Research Article

Enhancing Mobile Military Surveillance Based on Video Streaming by Employing Software Defined Networks

Iulislol Zacarias ¹, **Janaína Schwarzrock** ¹, **Luciano P. Gaspary** ¹, **Andersonn Kohl** ²,
Ricardo Q. A. Fernandes², **Jorgito M. Stocchero** ², and **Edison P. de Freitas** ¹

¹Informatics Institute, Federal University of Rio Grande do Sul, Porto Alegre, RS 91509–900, Brazil

²Brazilian Army, Brasília, DF 70630–901, Brazil

Correspondence should be addressed to Iulislol Zacarias; izacarias@inf.ufrgs.br

Received 28 June 2018; Accepted 10 September 2018; Published 2 October 2018

Academic Editor: Pierre-Martin Tardif

Copyright © 2018 Iulislol Zacarias et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Situation awareness in surveillance systems benefits from high-quality video streaming service. This is even more important considering military systems, in which delays in image transmission may have a significant impact on the decision-making process. However, in order to deliver high-quality video streaming service, the required network infrastructure may be prohibitively complex, or even completely impossible to deploy, if mobile data providers are considered. Moreover, the demand for high network throughput poses extra requirements on the network. Considering this context, this paper addresses the problem of highly mobile networks composed of unmanned aerial vehicles (UAVs) as data providers of a military surveillance system. The proposed approach to tackle the problem is based on a Software Defined Networking (SDN) approach aiming at providing the best routes to deliver the data, enhancing the end-user quality of experience. An extensive experimental campaign was performed by means of simulations and the acquired results provide solid evidence of the usefulness of this proposal.

1. Introduction

Due to advances in the manufacturing process, component miniaturization, and continuous cost reductions, unmanned aerial vehicles (UAVs) became a widely used technology nowadays. With the UAVs readily available to the public, new applications have emerged, in many fields, such as precision agriculture, remote sensing, weather monitoring, support for communication networks in disaster scenarios, wireless coverage expansion, and even delivery of goods [1]. In addition to the use in civilian applications, UAVs are still widely employed in military applications, with early uses dating from 25 years ago. Typical examples of military applications are border surveillance, ground reconnaissance, and offensive missions [2].

Modern military systems make use of advanced imaging and video resources to carry surveillance, reconnaissance, and information gathering missions [3]. The media transmitted through the wireless network must comply with Quality of Service (QoS) requirements to ensure acceptable video quality from the user perspective. If those requirements

are not met, the video streaming may only contribute to network jamming [4]. Moreover, connections used by video streaming applications must comply with rigorous requirements concerning latency, latency variation (jitter), and throughput. Although the requirements for a useful video stream transmission in military applications differ from entertainment applications (e.g., YouTube, Netflix), they can be measured similarly [5].

Measurements used to evaluate video stream playback perceived by the final user can be classified into objective and subjective metrics [6]. Objective measurements can be collected in the user video player (e.g., playback start time, the number of video interruptions, and duration of interruptions). Subjective metrics like mean opinion score (MOS) are based on the user feedback, collecting measurements directly from the users. Researchers have demonstrated that it is possible to infer the user subjective evaluation (e.g., MOS) based on the observed objective measurements [7, 8].

In entertainment applications, more attention is given to the overall user satisfaction (e.g., MOS), whereas in military applications some objective measures are critical. For

example, a long time freeze on the video playback may omit important events and lead to erroneous military decisions, whereas several short video freezes result in a low MOS evaluation by an ordinary user, but the provided information in this frozen video frame may be sufficient for specific military purposes. Observing these aspects, video quality assessments can be used as a feedback to adjust network settings and implement policy enforcement to improve or preserve video quality according to the application requirements [9, 10].

In light of this discussion, SDNs (software-defined networks) [11] provide ways to apply configuration changes and policy enforcements in a dynamic and practical manner. These benefits are allowed by decoupling the network into two different planes: a control plane, which implements the network control, and a data plane, which just forwards packet according to control directives. The network control is centralized in the SDN controller, which provides a programmatic interface to the network, allowing external applications deal with the network as a single system. Due to this abstraction, the SDN model can be deployed in heterogeneous networks [12, 13] and represents a suitable model to address the needs of the military video surveillance networks.

It is a major challenge to keep video quality requirements at acceptable levels, given the high complexity of managing the resources of the network in a highly dynamic environment such as a multi-UAV-based surveillance setup. Most of the solutions proposed to UAV surveillance networks (commercial or military) make use of ad hoc solutions based on wireless mesh networks [14]. The use of conventional network solutions (non-SDN) in these applications makes the device reconfiguration process difficult, or dependent on proprietary solutions. Moreover, proprietary solutions add extra difficulty to interface new system to already deployed applications [15]. The configuration of new equipment joining the network also becomes more costly using conventional ad hoc solutions, thus hindering network scalability.

