1.2 \cdot BW_{MIN} \\ BW_{MIN} \le BWe_k^i \le 1.2 \cdot BW_{MIN} \\ BWe_k^i < BW_{MIN} \end{pmatrix} \Rightarrow MBW_k^i \equiv \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}$$

$$\text{IF} \begin{pmatrix} v_k^i < 0.4 \cdot v_{MAX} \\ 0.4 \cdot v_{MAX} \le v_k^i \le 0.8 \cdot v_{MAX} \\ v_k^i > 0.8 \cdot v_{MAX} \end{pmatrix} \Rightarrow Mv_k^i \equiv \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix} \tag{4}$$

Finally, RM_k^i and MM_k^i are computed as explained next.

A. Reliability Metric computation for each path k, RM_k^i

We propose to compute a performance measure of each entire path from the measure of the Signal-to-Interference plus Noise Power Ratio (SINR) between consecutive neighbors. For each iteration i and each path k, we obtain the SINR values of each node j with respect to node j-1 within the downstream path from source to destination, and then we assign marks $x_j^{k,i}$ to each node j heuristically:

$$\text{IF}\begin{pmatrix} SINR_j^{k,i} \ \geq \ 25 \text{ dB} \\ 15 \text{ dB} \ \leq \ SINR_j^{k,i} < 25 \text{ dB} \\ 10 \text{ dB} \ \leq SINR_j^{k,i} < 15 \text{ dB} \\ SINR_j^{k,i} < 10 \text{ dB} \end{pmatrix} \Rightarrow x_j^{k,i} = \begin{pmatrix} \text{3, very good link} \\ \text{2, good link} \\ \text{1, regular link} \\ \text{0, bad link} \end{pmatrix} \quad (5)$$

Now, we compute the geometrical mean $\hat{x}^{k,i}$ of the partial links within each path k:

$$\hat{x}^{k,i} = \sqrt[L_k^i]{(\prod_{j=1}^{j=L_k^i} x_j^{k,i})} \quad (6)$$

where L_j^i stands for the number of partial links within each path k.

Finally, we assign heuristic values to the Reliability Metric, RM_k^i, for each available path k as follows:

$$\text{IF}\begin{pmatrix} \hat{x}^{k,i} > 2 \\ 1.5 < \hat{x}^{k,i} \leq 2 \\ 1 < \hat{x}^{k,i} \leq 1.5 \\ \hat{x}^{k,i} \leq 1 \end{pmatrix} \Rightarrow RM_k^i = \begin{pmatrix} \text{3, very good path} \\ \text{2, good path} \\ \text{1, regular path} \\ \text{0, bad path} \end{pmatrix} \quad (7)$$

From (5), if a node j along the path k breaks down, the geometrical mean for that path in the iteration i, $\hat{x}^{k,i}$, will be zero. Thus, the Reliability Metric for that path, RM_k^i, will be also zero.

B. Mobility Metric computation for each path k, MM_k^i

Each node X detects the received signal power $Rx\,\text{Pr}_{Y \to X}$ with respect to its neighbors Y from successive packet transmissions (periodic "Hello" messages). As the signal power is inversely proportional to the distance, we can agree if a node is moving fast or slowly by taking consecutive measures of the signal power. Then, node X computes the relative mobility metric with respect to each neighbor node Y, $M_X^{rel}(Y)$. "Hello" messages are sent once a second to the neighbors of each node X involved in each one of the D paths discovered by our DSR-modified protocol. Once a "Hello" message arrives at a neighbor Y, it takes the value of the received power signal from node X, and a "Hello Reply" message is generated and sent to node X. As done in [40], each local mark is computed at node X as follows:

$$M_X^i = E\left[M_X^{rel}(Y)\right]_{s=1}^m = E\left[\left(10 \cdot \log_{10} \frac{Rx\,\text{Pr}_{Y \to X}^s}{Rx\,\text{Pr}_{Y \to X}^{s-1}}\right)^2\right]_{s=1}^m \quad (8)$$

where m is the number of mobility measures between nodes X and Y within an iteration i. In our case, m equals 10 as "Hello" messages are sent once a second and the period of the routing algorithm has been set to 10 seconds. This way, each node X has a mobility mark computed from the average of m consecutive mobility measures with respect its neighbor Y. A low value for MM_X^i means that node X is almost motionless with respect to its neighbor Y, while a high value indicates that node X is highly mobile. We assign marks as follows:

$$\text{IF}\begin{pmatrix} M_X^i < 0.02 \\ 0.02 \ \leq \ M_X^i < 0.08 \\ 0.08 \ \leq \ M_X^i < 0.5 \\ M_X^i \geq 0.5 \end{pmatrix} \Rightarrow MM_{X,j}^i = \begin{pmatrix} \text{3, motionless node} \\ \text{2, low mobility node} \\ \text{1, high mobility node} \\ \text{0, very high mobility node} \end{pmatrix} \quad (9)$$

where $MM_{X,j}^i$ stands for the relative Mobility Metric of node X with respect to its neighbor Y, the two of which form link j in path k. Then, the *Probe Message* (PM) of the next iteration i collects all the partial mobility measures from each node X regarding its next neighbor Y downstream (i.e. from source to destination) within each available path k. These D available paths had previously been discovered by our DSR-modified protocol. As it was said in Section IV, once the destination has all the PM packets that arrive successfully, a *Probe Message Reply* is generated and sent back to the source. This PM Replay includes all the mobility marks of all nodes j in each path k, i.e $MM_{X,j,k}^i$. Finally, the source computes the Mobility Metric of each path k for the i-th iteration, MM_k^i as follows:

$$MM_k^i = \frac{\sum_{j=1}^{j=L_k^i} MM_{X,j,k}^i}{L_k^i} \quad (10)$$

where L_k^i is the number of links in path k at iteration i, and j stands for each upstream node in that path from source to destination.

