Mahdi Mousavi, Hussein Al-Shatri, Wasiur R. KhudaBukhsh, Heinz Koeppl and Anja Klein, ”Cross-layer QoE-based Incentive Mechanism for Video Streaming in Multi-hop Wireless Networks,” in *Proc. IEEE 86th Vehicular Technology Conference (VTC),* Toronto, Canada, Sept. 2017.

Cross-layer QoE-based Incentive Mechanism for Video Streaming in Multi-hop Wireless Networks

Mahdi Mousavi*, Hussein Al-Shatri*, Wasiur R. KhudaBukhsh†, Heinz Koeppl†, Anja Klein*

* Communications Engineering Lab, TU Darmstadt, Germany, {m.mousavi, h.shatri, a.klein}@nt.tu-darmstadt.de
† Bioinspired Communication Systems Lab (BCS), TU Darmstadt, Germany, {wasiur.khudabukhsh, heinz.koeppl}@bcs.tu-darmstadt.de

***Abstract*—We study video dissemination in a multi-hop wireless network with a source and several users. The source intends to stream a video to the users of the network. For the sake of energy-efficiency, the video is disseminated through the whole network by the help of some users that forward the video to who other users. In such networks, designing a proper incentive for the forwarding users who consume energy for forwarding the video to others is of high importance. In this paper, we design an incentive mechanism based on a game-theoretic model in which a user is paid by its receiving users in case of forwarding the video to them. The video is layered and a higher quality of experience (QoE) at a receiving user is possible by receiving more layers of the video. A utility function is proposed for every user that captures the perceived QoE at the user and the cost she pays for the video. Moreover, it captures the reward the user receives from others in exchange for forwarding the video to them. The utility function is designed in a way that the users who contribute more in the network, in terms of forwarding the video to others, are paid more. A non-cooperative game is formulated in which every user selfishly maximizes its own utility and determines the number of video layers she prefers to receive. The game is iterative and converges to the Nash equilibrium point. The simulation results demonstrate that the proposed game theoretic model results in a higher QoE at the users as compared to that of a non-incentive video dissemination model.**
Keywords: Cross-layer model; quality of experience; cost sharing game; video streaming; multi-hop wireless network.

I. Introduction

Interest in video contents has significantly increased among the mobile users during the past years and a huge amount of network traffic is dedicated to video streaming [1]. Multi-hop communication is a promising communication technique that can be exploited to fulfill the ever-growing users requirements. In such a communication, a user, instead of receiving the data from a centralized access point, can obtain the data from another user in her proximity who has already received the same data. Since in a multi-hop network, the data is forwarded through the network by the help of some intermediate users, designing proper incentives for the users to forward the data to others is of high importance. Studies show that most of the users are reluctant to contribute in networks without receiving an incentive [2]. This selfish behavior of the users can degrade the performance of the whole network, especially when a user is located closer to the source than the others and her contribution in forwarding the data to others and acting like a gateway is important for the whole network. Thus, designing proper incentives can significantly affect the user experience in multi-hop wireless networks.

In this paper, we propose an incentive mechanism for video streaming in multi-hop wireless networks that provides a better video quality for the users who contribute more in the network. A well-known technique for representing a video in different qualities is by encoding the video into several layers. A layered video is composed of multiple layers, a base layer and several enhancement layers. Each layer has its own properties in terms of data rate requirement and affects the quality of the video perceived by the user differently. Receiving more enhancement layers results in a higher video quality.

Typically for network optimization, a set of technical parameters, like throughput maximization [3] or energy minimization [4], are considered as the Quality of Service (QoS) constraints at the users while the application layer requirements are usually ignored. Considering merely QoS constraints in video streaming scenarios may not necessarily result in user satisfaction. The overall user satisfaction is defined subjectively and is called quality of experience (QoE). Mean opinion score (MOS) [5] and video quality metric (VQM) [6] are two of the subjective QoE metrics. The MOS metric is divided into five levels according to the user perception: 5 and 1 represent the best and worst QoEs, respectively. The VQM is in the range of 0 and 1 such that the VQM value closer to 0 shows a higher QoE. Maximizing the subjective QoE are proposed in [7] and [8] over WLAN and Ad Hoc networks, respectively. In [7], a single-hop multicast transmission of a video over IEEE 802.11 is studied and the authors maximize the MOS by choosing a proper multicast group and modulation scheme at the transmitter. The approach proposed in [8] is to maximize MOS-based QoE by minimizing the packet loss in a network with lossy links. The aforementioned works, however, study different objective and subjective parameters in video streaming scenarios, none of them discusses incentive design for video streaming in user centric networks.