Observing these problems in conventional solutions for UAV-based video surveillance systems and given the flexibility and agility provided by SDN, in [16] the seminal ideas of exploring the network programmability offered by the SDN (i.e., adjusting end-to-end paths according to network measurements) to improve the quality of video stream transmissions were presented. In this current paper, those ideas from [16] are further explored, providing a complete SDN-based solution for the target problem. Moreover, a complete experiment campaign was performed. Relevant parameters to video surveillance applications are considered, such as playback start time and the duration of interruptions. An assessment of video quality is conducted employing objective measurements collected on the client side. From these collected measurements, the MOS assessment representing the overall video quality is performed. The obtained results show that the proposed approach can handle the challenging UAV-based military surveillance operational scenario.

The major contributions of this paper are (i) the application of software-defined networks in UAV-based military surveillance systems aiming at compliance to strict video streaming QoS requirements; (ii) the demonstration of

the feasibility of this proposed approach using OpenFlow [11], enabling the efficient usage of commercial off-the-shelf (COTS) equipment in military networked surveillance systems; (iii) the analysis based on quality of experience indicators and experimental results supporting the proposed approach; and (iv) a comprehensive literature review of relevant works in the area, which highlights the step ahead provided by the proposal here presented.

The paper is organized as follows: Section 2 presents and discusses the related work. Section 3 introduces the application scenario for UAV-based military surveillance used in this work. Section 4 presents the proposed SDN approach to support video streaming in dynamic networks composed of mobile nodes. Section 5 describes the experiments carried out and the obtained results, while Section 6 draws the conclusions and provides directions for future work.

2. Related Work

Tortonesi et al. have discussed in [15] that the adoption of commercial off-the-shelf (COTS) hardware and software has gained an increasing interest in military Tactical Edge Networks (TEN), which is noteworthy in UAV-based systems [17]. In fact, there is a growing research interest in the use of IoT equipment and COTS hardware and software in modern UAV applications. In [18], the authors present a video surveillance application that uses facial recognition techniques to allow remote monitoring of crowds in places of interest. The use case reported in this work consists on offloading the data, through a wireless connection, to process the video in a mobile edge computing environment. While the data offloading process can deal with flexible network requirements, a live video transmission requires a very strict delay bound and high bandwidth [1].

The use of standards designed for wired connections or corporate networks and the reliance on TCP connections can often lead to low performance when the COTS solutions are running in TEN, especially when highly QoS sensitive applications are taken into account [15]. COTS solutions are not prepared to cope with some specific characteristics of TEN, as the frequent disconnections of nodes caused by several reasons. The use of legacy systems can add extra difficulties to the scenario, since these applications were not developed to deal with intermittent links. Additionally, the UAVs need to opportunistically explore new resources in the network, switching to different service providers. As the UAV moves to an area to provide network connectivity to a remote network partition, or when there is a need to select links offering different QoS characteristics, new connections will appear and existing connections can be dropped.

In [19], some of the benefits of using SDN in the military scenario are highlighted. The authors propose an architecture to apply the software-defined network paradigm to cope with the inherent properties of military networks. The cooperation of legacy networks and software-defined networks is addressed in the proposed architecture. The authors argue that battlefield networks can take advantage of the benefits of software-defined networks, such as ease of setup, flexibility in policy enforcement, flow optimization,

and network adaptation. To demonstrate some of the benefits, a use case involving a real-time video application, with end-to-end delay and video quality constraints, was described. The network controllers, with the help of a flow optimization application, can select the most appropriate path to forward the video stream, meeting the task requirements. Despite the consistent argumentation, the paper brings no experimental evidence supporting the proposed approach.

Recent research work discusses video flow optimizations through networks configuration changes. Aiming at providing a better quality of experience (QoE) to the end user, network optimization techniques, policy enforcements, and appropriate path selection are proposed. The decision-making process makes use of statistics collected directly from video players or from network measurements. In [9], Nam et al. propose an architecture and an SDN controller that can adjust the network parameters to deliver video traffic to the final user with better QoE. An improved HTML5 player obtains the video QoE measurements on the client side. The video player collects information about the status of the player, selected video resolution, video buffer rate, and playback start time and reports it to the delivery node. Buffering events and packet loss thresholds trigger network reprogramming routines, and according to the gathered information, new routing paths are selected. The authors use the Junos Space SDK from Juniper Networks to implement the prototype.

In [20], the authors explore the overall view of the network, provided by the SDN controller, to assist the Dynamic Adaptive Media Players (DASH) to select the best video resolution supported by the network. The Linux built-in traffic control mechanisms are used to apply the QoS configurations and enable concurrent video players to use the available bandwidth in a fair way. The DASH player has been extended to report media information and buffer status to a Service Manager. The Service Manager interacts with the SDN controller that is in charge of applying the necessary changes to the network hardware using the OpenFlow [11] protocol. The work presented in [10] proposes a solution to cope with network bandwidth competition among concurrent video flows in the network. The authors aim to reduce the player instability (e.g., the need for DASH players to switch between video streams with different resolutions frequently) and maximize the fairness among different video clients on the same network. A peculiarity of these solutions presented above is the acquisition of network measurements to estimate user QoE and perform adjustments in the network behavior. As Juluri et al. [6] pointed out, traditional network QoS measurements are not sufficient to determine users' satisfaction. Instead, it is necessary to collect measurements perceived by the users, allowing determining their quality of experience (QoE).