It is worth noting that the thresholds in (5), (7) and (9) have been heuristically selected after analyzing many simulations. However, as we state in Section VI in a future work we are going to establish dynamic thresholds instead of static values, in order to take into account the inherent characteristics of the Ad Hoc networks, i.e. the high variations in mobility, the frequent paths breaks.

C. The QoS-provisioning algorithm

First of all, we must check if there are enough available resources to accommodate the stream the system is required to send to the customer during the current iteration i. We know the available bandwidth which remained from the previous iteration of the algorithm, BW_a^{i-1}. In case of having the same video encoded with different qualities, we must seek for the bandwidth required for each video-stream ($BW_{MIN\_required}$) and select the maximum one which does not exceed either the available bandwidth BW_a^{i-1} or the bandwidth required by the customer BW_{MIN}:

$$BW_{MIN\_required} \le \max\left[BW_a^{i-1}, BW_{MIN}\right] \qquad (11)$$

The selected stream requires a minimum bandwidth, which we name BW_u^i. Then, the bandwidth that remains available for the next iteration is:

$$BW_a^i = BW_a^{i-1} - BW_u^i \qquad (12)$$

Obviously the available bandwidth is updated whenever a connection is released as well. From (2) we select the set of valid paths for the current iteration *i*, named $PathSet_i$, that fulfill the customer's requirements expressed in (1):

IF $path-state_k^i$ fulfills

$$\left(BW_{ek} \ge BW_{MIN}\right) \cap \left(p_k \le p_{MAX}\right) \cap \left(d_k \le d_{MAX}\right)$$
$$\cap \left(j_k \le j_{MAX}\right)$$

THEN include $path-state_k^i$ in $PathSet_i$.

Let's remark that, even if finding an optimal path with multiple constraints may be an NP-complete problem if it involves multiple additive metrics (i.e. delay, cost) [4], this is not a problem here due to the characteristics of video-streaming applications. Indeed, video-streaming applications use receiver buffers to temporally store the received video frames before to be decoded, in order to allow some initial delay and to diminish the effects of the delay jitter. Thus, the delay is not a severe constraint as soon as the delay jitter remains stable and reasonably low (some ms or less). Also, because not all the video frames are absolutely necessary to decode a GoP, some packet losses are tolerated even if a whole P or B video frame is lost. Moreover, even some whole GoP losses can be tolerated (by the human eye), if that doesn't occur very often or as a burst. In fact, BW_{MIN} is the most stringent constraint that a path has to meet, because all other QoS constraints (i.e. l_{MAX}, d_{MAX} and j_{MAX}) actually depend on it. Therefore it is possible to relax the other constraints in order to allow the routing protocol to find some paths (if any) that meet the QoS constraints imposed by the actual video-streaming transmission.

Once the set of valid paths $PathSet_i$ (which fulfills the customer's requirements) for the current iteration *i* has been set, the algorithm sorts these paths according to the following rules. We focus on the two new QoS parameters which we have proposed here, i.e. the Reliability Metric (*RM*) and Mobility Metric (*MM*), as they are of major importance to arrange the available paths in an Ad Hoc network where the channel is highly variable and links break frequently due to the mobility of the nodes. In this work, we give the same importance to both parameters so they have the same weight and we just add them.

- The $PathSet_i$ is arranged as the addition $RM_k^i + MM_k^i$ (reliability plus mobility) decreases, for each path *k*.
- If there are any coincidences, sort coincident paths as MBW_k^i (bandwidth) decreases.
- If there are any remaining coincidences, sort coincident paths as $Mj_k^i + Ml_k^i$ (delay jitter plus data losses) decreases.
- Finally, if still there are any coincidences, sort coincident paths as the delay of the paths Md_k^i decreases.
- Select the first *N* paths from the sorted $PathSet_i$.

With the *N* best paths selected, a multipath scheme must be applied to the multi-layer encoded video-stream in order to achieve the end-to-end QoS requirements (1). Because not all the video frames have the same importance, the corresponding video packets do not have the same treatment. Actually, our MM-DSR manages different queues for different priority packets, looking for cooperation between layers, so the effort done at the upper layers will not be lost at the lower ones. With our architecture, it is easier to provide QoS by service differentiation.

Regarding an MPEG-2 video-streaming service, the system distinguishes the different I, P and B frames of each video-streaming session, giving different priorities to them according to their importance in the decoding process of the received video streams. For instance, if we have selected a 3-paths scheme between source and destination (see Fig. 3), I frames will be sent for the best path with maximum priority (as I frames are the most significant video frames), and P and B frames will be sent by the other two paths respectively giving them lower priorities.

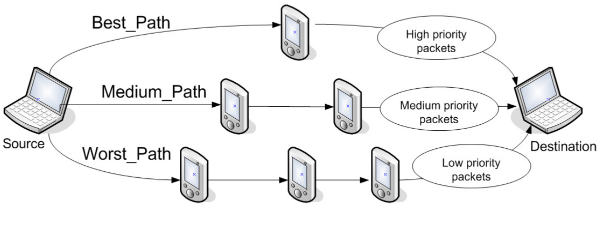


Fig. 3. Multipath routing scheme with 3 disjoint paths

A novelty of this proposal, is that it is able to offer a minimum level of QoS to multiple multimedia transmission sessions sharing the same paths (or a part of them) simultaneously in a MANET.

V. SIMULATION RESULTS

In this section we present simulation results obtained with a cross-layer QoS-aware algorithm based on ViStA-XL design principles. The developed algorithm uses the MM-DSR routing protocol described in Section IV, in order to provide multiple routes between source and destination [39]. All simulations have been carried out using the NS-2 v2.27

simulator [41], over which we have implemented the MM-DSR protocol and our cross-layer QoS-provisioning algorithm.