In this paper, we propose a novel decentralized cross layer model for layered video dissemination in a multi-hop wireless network that takes parameters from physical layer and application layer into account. Current work is built upon our recent work [9] where we proposed a game theoretic algorithm for energy-efficient multi-hop communication. In the current work, we consider layered video streaming as the application and design an incentive mechanism to maximize the QoE of the users while preserving the energy efficiency of the network.

We propose a game theoretic model in which the users of the network are considered as the players of the game and a user, in case of forwarding the video to others, will be paid by its receiving users. The payment from a receiving user to a transmitting user is considered to be via virtual currency [10] depending on the energy a transmitting user consumes. By having more contribution in the network, a node can collect higher reward (virtual currency) and therefore, can get higher layers of video which results in higher QoE. A utility function is defined for the users that takes the QoE of the user, the cost she pays for such a QoE and the reward it receives for her contribution in the network. By maximizing her utility function, the user determines the number of video layers she prefers to receive. Moreover, she determines which of the users in the network she prefers to receive each of the video layers from.

In order to benefit from broadcast nature of wireless channels, the users exploit multicast transmission. When a group of users receive a certain layer from the same transmitting user, a multicast transmission can serve all the receiving users at once. In such a case, the cost that has to be paid to the transmitting user is shared among the multicast receiving users using cost sharing methods [11]. This not only reduces the cost paid by the receiving users in a multicast group, but also decreases the energy spent in the network. Moreover, the proposed game converges to the Nash equilibrium (NE) point [9].

The rest of this paper is organized as follows: Section II describes the network model and the assumptions. In Section III, the proposed game theoretic model is described. Simulation results are provided in Section IV and Section V concludes the paper.

II. Network Model and Assumptions

In this section, we first explain the network model and the properties of the video which has to be streamed in the network. Then in subsection II-B, the interactions among the nodes of the network are discussed.

A. Video and Transmission Properties

A wireless network is considered composed of $N+1$ nodes, with a source S that intends to stream a video to a set $\mathcal{P} = \{1, \ldots, N\}$ of other nodes. The set $\mathcal{W} = \mathcal{P} \cup \{\mathrm{S}\}$ represents the set of all nodes in the network and every node $j \in \mathcal{W}$ has a transmit power constraint $P_j^{\max}$. The video is encoded into L layers and each layer $l \in \mathcal{L} = \{1, \ldots, L\}$ has data rate d^l bits per second. The layers of the video must be received at a node in consecutive order, that is, in order to decode layer $l > 1$, all the layers $1 \leq k \leq l-1$ are required at the node. A layer l increases the QoE of a user by q^l and $\mathbf{q} = [\mathrm{q}^1, \ldots, \mathrm{q}^L]^{\mathrm{T}}$ is an $L \times 1$ vector containing the qualities q^l of the layers such that $\mathrm{q}^l > 0$. Each video layer has a different impact on the QoE of the user and therefore, the values of q^l are different depending on the layer. Node i receives the video layers either directly from the source or via another node $j \in \mathcal{P}$.

The video dissemination flow from the source node to all other nodes of the network form a tree-graph rooted at the source, called the broadcast tree. Thus, as the video is encoded into L layers, the nodes form L separate broadcast trees such that each video layer is disseminated by a certain broadcast tree. The node determines how many of the broadcast trees it should join to. Due to the power constraint at the nodes of the network, node i cannot be served by any arbitrary node in the network. The set of nodes that can serve node i considering the power constraint is called the neighboring nodes of node i which is denoted by $\mathcal{N}_i$ and defined as

$$\mathcal{N}_i = \left\{ j \middle| j \in \mathcal{W}, P_j^{\mathrm{Tx}} < P_j^{\max} \right\} \tag{1}$$

in which P_j^{Tx} is the transmit power of node j. A node $j \in \mathcal{W}$ that transmits a video layer to the receiving node $i \in \mathcal{P}$ is called the parent node (PN) of node i for layer l and is denoted by a_i^l. Conversely, node i is referred to as the child node (CN) of PN j for layer l. The set of CNs of PN j for layer l is given by $\mathcal{M}_j^l$.