The QoE is the subject of an extensive survey conducted by Juluri et al. in [6]. The authors offer a tutorial overview of the existing video delivery methods along with the presentation of measurement techniques of video QoE. According to the employed measurement mechanism, the QoE metrics are classified into objective or subjective metrics. Objective metrics, as playback start time, the number of interruptions, and

duration of interruptions, can be collected by measurement tools on the video-player software. Meanwhile, subjective metrics are based on the experience reported by the user while using a video service. The most popular subjective and default metric for subjective video application assessments is the mean opinion score (MOS). Recent investigations have shown a correlation between objective and subjective metrics. Therefore it is possible to predict the QoE of the final user based on objective metrics [21, 22].

The present work differs from the previous ones by performing SDN within a new model to meet temporal requirements, considering extremely dynamic networks and applications. While the studied literature focuses on video stream optimizations in local wired networks, the proposed video application is intended to be used in military surveillance applications supported by wireless networks. The data (i.e., video) source consists of UAVs carrying video cameras collecting images and videos on surveillance missions. Therefore the video sources are in a highly mobile environment. Further, the degradation of communication links directly affects the video quality experienced by the user and, therefore, the application evaluation, as it occurs in commercial applications. The cited QoE objective and subjective metrics studied literature can be used to obtain an overall quality assessment of the service provided to the final user. Additionally, the collected QoE metrics can be used to optimize or select the appropriate path for data transmissions through the network, taking advantage of easy reconfiguration provided by the SDN paradigm, as proposed in this work.

3. Military Surveillance Application Scenario

Several areas need to be monitored by the military forces (e.g., borderline, critical infrastructure, enemy-occupied areas, and other harsh environments). To carry out this monitoring, a suitable proposal is to use a fleet of unmanned aerial vehicles (UAVs). UAVs equipped with visible light or infrared cameras should obtain surveillance video of the monitored site. The UAV sends the captured multimedia data to a command center, where the video will be examined and the information gathered from it will be used for decision-making.

The UAVs and the Command and Control (C2) systems form a network which may be partially disconnected or disrupted due to the possibility of wide UAV movements. On one hand, wide movements allow the surveillance of a larger area. On the other hand, by flying too far from the access points, UAV may be temporarily disconnected from the remaining of the network. A study considering the trade-off between coverage area and maintenance of a relay network was addressed in [14].

Focusing on military reconnaissance missions, in which military troops have to survey an area to gather information about the enemy occupation, the combined use of small UAVs and conventional ground military vehicles is a promising setup. This combination can provide awareness of threats expected ahead of the troop's line of sight. In this kind of reconnaissance missions, the military vehicles move along an axis of advance in the direction of the enemy, and the enemy

FIGURE 1: Schematic scenario in which UAVs provide clearance to the ground military vehicles advance along the axis.

FIGURE 2: Schematic scenario in which UAVs provide clearance while the ground military vehicles explore an area.

is located ahead of the platoon as shown in the schematic scenario in Figure 1.

Another reconnaissance situation in which UAVs can help the troops on the ground occurs when a reconnaissance platoon arrives at a particular area and explores the area to secure it for the installation of incoming additional troops, as schematically presented in Figure 2. The UAVs in this scenario are not organized ahead of a platoon, as in the first scenario, and the ground military vehicles themselves do not form a platoon. Both the ground military vehicles and the UAVs move freely in the area being explored searching for a possible threat.

In the situation presented in Figure 1, the goal is to acquire visual information about the clearance of the area ahead.

Using this information the platoon commander can decide about the advancement of his troop. The vehicle occupied by the platoon commander is in general one of the last vehicles in the row. The video acquired by the UAVs has to be delivered then to this vehicle and played in the commander's C2 terminal. In the situation presented in Figure 2, the idea is to provide immediate response to hostile elements that may be detected in the area that is being explored. Thus, the idea here is to deliver the video primarily to the closest vehicle. Additionally, the captured video is forwarded to the C2 terminal, located in the commander's vehicle.

The video transmission in the described reconnaissance scenarios must meet strict requirements, under the risk of losing significant events about the enemy movements. A very important requirement relates to the duration of each possible video interruption. A long interruption in the monitoring video allows an undetected enemy to come close to the platoon, posing security risks. For video transmissions triggered by some event (such as the detection of a person or a vehicle), the video playback start time is also important. Due to intermittent characteristics of these types of video stream, delays in the beginning of the stream can lead to the same situation caused by long term video freezes.

Taking a concrete example in the first scenario presented in Figure 1, the platoon moves at a speed of 60 km/h along the axis of advance. The enemies approach at the same speed in the opposite direction and the UAVs can provide videos of the enemies movements 1 km ahead. It gives only 30 seconds before the line of contact. Thus, video freezes closer to this period make a reaction practically impossible. Another example, but related to the second scenario presented in Figure 2, may consider the video acquisition to gather detailed information about possible enemies infiltrated in the area under surveillance being explored. In this case, considering the second scenario, differently from the first one, possible freezes are not too important, but the resolution of the provided images once something is detected is of primary importance.