We have carried out two series of simulation experiments. In both of them we have transmitted the same video sequence with main parameters shown in Table I.

TABLE I
TRANSMITTED VIDEO SEQUENCE

Video sequence	Blade Runner (100 s)
Video format	YUV 4:2:0, CIF, 25 fps
Video encoding	Temporal scalable hierarchical MPEG-2
GoP format	15 f/GoP, (4P, 2B), "IBBPBBPBBPBBPBB"
Traffic type	VBR at 1.6 Mbps$^{(a)}$; 666 Kbps$^{(b)}$

(a) First series of simulations. (b) Second series of simulations

A. First series of simulations: one video-streaming communication over different multipath schemes

In the first series of simulations there is only a single video-streaming communication over the Ad Hoc network. The main objectives of these simulations are to determine the benefits and drawbacks of several multipath schemes, as well as to measure the effects of the unequal protection capability provided by the ViStA-XL design with the MM-DSR protocol. Table II summarizes the simulation settings. We need to mention here that, in order to avoid the convergence problems of the Random Waypoint (RWP) mobility model pointed out in [42, 43, 44], the mobility scenarios were created by using the BonnMotion v1.3a software [45], which allows us to avoid the transitory effects of the RWP mobility model. For instance, if you need a 100 s long scenario, BonnMotion creates a 3700 s long scenario and cuts the first 3600 s. In this work, priorities have been considered. This way, packets are treated with different priorities at the network and MAC layers, according to the importance of each type of frame. At the different multipath schemes, the higher priority packets are transmitted through the best paths between source and destination, while lower priority packets are not allowed to be transmitted through these paths. This is done to avoid flooding the best paths with low priority packets, and to augment the probability of successful transmissions of the more important frames (i.e. I frames).

Extensive simulations have been carried out in order to show the benefits of our approach to DiffServ QoS provision, like the received-to-sent video frames ratio.

Five scenarios have been simulated, each one using a different multipath scheme (Fig. 4). We have set high priority to I frames, medium priority to P frames and low priority to B frames. For example, if a multipath scheme with two paths (N=2) has been considered, as (b) scheme shows in Fig. 4, control packets and I frames would be sent through the best path managed according to a PQ (Priority Queue) scheduler which gives higher priority to control packets. Packets which transport P and B frames have higher priority than Best Effort (BE) and they are sent through the worst path. The rest of the schemes in Fig. 4 display the management options set for multipath schemes of N=1, 2, 3, 4 and 5 paths. The options show how the different packets are sent through the different paths depending on their priority and the quality of the paths.

TABLE II
SIMULATIONS SETTINGS

Simulator	NS-2 v2.27
Simulation area	500 m x 500 m$^{(a)}$; 200x200$^{(b)}$
Number of nodes	100$^{(a)}$; 30$^{(b)}$
Speed of the nodes	0 to 10 m/s$^{(a)}$; 0 to 5 m/s$^{(b)}$
Mobility model	Random Waypoint
Transmission range	120 m
MAC Transmission rate	11 Mbps
UDP Packet size	1460 Bytes
Simulation time	100 s$^{(a)}$; 40 s$^{(b)}$
Number of runs per multipath scheme	5

(a) First series of simulations. (b) Second series of simulations

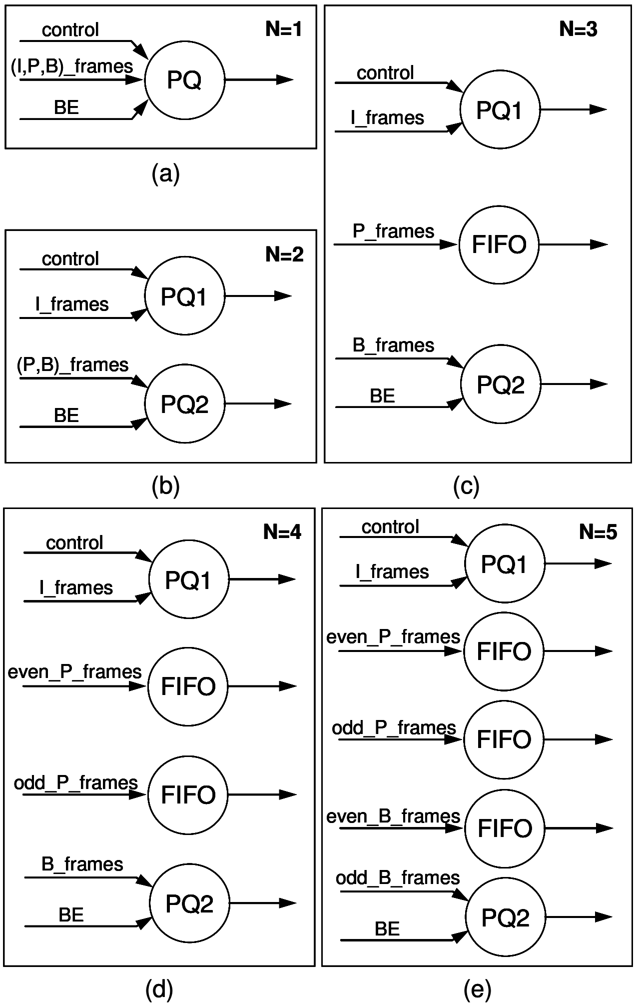
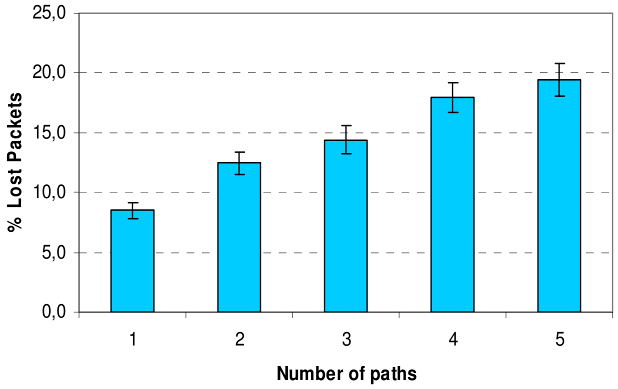
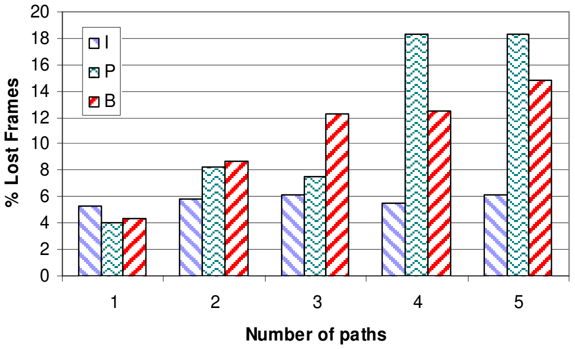