We define the vector of video layers *received* by node i as an $1 \times L$ binary vector $\mathbf{b}_i^{(\mathrm{r})} = [b_i^{1,(\mathrm{r})}, \ldots, b_i^{\mathrm{L},(\mathrm{r})}]$ in which $b_i^{l,(\mathrm{r})} \in \{0, 1\}$ and

$$b_i^{l,(\mathrm{r})} = \begin{cases} 1, & \exists j \in \mathcal{N}_i, a_i^l = j, b_i^{l-1,(\mathrm{r})} \geq b_i^{l,(\mathrm{r})} \\ 0, & \text{otherwise.} \end{cases} \quad . \tag{2}$$

Condition $b_i^{l-1(\mathrm{r})} \leq b_i^{l,(\mathrm{r})}$ indicates that the layers of the video must be received in consecutive order so that they can be decoded. The QoE of node i is given by the aggregated quality node i obtains from each layer, i.e.,

$$Q_i = \mathbf{b}_i^{(\mathrm{r})} \mathbf{q}. \tag{3}$$

We consider a threshold model for decoding the received signal at a node, such that a minimum signal to noise ratio (SNR), denoted by γ^{th}, is required at a CN in order to decode the signal transmitted from its PN successfully. In the transmission from PN j to CN i, the received SNR at the CN is calculated by

$$\gamma_{i,j} = \frac{p_j^{\mathrm{Tx}} g_{i,j}}{\sigma^2} \tag{4}$$

in which p_j^{Tx} is the transmit power of PN j, $g_{i,j}$ is the channel gain between them and σ^2 is the noise power at node i. Consequently, the transmit power at PN j in a unicast transmission to CN i considering γ^{th} is obtained by

$$p_{i,j}^{\mathrm{uni}} = \frac{\gamma^{\mathrm{th}} \sigma^2}{g_{i,j}}. \tag{5}$$

It is assumed that γ^{th} and $g_{i,j}$ for the nodes i and j are the same for all the video layers. Thus, the minimum power required at the PN j for transmission to CN i, i.e., $p_{i,j}^{\mathrm{uni}}$, is independent of the video layer. The main difference in transmission of different video layers from PN j to CN i comes from different data rates required by each of the video layers that result in different energy consumption at the PN. The energy required at PN j for unicast transmission of layer

l to CN i, denoted by $e_{i,j}^{\text{uni}}$, depends on the data rate d^l of the layer as

$$e_{i,j}^{l,\text{uni}} = \frac{d^l}{\mathrm{n_b}} p_{i,j}^{\text{uni}} T_s \tag{6}$$

in which $\mathrm{n_b}$ is the number of bits per symbol transmitted from the PN j with symbol duration T_s which are assumed to be the same at all nodes and all the video layers. We stick with a simple resource allocation scheme and assume that each video layer is transmitted in a different orthogonal channel by each transmitting node and thus, there is no collision in the network. It is also considered that the capacity of the channels, utilized by the transmitting nodes, are higher than the data rate required for the layers.

In a multicast transmission, when a PN transmits to multiple CNs, the transmit power of PN j for layer l is given by

$$p_j^l = \max_{i \in \mathcal{M}_j^l} \{p_{i,j}^{\text{uni}}\}, \tag{7}$$

that is, the CNs in $\mathcal{M}_j^l$ that requires the highest unicast power determines the transmit power of node j for layer l. Similar to the transmit power, the energy required at transmitting node j for transmission of layer l to its CNs in $\mathcal{M}_j^l$ is given by

$$e_j^l = \max_{i \in \mathcal{M}_j^l} \{e_{i,j}^{l,\text{uni}}\}. \tag{8}$$

The vectors $\mathbf{p}_j = [p_j^1, \ldots, p_j^L]^{\mathrm{T}}$ and $\mathbf{e}_j = [e_j^1, \ldots, e_j^L]^{\mathrm{T}}$ are $L \times 1$ vectors with p_j^l and e_j^1 representing the transmit power and the consumed energy of node j for transmission of layer l, respectively. We define the vector of video layers *transmitted* by node j as a $1 \times L$ binary vector $\mathbf{b}_j^{(\mathrm{t})} = [b_j^{1,(\mathrm{t})}, \ldots, b_j^{\mathrm{L},(\mathrm{t})}]$ in which -

$$b_j^{l,(\mathrm{t})} = \begin{cases} 1, & \mathcal{M}_j^l \neq \emptyset \\ 0, & \text{otherwise,} \end{cases} \tag{9}$$

so that, the total transmit power of PN j is given by $P_j^{\mathrm{Tx}} = \mathbf{b}_j^{(\mathrm{t})} \mathbf{p}_j$.