4. An SDN Architecture to Enhance Video Streaming in Dynamic Networks

The expansion of the ground coverage area maintaining connectivity among UAVs and the ground platoon is one of the problems to be addressed in scenario 1. The increase of the coverage area while retaining connectivity to the rest of the ground squad can be acquired using intermediate UAVs as relay nodes. The relay nodes act forwarding the data until it reaches the destination node. The use of intermediate relay nodes has a drawback: the overload of the intermediate nodes. Intermediate nodes send data collected by their sensors (e.g., video cameras) and additionally the data sent by the neighbors to the intermediate node. The neighbors are not able to establish an end-to-end connection with the destination nodes; therefore they should use other nodes within their communication range to deliver the acquired data (the video stream) to the destination node. In this arrangement, the throughput of the communication channels is shared among different video streams.

The concurrency of the throughput by many video flows directly affects the video QoE perceived by the user, negatively impacting video applications and leading to a bad evaluation of the application by the user. The negative feedback of a video application is expressed by a low MOS score. The selected path to forward the video stream and the number of hops needed to reach the destination have a significant impact on this assessment. A management entity with global view of network status is able to select optimal link paths to route the data, avoiding congested links and choosing paths with fewer hops. A software defined network based on the OpenFlow [11] architecture enables the network controller to gather information about the global state of the network. The centralized controller can detect congested and deteriorated communications links that lead to low throughput and high latency, therefore insufficient resources for the video stream.

The centralized control plane and the features present in the OpenFlow implementation allow the network reprogramming according to the link usage and applying routing protocols capable of distributing the data streams among redundant links [23], splitting the data among different links [24], or selecting a path that offers the appropriate resources for the video transmission avoiding video freezes. The data exchanged among military applications might have different QoS requirements which can vary while the mission is under execution. Also, sometimes there is a need for traffic isolation, such as in collaborative missions carried out by distinct military forces [19, 25]. The global view of the network, the abstraction of the forwarding plane, and the programmatic interface provided by the logically centralized SDN controller allow applications to adjust network parameters and fulfill security and QoS requirements. Additionally, external applications can use the information of the global view of the network to ensure that those network policies (such as security and QoS policies) are correctly implemented and react to anomalous behavior and incorrect configurations [26].

Video streams used in military surveillance applications show slightly different characteristics to those used in home entertainment systems. The purpose of the captured video in military missions is the rapid detection of threats, gathering information of the resources or activities of a potential enemy, to obtain reliable information about certain areas or assist in the organization of the units. The features of the remote video terminals (RVT) and displays on embedded systems also influence the choice of the videos resolutions. The images are usually obtained by visible light cameras, synthetic aperture radar (SAR), and electro-optical and infrared devices. The videos used in the military context often range from images with resolutions of few kilopixels to HD images (e.g., 1920x1080 pixels) [27–29]. Focusing on reconnaissance missions in which the UAVs are used to survey an area ahead of troop line of sight, like that presented in Figure 1, videos with low resolution (e.g., 640x512 pixels) are enough to provide the information to enable decisions about the current situation, whereas, in the combination of UAVs and ground vehicles used to secure an area for the installation of additional troops, like depicted in Figure 2, a higher video

FIGURE 3: Overview of the proposed SDN employment in network of ground military vehicles and UAVs.

resolution is required to detect and identify threats nearby ground troops [5, 30].

The proposed network architecture provides a software-defined environment to manage the data and the SDN controller to handle the network of ground vehicles. This network of the ground vehicles can be considered stable compared to the network of UAVs, which enables an efficient usage of the SDN controller. Each of the ground vehicles is equipped with a data forwarding device supporting the OpenFlow protocol enabling the creation of a programmable network among them. All of the data forwarding devices are connected to the same SDN controller, enabling a centralized management of the network. Thus, in this proposed architecture each ground vehicle is considered a network switch controlled by the SDN controller, as presented in Figure 3. This figure presents the general idea of the proposed architecture, which can be implemented for both the first and the second scenario configurations presented in Figures 1 and 2.

The UAVs move around the area where they are conducting the surveillance mission, and therefore, the connection to the network composed of ground vehicles is not stable. During the mission, the UAVs can often reconnect to the same ground vehicle or connect to a different one. The SDN controller must optimize the network to make the reconnection process as smooth as possible, causing minimal impact on the quality of the video being displayed to the user.

When starting the network operation, a topology discovery process is performed, depicted in Figure 4(a). The topology discovery process allows the controller to select the most suitable path for the transmission of the video streams generated by each of the UAVs to the final users' remote video terminal. Once an OpenFlow-enabled device connects to the SDN controller through a TCP connection, the controller sends a feature request message to the device and waits for a response. The device answers with a feature reply message containing its characteristics (e.g., datapath id and ports).

FIGURE 4: Schematic SDN controller architecture.

This message exchange is part of the OpenFlow protocol handshaking process and allows the controller to be aware of all forwarding devices in the SDN data plane. The LLDP protocol performs the discovery of data links among the already discovered devices.