Fig. 4. Simulated scenarios for N=1, 2, 3, 4 and 5 paths

Fig. 5 shows the average percentage of packet losses and video frames losses for each one of the simulated scenarios, i.e. multipath scenarios with 1 to 5 available paths.



(a)



(b)

Fig. 5. Losses percentages for *N*=1, 2, 3, 4 and 5 paths: (a) packet loss and (b) frames losses

In Fig 5a we can see the packet losses behavior as the number of available paths in the multipath scheme increases. We also show the 99% confidence interval for these values, where five simulations per multipath scenario have been carried out. As it can be observed, the packet losses increase as the number of used paths increases. This is due to the combination of several factors. First of all, as *N* grows, less quality paths are used to transmit the packets from all P and B frames (which counts for 93.3% of all transmitted video frames), and those paths have higher probabilities of becoming broken routes by the end of each algorithm iteration (when there would be another chance to select different paths). Therefore, there is a higher probability to lose the low priority packets we sent through these routes. However, it is worth performing a multipath scheme due to the following reasons. Basically, packet losses increase as we take into account more paths, i.e. worse paths. Nevertheless, less important frames (i.e. P and B frames) are the ones sent through the worse available paths, and we sent the more important ones (i.e. I frames) through the best paths. However, it is also worth looking into the frame losses in addition of the packet losses. We can see in Fig. 5b that losses for I frames remain low and stable as the number of paths in the multipath scheme grows, while P and B frames are the ones which experiment higher losses. Thus as we assign the best path only for I frames, we can serve more users with a higher quality. Otherwise, in a unipath scheme all the IPB frames would be sent through the same path (usually the shorter one) and throughputs would decrease. Besides, we apply a load balancing scheme as the traffic of a video stream session is transmitted through several paths instead of through only one, therefore resources are used more efficiently.

It is important to remind that I frames are bigger than P and B frames, and thus they have a higher probability to be fragmented in more packets. Besides, once a packet of a fragmented video frame is lost, the entire frame will be found as lost at the receiver side. So it is worth protecting the I frames due to their importance within the GoP. In Fig. 5b we can see that the I frame losses are almost the same for all the five multipath schemes proposed, thus our algorithm provides adequate protection to I frames by sending them through the best path. It is important to notice that in the case of only one path being used by a user (i.e. *N*=1), I frames suffer slightly more losses than P and B frames. This is due to the nature of I frames, which, as we have said it before, are much longer than P or B frames, and thus they must be fragmented in more packets. However, it can be seen in Fig. 5b that, when using multipath schemes (i.e. *N*>1), the percentage of losses of I frames remains almost constant independently of the number of paths. This way, we achieve a higher equivalent bandwidth by including a multipath scheme with different priorities in the paths.

Setting higher priority to I frames has proved to be a good choice, as they are absolutely needed for the decoding process of a GoP. Thus, the user will notice a higher video-quality as I frames are closely related to the subjective quality. Also, it can be seen that lower priority frames (P and B frames) have increased their losses when applying the proposed multipath schemes, as it was expected because they are transmitted through worse quality paths. It is interesting to observe, however, the important increase of the percentage of P frames lost in the proposed multipath schemes for *N*=4 and 5 paths (schemes d and e in Fig. 4). This is mainly due to the fact that P frames are bigger than B frames, and so they have a higher probability to be fragmented in more packets. In addition, packets of some P frames (odd P frames) are transmitted through a worse path than packets from other P frames (even P frames). By doing this, we are increasing the probability to loss P frames packets, and thus we are increasing the probability to loss entire P frames. Actually, the proposed schemes with *N*=4 and 5 paths show a bad protection policy of P frames (which actually are also quite important in the decoding process of a GoP). Looking to improve the user-level perceived quality of the transmitted video, it would be necessary to define a better protection policy of P frames when having more than 3 paths in a multipath scheme.

In Fig. 6 we show the mean delay jitter obtained for the five simulated schemes. As we can see, the delay jitter decreases when using 2 and 3 paths schemes with respect to the 1 path scheme, while it increases considerably when using 4 and 5 paths schemes. We also show the 99% confidence interval after having run five simulations per scheme.