B. Interactions of the Nodes in the Network

Before introducing the mathematical expression of different parameters related to the our proposed mechanism, using Fig. 1, we briefly explain the reason for defining parameters. In this network, for any one-hop transmission from a PN to a CN (or a group of CNs in a multicast transmission), a *cost* is paid to the PN from the CN that receives the video layer. To avoid confusion, the cost that the CN pays, from the PN's point of view, is referred to as the *reward* of the PN due to forwarding the video layer, however, they are equal. The cost that a CN pays to its PN (or the reward the PN receives) is a term defined between a PN-CN pair, and as we will show, depends mainly on the energy the PN consumes to serve the CN, so that, the more energy the PN spends the higher the reward it receives. Beside the cost, another parameter ,called the *incentive*, is defined in this network for the case that a layer received by a CN is forwarded by to other other nodes of the network. In such a case, as the CN, node i in Fig. 1, receives reward in exchange for forwarding the video layer to other nodes, sends part of its reward, which is received from its CNs in $\mathcal{M}_i^l$, as incentive to its PN, i.e., node j in Fig. 1. This incentive at node j is like a credit for node j's contribution and shows the importance of node j in the network, or in other words, the importance of the link between the nodes j and i for the rest of the network. Therefore, unlike the cost (or reward) which depends on the properties of the link between a PN and a CN, incentives reflect the importance of the role that a node plays in the network.

In our mechanism, the concept of incentive will be exploited in order to motivate the intermediate nodes to get the video layers form their PNs and distribute in the network. For instance in Fig. 1, when node i is requested by node m for a video layer, it can get the video layer from its PN, node j in Fig. 1, transmit it to node m, receive reward from node m in exchange, and send part of it as incentive to node j because of its contribution.

Now, we model the cost, the reward and the incentive. We assume that the cost paid by node i to receive the video layer l from node j is denoted by $c_{j,i}^l$ but, in general, the cost that node i pays for layer l to its PN, which is one of its neighboring nodes, is denoted by c_i^l such that $c_i^l = c_{j,i}^l$ if node j is the PN of CN i for layer l, i.e., $a_i^l = j$.

By defining an $L \times 1$ vector $\mathbf{c}_i = [c_i^1, \ldots, c_i^L]^{\mathrm{T}}$ that contains the cost *paid* by node i to the PNs of each layer, the total cost paid by node i to its PNs in order to receive the video is given by

$$C_i = \mathbf{b}_i^{(\mathrm{r})} \mathbf{c}_i. \tag{10}$$

Similarly, if node i forwards the video layers to other nodes, it will be paid by its CNs. The reward of node i for forwarding layer l to its CNs $m \in \mathcal{M}_i^l$, is given by

$$r_i^l = \sum_{m \in \mathcal{M}_i^l} c_{i,m}^l. \tag{11}$$

Hence, by defining an $L \times 1$ vector $\mathbf{r}_i = [r_i^1, \ldots, r_i^L]^{\mathrm{T}}$ that contains the reward received by node i from the CNs of each layer, the total reward that node i obtains in this network is given by

$$R_i = \mathbf{b}_i^{(\mathrm{t})} \mathbf{r}_i. \tag{12}$$

The incentive received by node i due to transmission of layer l to its *CNs* is denoted by $I_i^{l,(\mathrm{c})}$ and given by

$$I_i^{l,(\mathrm{c})} = \sum_{m \in \mathcal{M}_i^l} I_{i,m}^l \tag{13}$$

in which $I_{i,m}^l$ is the incentive sent to PN i by CN m for layer l. The $L \times 1$ vector $\mathbf{I}_i^{(\mathrm{c})} = [I_i^{1(\mathrm{c})}, \ldots, I_i^{l,(\mathrm{c})}]^{\mathrm{T}}$ contains the incentive a node receives for forwarding each of the layers. Considering the reward and the incentive received by a node, we define the *virtual income* of a node as

$$V_i = \mathbf{b}_i^{(\mathrm{t})} \left(\mathbf{r}_i + \mathbf{I}_i^{(\mathrm{c})} \right). \tag{14}$$

In the next section, we propose a game theoretic framework for video dissemination and explain how the cost and incentive functions are calculated and how the incentive mechanism works.