In the SDN controller, the topology data gathered is represented by an undirected graph, block (b) in Figure 4. The graph vertices represent the forwarding devices and access points found, and the graph edges represent the data links among discovered devices. The controller computes the routes among the devices based on the information contained in the network graph representation. During the mission, the topology of the network can change. The network devices (forwarding devices and access points installed in the ground military vehicles) can establish new links among them and between network devices and UAVs. Additionally, due to UAV moves, existing connections may disappear. In this case, the graph needs to be updated to correspond to the new network topology. The controller detects OpenFlow-enabled devices that are joining or leaving the data plane through the OpenFlow Channel. Furthermore, the controller checks for updates in the links by periodically checking their state using the LLDP protocol. When a topology event occurs, the Topology Manager depicted in Figure 4, block (d), updates the network graph representation.

After the initial topology discovery of the OpenFlow-enabled devices, the SDN controller is ready to process the requests from forwarding devices in the data plane and the Path Selection Algorithm represented in Figure 4(c) will select a network path to forward data according to the flowchart depicted in Figure 5. The OpenFlow/LLDP discovery process does not detect the UAVs in the TEN network, and the prototyped SDN controller identifies the UAVs based on a Layer 2 learning process. When the UAV joins the TEN through a wireless connection with a ground vehicle, the OpenFlow-enabled access point does not have a configured matching rule to forward the packets. Following the default behavior of OpenFlow specification by the table

FIGURE 5: Path Selection Algorithm.

miss entry, the forwarding device sends the data packet to the SDN controller. The key components of the SDN Controller that perform these tasks and their relations among each other are presented in the schematic SDN controller architecture illustrated in Figure 4.

Each data packet that arrives at the SDN controller triggers the *Packet_in* event. The SDN controller follows the algorithm depicted in the SDN controller flowchart (Figure 5). As a UAV joins the TEN, the network graph representation does not contain information about the UAV; thus the SDN controller updates the network graph representation. Next, the SDN controller performs a search for the destination address host *dst_address* in the network graph. If the target host specified by the *dst_address* field in the data packet is found, the network controller installs an OpenFlow entry in the forwarding device that originates the *Packet_in* event and forwards the packet to the next hop, or to the destination host. Otherwise, if the network graph representation does not have information about the host specified by the *dst_address*, the controller floods the network with an ARP Request packet until the destination host answers the request with an ARP Response packet. To prevent packets from being forwarded in network loops, the controller gets information about all ports in the *datapath* (logical representation of the forwarding device), computes the ports that may cause loops in the network using a minimum spanning tree (MST) algorithm, and sends the ARP Request package only to loop-free ports.

5. Experiments and Results

This section presents the experiments carried out by means of simulations and the obtained results. Initially, a description of the QoE parameters taken from the media player application and their influence on the quality of experience provided

to the user are briefly described. Then the simulation environment, the selected tools, and the changes implemented to the player to enable collection of QoE measurements are reported. Additionally, the characteristics of the simulated scenario and the parameters used in the simulations are described. Finally, the obtained results and discussions about their effects in the proposed application scenario are presented.

5.1. Selected Evaluation Metrics. Three objective metrics were elected to quantify the Application Quality of Experience (AppQoE) on the client side of the surveillance system using video over HTTP [6, 31]. These metric are as follows:

- (i) *Video playback start time*: it corresponds to the time taken by the player to start the *playout*. As it was used as a standalone video player application, the measured time corresponds to the request of the stream, the download of the initial part of the video filling the player buffer to a threshold set by the application (*Initial Buffering*), and the playout of the initial part of the video.
- (ii) *Number of interruptions*: when the playback is temporarily frozen a video interruption is computed. This event occurs when the throughput of the network is not enough to keep the video player reproducing the video. The player buffer decreases to a low value, nearly zero, and the player waits for the buffer to be partially filled again to resume the video playout. This event is also referred to as a *(re)buffering* event.
- (iii) *Total duration of interruptions*: this metric is a sum of the durations of all interruptions (*Buffering Time*) during video playout. The first buffer event is ignored because it corresponds to *Initial Buffering* [6].

The predictions of the MOS values were obtained from the AppQoE data collected. Recent studies [21, 22, 32] relate the influences of the three selected metrics with the degradation of user experienced quality, represented by MOS score that would be assigned by the user. In [22], the authors relate the video playback start time with the MOS value using (1), where $Qini$ is the MOS influenced value and t_0 is the video playback start time obtained in measurements.

$$Qini = -0.963 \times \log(t_0 + 5.381) + 5 \quad (1)$$

The MOS value influenced by video stalls ($Qstall$) can be obtained by applying (2) and (3) to the AppQoE data [32]. The λ factor is the ratio of the total time that the video was stalled (σ) and the interval which elapsed since the beginning of observation, given by the sum of σ with the effective video play time (ρ). The value of λ is used to determine the value of the a_i , b_i , and c_i constants in (3) according to predefined values observed by Casas et al. [33]. The n variable corresponds to the number of video interruptions on the T time observation interval, considered as one minute in the present work. The values of “N” are mapped for up to $n = 6$ because, with a

value greater than this threshold, the MOS assumes the value 1, which means a very bad quality experienced by the user.