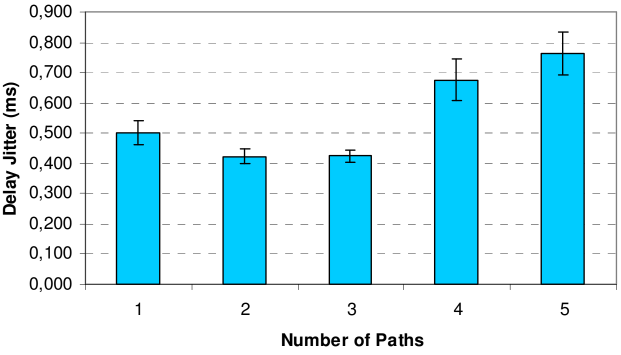
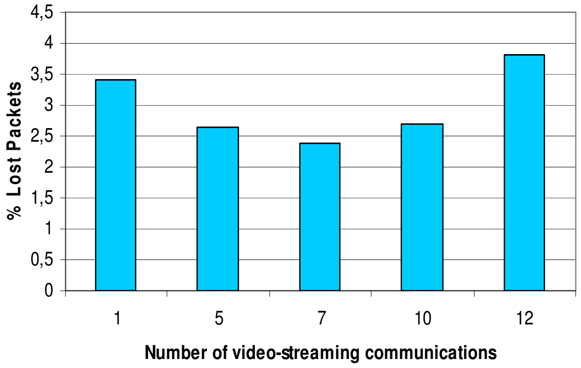


Fig. 6. Delay jitter for N=1, 2, 3, 4 and 5 paths

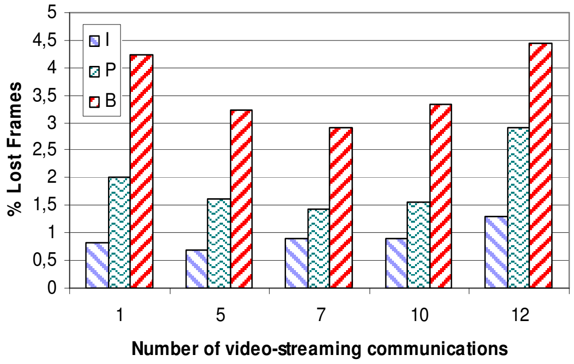
All things considered, we can say that it is not worth using more than three paths in order to achieve a good compromise between video-frames protection and the QoS provided. Thus, from the obtained results from this first series of simulations, we believe that the most appropriate multipath scheme is the one with N=3 paths (Fig. 4c).

B. Second series of simulations: several video-streaming communications over a multipath scheme with three paths

For the second series of experiments, we have fixed the N=3 paths option in the multipath scheme (Fig. 4c) and we have varied the number of simultaneous video-streaming communications from 1 to 12 between different source and destination nodes. In these simulations, we have reduced the simulation area to 200 x 200 m and the number of nodes to 30, in order to force high connectivity scenarios (several paths between any source and destination) with a high probability of communications sharing links or paths. Also, we have reduced the velocity of nodes to 0-5 m/s and the simulation time to 40 s. Again, each simulation scenario has been repeated 5 times. The obtained results are shown in Figs. 7 and 8. As it can be seen from Fig. 7a, the percentage of lost packets remains barely the same for 1 to 10 simultaneous communications (around 3 %), while it increases lightly (to almost 4%) for 12 sources. In Fig. 7b, we can see that the percentages of frame losses per type of video frame follows the same behavior. We explain this by the fact that, in our framework, when using a multipath scheme only the I frames are sent through the best path, while P and B frames are sent through other worse paths. Thus, the multipath scheme exploits the benefits of load-balancing (i.e. achieving higher equivalent rates, using the available resources more efficiently and decreasing the end-to-end delays) in order not to saturate the best path with packets from P and B frames. Therefore, other video-streaming communications could use the better paths (or part of them) to send their high priority packets. Load balancing is certainly important in MANETs due to the dynamic and limited available resources, as bandwidth or remaining battery.



(a)

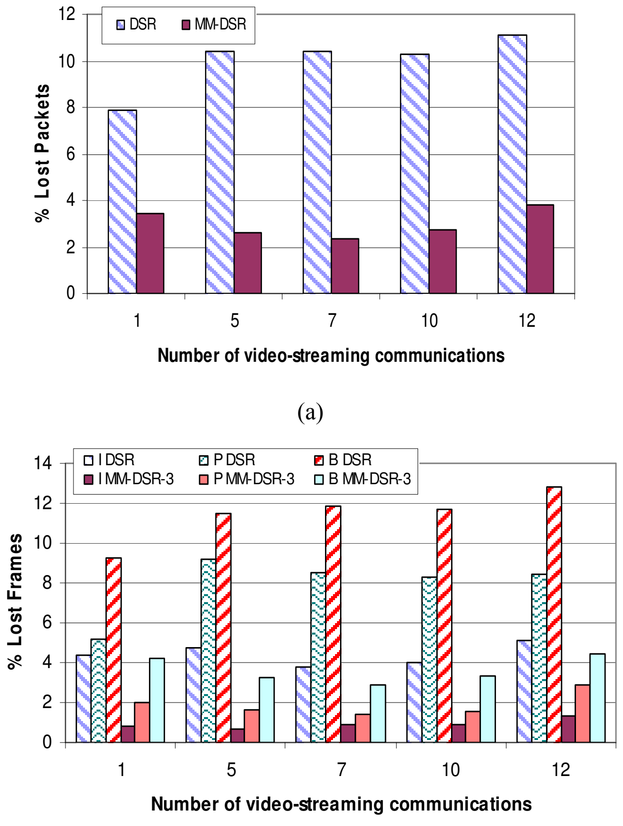


(b)

Fig. 7. Losses percentages for N=3 paths and C=1, 5, 7, 10 and 12 simultaneous video-streaming communications between different source and destination nodes: (a) packet loss and (b) frames losses

In Fig. 8 we compare the results obtained by applying our algorithm with MM-DSR (dense colored bars) for a multipath scheme with 3 paths, versus the results obtained by using the original version of DSR (slashed blue bars) which uses a single path (the shortest path). In both cases we have applied our QoS-provisioning algorithm. It can be seen that the combination of our QoS-provisioning algorithm with MM-DSR for a 3 paths scheme performs much better than the legacy DSR. For example, as we can see in Fig. 8a, the total percentage of lost packets obtained using the MM-DSR algorithm is less than 45% that of the legacy DSR in the worst case shown (one video-streaming communication).

From the results shown in Fig. 8b, we can say that in all the cases our QoS-provisioning algorithm protects very well the most important video frames, i.e. the I frames. Besides, the load distribution through the multiple paths provided by MM-DSR shows to be effective in reducing the percentage of lost video frames. For example, in the case of just one video streaming communication happening, the I frames losses for MM-DSR are less than 20% that of DSR, while P frames losses are less than 40% and B frames losses less than 50%. In the case of 12 simultaneous video-streaming communications over the simulated MANET, the I frames losses for MM-DSR are just about 25% those of DSR, while P and B frames losses are both less than 35% those of DSR.



(a)

(b)

Fig. 8. Performance comparison of our QoS-provisioning cross-layer algorithm with MM-DSR for *N*=3 paths vs. DSR. There are C=1, 5, 7, 10 and 12 simultaneous video-streaming communications between different source and destination nodes

Taking into consideration the results shown in Figs. 5 to 8, we can say that, even if packet losses increase by increasing the number of available paths in the multipath routing scheme (i.e. the additional available paths are worse), the multipath scheme with priorities assures that I frames will be protected as they are sent over the best available path, and this allows the system to achieve a higher equivalent bandwidth available to support more video-streaming communications between users.

To summarize, using the proposed system with different multipath schemes and considering different priorities for the video frames, the performance of video-streaming applications improves with respect to the case of having only one available path, as the usual single-path routing algorithms provide. When there are several users in the network sharing the same paths, our scheme protects the main I frames of the video stream by sending them through the best available path. This framework assists to transmit video-streaming over MANETs, by applying load balancing and thus decreasing the end-to-end delay and increasing the overall throughput.

VI. CONCLUSIONS AND FUTURE WORK

In this work, we have presented ViStA-XL, a cross-layer network architecture design for QoS-provisioning to video-streaming applications over MANETs. Based on ViStA-XL principles that we exposed, we have developed and tested by simulation a cross-layer QoS-provision algorithm that supports multipath routing schemes for video-streaming applications over Ad Hoc networks. This way, our design is also able to provide load-balancing and unequal protection to different types of video substreams. This approach allows to use the available network resources more efficiently, which is certainly important in this type of networks. Besides, several nodes can share the best paths to send their most important packets, improving the final user-level QoS. In a first stage, we have developed the multipath scheme and analyzed the performance of a video-streaming service when there is a single connection, in order to measure the effect of the unequal protection capability of the proposal. Then, we have evaluated the performance of the system for more than one video-streaming communications happening at the same time, in order to measure the benefits of the load-balancing technique of path diversity over the user-level degree of video perception.

From the obtained results, we can say that ViStA-XL seems to be an appropriate cross-layer design to provide for QoS to video-streaming applications over a MANET. Also, the multipath routing MM-DSR algorithm proposed showed to have a better performance than DSR. In fact, by using a multipath scheme, the total available bandwidth between source and destination increases and load balancing is possible too. In addition, as MM-DSR identifies different quality paths, unequal error protection and load balancing is provided by sending the most important video information through the highest quality paths. This way, more video-streaming transmissions can happen simultaneously with a better QoS.

As future work, we are considering the option of working with relative thresholds values in the equations of the algorithm, instead of absolute values. This way, the different parameters involved in the algorithm would vary dynamically depending on the network evolution, taking into account the mobility of the scenario, and the number of paths between source and destination. Also, we are planning to implement a Proportional Differentiation (PD) approach [46] to guarantee proportional QoS between different classes of services.

As we mentioned before, we base our ViStA-XL design on some IEEE 802.11x standards and drafts. However, until now, we have been working with the IEEE 802.11b standard. In a future work we will have some work based on IEEE 802.11e and the IEEE 802.11k draft proposal.

Finally, we devise to evaluate the benefits of introducing redundancy to protect some video packets, looking for increasing the probability of data delivery to improve the subjective video quality [47].

REFERENCES

[1] I. Chlamtac, M. Conti and J. J.-N. Liu, “Mobile Ad Hoc networking: imperatives and challenges”, Ad Hoc Networks, Elsevier, vol. 1, no. 1, pp. 13-64, July, 2003.

[2] X. Hannan, C. K. Chaing and S. K. G. Winston, “Quality of Service Models for Ad Hoc Wireless Networks”, in M. Ilyas

(Ed.), Handbook of Ad Hoc Networks, CRC Press, New York, 2003 (Chapter 28).

[3] S. Chakrabarti, A. Mishra, “Quality of service challenges for wireless mobile Ad Hoc networks”, Wireless Communications and Mobile Computing, John Wiley & Sons, vol. 4, no. 2, pp. 129–153, March, 2004.

[4] T. Bheemarjuna Reddy, I. Karthigeyan, B. S. Manoj, C. Siva Ram Murthy, “Quality of Service provisioning in Ad Hoc wireless networks: a survey of issues and solutions”, Ad Hoc Networks, Elsevier, vol. 4, no. 1, pp. 83-124, January, 2006.

[5] N. Nikaein and C. Bonnet, “A Glance at Quality of Service Models for Mobile Ad Hoc Networks”, in proceedings of 16th Conference De Nouvelles Architectures pour les Communications (DNAC 2002), Paris, December, 2002.

[6] S. Sivavakeesar and G. Pavlou, “Quality of Service Aware MAC Based on IEEE 802.11 for Multihop Ad-Hoc Networks”, in proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2004), vol. 3, pp. 1482-1487, Atlanta, March, 2004.

[7] L. Peterson and B. Davie, Computer Networks: A Systems Approach, third edition, Morgan Kaufmann Publishers, San Francisco, CA, pp. 19-30, 2003.