$$\lambda = \begin{cases} \frac{\sigma}{\sigma + \rho}, & \text{if } \sigma + \rho < T \\ \frac{\sigma}{T}, & \text{otherwise} \end{cases} \quad (2)$$

$$Qstall = \begin{cases} 1, & \text{if } n > 6 \\ a_i \times e^{-b_i \times n} + c_i, & \text{if } n \leq 6 \end{cases} \quad (3)$$

Since $Qini$ and $Qstall$ were obtained independently, from different functions, the minimum value was assumed as the final video MOS, as can be seen in (4). Therefore, the final video MOS will be limited by the minimum value observed in $Qini$ and $Qstall$.

$$MOS = \min \{Qini, Qstall\} \quad (4)$$

5.2. Simulation Environment and Tools. This work gives more emphasis on simulation of the first scenario, where the relay network is used to expand the coverage area. Additionally, this topology stresses the network connection point between the network established among UAVs and the ground vehicles network.

The video AppQoE measurements were performed using a modified version of the FFplay player. According to the software documentation, FFplay is a portable media player using the FFmpeg libraries [34]. It is mainly written using the “C” programming language. Small changes were made in the source code of the FFplay, and a recompiled version of the player including the modifications was used in experiments. The first change was implemented to make it possible to compute the video initialization time. A timer starts when the player requests the video data to the server. Once the player receives enough data to display the first frame of the video, the video playback start time is computed. A modification that monitors the player video queue buffer computes each of the video stalls. The FFplay stores the received data in a video queue buffer. If the network throughput is lower than the required video bitrate, the buffer becomes empty, causing a video stall. The changes in the player source code can compute each video stall and the stall length. The acquired data is reported in the player log allowing a data analysis later.

Each of the UAV hosts, acting as video sources, runs an instance of the FFServer media server. The FFServer is a streaming server, for both audio and video streams, and it is also a part of the FFmpeg software package. No changes were introduced in FFServer software to act as a video stream server, and it was configured to serve a video stream over the HTTP protocol. The data is transferred using the TCP protocol.

The choice for the protocols used in this work, namely, HTTP over TCP, follows the previous studies performed in this domain which are reported in [3], which are based on the NATO standards for video streaming in military operations stated in [35, 36]. It is possible to discuss how suitable these protocols are to address the needs of video streaming applications in TENs, which is also detailedly explored in [15]. Based

TABLE 1: Parameters used in the simulations.

Parameter	Used Value
Number of UAVs	9
Number of ground vehicles	9
Total simulation area	4 Km x 4 Km
UAV moving area	4 Km x 1 Km
UAV communication radius (R_u)	360m
Ground communication radius (R_v)	650m
Mobility models	Random Walk, Random Waypoint
Video size	960x540 pixels
Video frame rate	30 fps
Video codec	H.264
Video length	60 seconds
Number of runs per number of streams being served	33 runs

the farthest vehicle of UAVs squad (the vehicle at the bottom of Figure 6). The node selection for the video playback is in accordance with the instructions of the army, considering that the mission commander will have access to video and that it usually takes one of the last vehicles of the convoy. Moreover, this choice represents the worst case for data transmission because the data flow should travel the entire ground vehicles network, before it reaches the destination host. Table 1 summarizes the main simulation parameters.

5.4. Results Presentation and Discussion. Although the results in extreme cases present MOS notes below the expectations, positive facts can be listed. In all of the performed experiments, the video playback start time measurements hold at an acceptable value for a video surveillance application and its value is negligible. The values are in the range from 800 milliseconds to 1300 milliseconds as depicted in Figure 7. Regardless of video playback start time influence in the final MOS note, i.e., the value being slightly greater than 4 for the cases with up to 4 simultaneous streams, this value is still acceptable for the majority of video applications, for both entertainment and surveillance applications.

Revisiting the requirements presented in Section 3, particularly examining the provided example in which a platoon moves at a speed of 60Km/h against enemies coming from the opposite direction at the same speed, the use of the UAV squad providing clearance in the area 1Km ahead sufficiently addresses the needs. The video playback starts within an acceptance time interval, even in the worst case, as can be observed in Figure 7.

The number of video interruptions, or video stalls, was strongly influenced by the number of simultaneous video flows. The various streams competing for available bandwidth generated resource contention, especially in the context of the wireless network environment.

The simplest mobility model, i.e., the Random Walk, resulted in more stable wireless associations, both among the UAVs and between them and the access points. Stable and lasting wireless associations produce better results in

FIGURE 7: Video playback start time measured in scenario 1.

FIGURE 8: Number of interruptions measured in scenario 1.

the video assessments because they allow a higher data rate transfer, avoiding the starvation of the player buffer (i.e., *rebuffering events*). As a consequence, the number of video stalls and the total stall length depicted in Figures 8 and 9 for the Random Walk mobility model present slightly better values compared to Random Waypoint mobility model. Even in the simulations using the Random Waypoint mobility model, the total video stall time is satisfactory up to eight simultaneous video streams. Figure 10 depicts the average length per video stall, i.e., the time that the image remains frozen per video stall. Even in the worst case, with nine video streams competing for the network bandwidth, the time of each video stall is shorter than 1 second, which is an acceptable value for surveillance and reconnaissance missions making use of video streaming [3]. Still considering the example presented in Section 3, the total stall length is also within an acceptable interval, being in the worst case 10s,

FIGURE 9: Total duration of interruptions in scenario 1.