[8] J. A. Stine, “Cross-Layer Design of MANETs: The Only Option”, in proceedings of IEEE Military Communications Conference (MILCOM’06), October, 2006.

[9] M. Conti and S. Giordano, “Multihop Ad Hoc Networking: The Theory”, IEEE Communications Magazine, vol. 45, no. 4, pp. 78-86, April, 2007.

[10] M. Conti and S. Giordano, “Multihop Ad Hoc Networking: The Reality”, IEEE Communications Magazine, vol. 45, no. 4, pp. 88-95, April, 2007.

[11] V. Srivastava and M. Motani, “Cross-Layer Design: A Survey and the Road Ahead”, IEEE Communications Magazine, vol. 43, no. 12, pp. 112-119, December, 2005.

[12] S. V. Adve et al., “The Illinois GRACE Project: Global Resource Adaptation through Cooperation” in proceedings of Workshop on Self-Healing, Adaptive, and self-MANaged Systems (SHAMAN) (held in conjunction with the 16th Annual ACM International Conference on Supercomputing), June, 2002.

[13] M. Conti, G. Maselli, G. Turi, and S. Giordano, “Cross Layering in Mobile Ad Hoc Network Design”, IEEE Computer, vol. 37, no. 2, pp. 48–51, February. 2004.

[14] R. Winter, J. H. Schiller, N. Nikaein and C. Bonnet, “CrossTalk: Cross-Layer Decision Support Based on Global Knowledge”, IEEE Communications Magazine, vol. 44, no. 1, pp. 93-99, January, 2006.

[15] W. He and K. Nahrstedt, “Impact of Upper Layer Adaptation on End-to-end Delay Management in Wireless Ad Hoc Networks”, in proceedings of 12th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS’06), April, 2006.

[16] S. Khan, M. Sgroi, Y. Peng, E. Steinbach, and W. Kellerer, “Application-driven cross-layer optimization for video streaming over wireless networks”, IEEE Communications Magazine, vol. 44, no. 1, pp. 122-130, January, 2006.

[17] P. Mohapatra, J. Li, and C. Gui, “QoS in Mobile Ad Hoc Networks”, IEEE Wireless Communications, Special Issue on QoS in Next-Generation Wireless Multimedia Communications Systems, vol. 10, no. 3, pp. 44-52, June, 2003.

[18] K. Chandran et al., “A Feedback-Based Scheme for Improving TCP Performance in Ad Hoc Wireless Networks”, IEEE Personal Communications, vol. 8, no. 1, pp. 34-39, February, 2001.

[19] G. Holland and N. Vaidya, “Analysis of TCP Performance over Mobile Ad Hoc Networks”, ACM MobiCom, Seattle, WA, August, 1999.

[20] A. Ksentini, M. Naimi and A. Guéroui, “Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture”, IEEE Communications Magazine, vol. 44, no. 1, pp. 107-114, January, 2006.

[21] D. Thomson, N. Schult, and M. Mirhakkak, “Dynamic Quality-of-Service for Mobile Ad Hoc Networks”, MobiHoc 2000, Boston, MA, 2000.

[22] C.-H. Wu and J. D. Irwin, “Multimedia and multimedia communication: a tutorial”, IEEE Transactions on Industrial Electronics, vol. 45, no. 1, pp. 4-14, February, 1998.

[23] Y. Wang, S. Wenger, W. Jiantao and A. K. Katsaggelos, “Error Resilient Video Coding Techniques”, IEEE Signal Processing Magazine, vol. 17, no. 14, pp. 61-82, July, 2000.

[24] S. Mao, Y. Thomas Hou, X. Cheng, H. D. Sherali and S. F. Midkiff, “Multipath Routing for Multiple Description Video in Wireless Ad Hoc Networks”, in proceedings of IEEE INFOCOM 2005, vol. 1, pp. 740-750, Miami, March, 2005.

[25] W. Li, “Overview of Fine Granularity Scalability in MPEG-4 Video Standard”, IEEE Transactions on Circuits and Systems for Video Technology, vol.11, no. 3, pp. 301-317, March, 2001.

[26] M. Van der Schaar and H. Radha, “Unequal packet Loss Resilience for Fine Granularity Scalability Video”, IEEE Transactions on Multimedia, vol. 3, no. 4, pp. 381-394, December, 2001.

[27] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, RTP: A Transport Protocol for Real-Time Applications, IETF Standard Track STD 64, RFC 3550, July, 2003.

[28] D. Johnson, Y. Hu and D. Maltz, The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4. IETF Experimental RFC 4728, February, 2007.

[29] J. Postel, Internet Protocol, IETF STD 5, RFC 791, September, 1981.

[30] J. Postel, User Datagram Protocol, IETF STD 6, RFC 768, August, 1980.

[31] IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999), “IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, The Institute of Electrical and Electronics Engineers, Inc. (IEEE), New York, USA, pp. C1-1184, June, 2007.

[32] IEEE 802.11 WG “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Specification for Radio Resources Measurement, Amendment to IEEE 802.11 Standard”, IEEE 802.11k/D3.0, New York USA, The Institute of Electrical and Electronics Engineers, Inc. (IEEE), 2005.

[33] P. N. Tudor, “MPEG-2 video compression”, Electronics & Communication Engineering Journal, vol. 7, no. 6, pp. 257-264, December, 1995.

[34] S. Mao, D. Bushmitch, S. Narayanan and S. S. Panwar, “MRTP: A Multiflow Real- Time Transport Protocol for Ad Hoc Networks”, IEEE Transactions on Multimedia, vol. 8, no. 2, pp. 356-369, April, 2006.

[35] R. Leung, J. Liu and E. Poon, A. C. Chan and B. Li, “MP-DSR: A QoS-aware Multi-path Dynamic Source Routing Protocol for

Wireless Ad-Hoc Networks", in proceedings of 26th Annual IEEE Conference on Local Computer Networks (LNC'01), pp. 132-141, Tampa, November, 2001.

[36] S. J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in Ad Hoc networks", in proceedings of IEEE International Conference on Communications (ICC'01), vol. 10, pp. 3201-3205, Helsinki, June, 2001.

[37] S. Mao, S. Lin, S. S. Panwar and Y. Wang, "Reliable transmission of video over Ad Hoc networks using automatic repeat request and multi path transport", in proceedings of the 2001-Fall IEEE Semiannual Vehicular Technology Conference (VTC 2001-Fall), pp. 615-619, Atlantic City, October, 2001.

[38] Y.-H. Wang, H.-Z. Lin and S.-M. Chang, "Interference on Multipath QoS Routing for Ad Hoc Wireless Networks", in proceedings of 24th International Conference on Distributed Computing Systems Workshops – W1: MNSA (ICDCSW'04), pp. 104-109, Tokyo, March, 2004.

[39] V. Carrascal Frías, G. Díaz Delgado, M. Aguilar Igartua, J. Alins Delgado and J. Mata Díaz, "QoS Provision for Video-Streaming Applications over Ad Hoc Networks", in proceedings of EUROCON 2005: Computer as a Tool, pp. 640-643, Belgrade, November, 2005.

[40] P. Basu, N. Khan and T. D. C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks", in proceedings of 21st International Conference on Distributed Computing Systems Workshops (ICDCSW'01), pp. 413-418, Arizona, April, 2001.

[41] Network Simulator - ns-2, http://www.isi.edu/nsnam/ns/

[42] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless Ad Hoc networks", IEEE Transactions on Mobile Computing, vol. 2, no. 3, pp. 257-269, July-September, 2003.

[43] D.M. Blough, G. Resta, P. Santi, "A statistical analysis of the long-run node spatial distribution in mobile Ad Hoc networks", in proceedings of the ACM MSWiM 02, Atlanta, GA, September 2002.

[44] J. Yoon, M. Liu, B. Noble, "Random waypoint considered harmful", in proceedings of INFOCOM 2003, San Francisco, April 2003.

[45] BonnMotion - a mobility scenario generation and analysis tool, Communication Systems Group, Institute of Computer Science IV, University of Bonn, Germany, http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/

[46] O. Salem, and A. Benzekri, "Towards End-to-End QoS in Ad Hoc Networks", in proceedings of 3rd IFIP Annual Conference on Wireless On demand Network Systems and Services (WONS), Les Ménuires, France, January, 2006.

[47] Y. Yuan, B. Cockburn, T. Sikora, and M. M., "A GoP Based FEC Technique for Packet Based Video Streaming", in proceedings of 10th WSEAS International Conference on Communications, pp. 187-192, Athens, Greece, July, 2006.

Guillermo Díaz Delgado received a B.Sc. degree on Electronic Systems from the Monterrey Institute of Technology and Higher Studies (ITESM), Mexico, and a M.Sc. degree from Laval University, Quebec, Canada. He has served as engineer at several companies and has been lecturer and researcher at different public and private institutions. Since 1991, Mr. Díaz Delgado is with the Faculty of Informatics of the Queretaro State University. His research interests include: QoS management, network modelling and performance evaluation, multimedia services over IP networks, optimal resources allocation, media streaming transmission with QoS, Ad Hoc and sensor networking, and cross-layer design for QoS-provision. Presently, he works on his Ph.D. thesis on Telematics Engineering at the Telematics Engineering Department of the Technical University of Catalonia (UPC), Barcelona, Spain. He has financial support from PROMEP-UAQ (Mexico), CONACYT-UPC (Mexico-Spain) and Fundación Carolina (Spain). He is member of IEEE, IEEE Communications Society, and IEEE Computer Society.

Víctor Carrascal Frías was born in Barcelona (Spain) in 1978. He received his Degree in Telecommunication Engineering from UPC in 2003. During the period 2002-2004 he was working at Tradia at the Telecontrol Department, doing some developments and demonstrations of the capabilities of the data transmission of TETRA over IP. Also, he was working as Project Manager in different projects for the METEOCAT department and ACA. In 2004 he joined the "Distributed Multimedia Services with QoS (Quality of Service) Group" in the Telematics Services Research Group - SERTEL (http://sertel.upc.es) at the Department of Telematics Engineering of the UPC (http://www-entel.upc.es). He is interested in research about QoS over video streaming on IP networks, QoS general protocols, specially routing ad-hoc protocols, including rate control protocols. He is currently working in some implementations for the NS2 simulator in order to develop a cross-layer algorithm which must be able to give certain QoS over adhoc networks. He is currently researching in the MDC(Multiple Description Coding) applications for video transport in multipath environments. He has done an implementation of the RTP/RTCP protocol for the well known NS2 Simulator, and he is working in how to applicate an modified implementation of the DSR protocol (Dynamic Source Routing) for multipath environments.

Mónica Aguilar Igartua received the Telecommunication Engineering degree and the Ph.D. in Telecommunications Engineering in 1995 and 2000 respectively, both from the Polytechnic University of Catalonia (UPC), Barcelona, Spain. Currently, she is working on service-level QoS management frameworks, focusing on dynamic procedures to compute Service Level Agreements (SLA) parameters. Her research interests also include modeling and performance evaluation of broadband network nodes, as well as of multimedia servers and video streaming. Her current research activity is focused on the provision and management of QoS over Mobile Ad Hoc Networks (MANETs). She is an associate professor in the Department of Telematics Engineering at the ETSETB. In 1995, she joined the High-Speed Systems and Networks Design and Evaluation Group, and in 2001 she joined the Telematics Services Research Group (SerTel) at the Department of Telematics Engineering of the UPC.