FIGURE 10: Average time per video stall measured in scenario 1.

which is still a short time considered the time to reach the line of contact.

From the user perspective, according to the predicted MOS values in Figure 11, good results were observed using up to 4 simultaneous video streams with the Random Walk mobility model, which represents approximately a half of the UAV platoon sending video traffic simultaneously. It is noteworthy to observe the video frame rate used in the simulation (~30 fps). Normally, video surveillance does not require this higher frame rate, and with smaller video frame rate the streams should be less resource-intensive. Thus a better user experience is expected, resulting in a higher MOS. The same applies to the video resolutions selected for the simulations. The selected video size is in the range of values used in the military context. However, many military applications make use of smaller video size. Finally

FIGURE 11: Predicted mean opinion score (MOS) perceived by the final user in scenario 1.

the case with all UAVs generating video traffic was being simulated. However, it is expected that a smaller number of video transmissions occur simultaneously in surveillance and reconnaissance applications.

6. Conclusion

In this work, an application of Software Defined Networking was demonstrated to enhance quality of experience in video streaming in the context of military mobile networks. The proposed approach aims at linking a heterogeneous network composed of UAVs and ground vehicles focusing on surveillance and recognition applications in military operations context. Although the work considers military applications, the proposed scenarios can also be used in the civilian domain. The proposal was evaluated in an emulator for Software Defined Wireless Networks. Objective QoE measurements were collected using a modified version of a popular media player. The collected measurements were used to predict the subjective quality of experience indicator mean opinion score (MOS). The results were promising and demonstrate that programmable networks can be successfully applied to heterogeneous networks, disruptive networks, and networks with opportunistic connections.

As future work, the application of SDN technologies to the network established among the UAVs can be suggested, allowing greater control of packet routing while they are forwarded in the relay network. The use of SDN in the relay network also enables the implementation of DTN protocols that could use the global knowledge of the SDN controller about the network to optimize the transmission and routing of packets on opportunistic links that appear due to the UAVs movement. Information Centric Networking (ICN) could also be considered in conjunction with SDN, so that caching mechanisms could be explored to store video in intermediary nodes to provide faster video delivery to multiple requesting users spread geographically. Finally, the full implementation of SDN protocols could be deployed in real UAVs networks, for more accurate assessment, and testing of real scenarios.

Data Availability

The simulation data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors wish to acknowledge State of Rio Grande do Sul Research Foundation (FAPERGS), Brazilian National Council for Scientific and Technological Development (CNPq), and Brazilian Army for the partial financial support to perform this work.

References

- [1] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [2] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2016.
- [3] J. Nightingale, Q. Wang, J. M. A. Calero et al., "Reliable full motion video services in disadvantaged tactical radio networks," in *Proceedings of the 2016 International Conference on Military Communications and Information Systems, ICMCIS 2016*, Belgium, May 2016.
- [4] V. Joseph, S. Borst, and M. I. Reiman, "Optimal rate allocation for video streaming in wireless networks with user dynamics," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 820–835, 2016.
- [5] Department of the Army, *Long range unit surveillance operations: Field manual No. 3-55.93*, Washington, DC, 2009.
- [6] P. Juluri, V. Tamarapalli, and D. Medhi, "Measurement of quality of experience of video-on-demand services: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 401–418, 2016.
- [7] Y. Liu, J. Liu, Z. Xu, and S. Ci, "Choquet integral based QoS-to-QoE mapping for mobile VoD applications," in *Proceedings of the 24th IEEE/ACM International Symposium on Quality of Service, IWQoS 2016*, China, June 2016.
- [8] International Telecommunication Union, "Recommendation P.800.1: Mean opinion score (MOS) terminology," Recommendations of the ITU, Rev. 4.0, Jul., 2016.
- [9] H. Nam, K.-H. Kim, J. Y. Kim, and H. Schulzrinne, "Towards QoE-aware video streaming using SDN," in *Proceedings of the 2014 IEEE Global Communications Conference, GLOBECOM 2014*, pp. 1317–1322, USA, December 2014.
- [10] R. M. Abuteir, A. Fladenmuller, and O. Fourmaux, "SDN based architecture to improve video streaming in home networks," in *Proceedings of the 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pp. 220–226, Crans-Montana, Switzerland, March 2016.
- [11] N. McKeown, T. Anderson, H. Balakrishnan et al., "OpenFlow: enabling innovation in campus networks," *Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.
- [12] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: past, present, and future of programmable networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.
- [13] Taimur Bakhshi, "State of the art and recent research advances in software defined networking," *Wireless Communications and Mobile Computing*, vol. 2017, Article ID 7191647, 35 pages, 2017.
- [14] D. Orfanus, E. P. De Freitas, and F. Eliassen, "Self-organization as a supporting paradigm for military uav relay networks," *IEEE Communications Letters*, vol. 20, no. 4, pp. 804–807, 2016.
- [15] M. Tortonesi, A. Morelli, C. Stefanelli, R. Kohler Jr., N. Suri, and S. Watson, "Enabling the deployment of COTS applications in tactical edge networks," *IEEE Communications Magazine*, vol. 51, no. 10, pp. 66–73, 2013.
- [16] I. Zacarias, J. Schwarzrock, L. P. Gaspary et al., "Employing SDN to control video streaming applications in military mobile networks," in *Proceedings of the 16th IEEE International Symposium on Network Computing and Applications (NCA)*, pp. 1–4, Cambridge, MA, October 2017.
- [17] M. Tortonesi, C. Stefanelli, E. Benvegna, K. Ford, N. Suri, and M. Linderman, "Multiple-UAV coordination and communications in tactical edge networks," *IEEE Communications Magazine*, vol. 50, no. 10, pp. 48–55, 2012.
- [18] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-Based IoT Platform: A crowd surveillance use case," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 128–134, 2017.
- [19] J. Nobre, D. Rosario, C. Both, E. Cerqueira, and M. Gerla, "Toward software-defined battlefield networking," *IEEE Communications Magazine*, vol. 54, no. 10, pp. 152–157, 2016.
- [20] J. W. Kleinrouweler, S. Cabrero, and P. Cesar, "Delivering stable high-quality video: An SDN architecture with DASH assisting network elements," in *Proceedings of the 7th ACM International Conference on Multimedia Systems, MMSys 2016*, pp. 36–45, Austria, May 2016.
- [21] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoffeld, and P. Tran-Gia, "A survey on quality of experience of HTTP adaptive streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469–492, 2015.
- [22] T. Hossfeld, S. Egger, R. Schatz, M. Fiedler, K. Masuch, and C. Lorentzen, "Initial delay vs. interruptions: between the devil and the deep blue sea," in *Proceedings of the 4th International Workshop on Quality of Multimedia Experience (QoMEX '12)*, pp. 1–6, July 2012.
- [23] S. K. Singh, T. Das, and A. Jukan, "A survey on internet multipath routing and provisioning," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2157–2175, 2015.
- [24] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses," RFC Editor RFC6824, 2013.
- [25] M. R. Brannsten, F. T. Johnsen, T. H. Bloebaum, and K. Lund, "Toward federated mission networking in the tactical domain," *IEEE Communications Magazine*, vol. 53, no. 10, pp. 52–58, 2015.
- [26] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: a comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.
- [27] Department of the Army, *Army Unmanned Aircraft System Operations*, Washington, DC, 2006.
- [28] L-3 WESCAM, *WESCAM's MX-15: EO IR ISR Systems*, p. 2, 2015.
- [29] Stark Aerospace Inc, *POP200 Specs*, p. 2, 2015, <http://stark-aerospace.com/products-services/sensors/pop200/>.

- [30] Department of the Army, *Tactics, techniques, and procedures for reconnaissance and surveillance and intelligence support to counterreconnaissance: Field manual No. 34-02.1*, Washington, DC, 1991.
- [31] R. K. Mok, E. W. Chan, and R. K. Chang, "Measuring the quality of experience of HTTP video streaming," in *Proceedings of the 2011 IFIP/IEEE International Symposium on Integrated Network Management (IM 2011)*, pp. 485–492, Dublin, Ireland, May 2011.
- [32] R. I. T. Da Costa Filho, W. Lautenschlager, N. Kagami, V. Roesler, and L. P. Gaspar, "Network fortune cookie: Using network measurements to predict video streaming performance and QoE," in *Proceedings of the 59th IEEE Global Communications Conference, GLOBECOM 2016*, USA, December 2016.
- [33] P. Casas, R. Schatz, and T. Hossfeld, "Monitoring YouTube QoE: Is your mobile network delivering the right experience to your customers?" in *Proceedings of the 2013 IEEE Wireless Communications and Networking Conference, WCNC 2013*, pp. 1609–1614, China, April 2013.
- [34] F. Bellard, "FFMpeg," 2000, Available: <http://ffmpeg.org/>.
- [35] *Standardisation Agreement: NATO digital motion imagery standard*, Brussels, 3rd edition, 2009.
- [36] *Motion Imagery Standards Board, Motion Imagery Standards Profile (MISP)*, Springfield, VA, 2nd edition, 2018.
- [37] R. R. Fontes, S. Afzal, S. H. B. Brito, M. A. S. Santos, and C. E. Rothenberg, "Mininet-WiFi: emulating software-defined wireless networks," in *Proceedings of the 11th International Conference on Network and Service Management (CNSM '15)*, pp. 384–389, Barcelona, Spain, November 2015.
- [38] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: rapid prototyping for software-defined networks," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks (Hotnets '10)*, pp. 19:1–19:6, ACM, October 2010.
- [39] F. Bai and A. Helmy, "A Survey of Mobility Models in Wireless Adhoc Networks," *Ad Hoc & Sensor Wireless Networks*, vol. 30, p. 1, 2004.
- [40] Ryu SDN Framework Community, *Ryu SDN Framework*, 2015, Available: <http://osrg.github.io/ryu>.