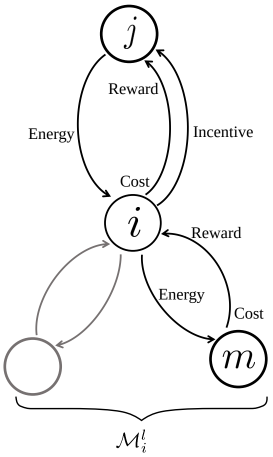

Fig. 1: The interactions of a node with its PN and CNs.

III. Proposed Video Dissemination Algorithm

A. Game theoretic Model

We propose a non-cooperative game model for video dissemination in the network. The players of the game are all the nodes of the network except the source, i.e., the elements of the set $\mathcal{P}$. The action of player i for layer l, denoted by a_i^l, is to choose a PN $j \in \mathcal{A}_i^l$ from whom it receives the video layer l. $\mathcal{A}_i^l$ is the action set of player i for layer l, consisting of the candidate parents of CN i that can transmit layer l to it such that we have $a_i^l \in \mathcal{A}_i^l$. The action set of all the nodes of the network except node i for layer l is given by $\mathbf{a}_{-i}^l$ The action profile of the game for layer l is denoted by $\mathbf{a}^l = (a_i^l, \mathbf{a}_{-i}^l) \in \mathcal{A}^l$ in which $\mathcal{A}^l = \mathcal{A}_1^l \times \mathcal{A}_2 \cdots \times \mathcal{A}_N^l$ is the joint action set of the game for layer l. From a tree-graph point of view, a node i cannot choose node m which is one of its descendants, as by doing so, a loop occurs in the broadcast tree and the connection between the node i and the source is lost [4]. We define set $\mathcal{R}_i^l$ as a set that contains the nodes on the path from the source to the node i. Thus, a node j can be candidate parent for node i if node i is not on the path of node j to the source for layer l. The action set of node i for layer l are the neighboring nodes of node i such that node i is not on their path to the source and defined as

$$\mathcal{A}_i^l = \left\{ j \middle| j \in \mathcal{N}_i^l, b_j^{l,(\mathrm{r})} = 1, i \notin \mathcal{R}_j^l \right\}, \tag{15}$$

in which $b_j^{l(\mathrm{r})} = 1$ states that node j possesses the layer l and can serve CN i. The set of actions of node i is the joint actions of node i for all the layers as

$$\mathbf{a}_i = \left\{ a_i^l \middle| a_i^l \in \mathcal{A}_i^l, 1 \leq l \leq L \right\}. \tag{16}$$

The proposed game is iterative and at each iteration of the game, one of the nodes of the network updates its action for all the layers $l \in \mathcal{L}$ at once. A utility function assigns a value to every node based on the action taken by the node such that $u_i^l(a_i^l, \mathbf{a}_{-i}^l) : \mathcal{A}^l \to \mathbb{R}_+, \forall i \in \mathcal{P}$ in which $u_i^l(a_i^l, \mathbf{a}_{-i}^l)$ is the utility of node i for layer l and $\mathbb{R}_+$ represents the positive real numbers. The game G is formally defined by the tuple $G =< \mathcal{P}, \{A_i^l\}_{i \in \mathcal{P}, l \in \mathcal{L}}, \{U_i\}_{i \in \mathcal{P}} >$.

The proposed game G is child-driven, that is, a node as a CN decides about $\mathbf{a}_i$ at every iteration. In other words, for a certain layer l, a node either refuses to receive the layer l, i.e., $b_i^{l,(\mathrm{r})} = 0$, and $a_i^l = \emptyset$, or it chooses the PN it wishes to receive the video layer from, i.e., $b_i^{l,(\mathrm{r})} = 1$, and $a_i^l = j, j \in \mathcal{A}_i^l$. In order to control the energy consumption in the network, the cost that a CN pays to its PN is defined based on the energy the PN spends to transmit the video layer. Thus, although a higher video layer results in a higher QoE for a CN, it results in a higher cost as it requires higher energy consumption at the PN.

The QoE of the node along with the reward and incentive it receives and the cost it pays are all captured by a utility function whose maximization determined the node's action.

B. Utility Function Definition

We first define the *profit function* of a node $i \in \mathcal{W}$ as

$$\Pi_i := \Pi_i(\mathbf{a}_i, \{\mathbf{a}_{-i}^l\}_{l \in \mathcal{L}}) = Q_i - \lambda_i C_i + \beta_i R_i \tag{17}$$

in which Q_i is the QoE of node i, C_i is the cost node i pays to its PNs in order to receive the video and R_i is the reward received by node i from other nodes for forwarding the video to them. λ_i in (17) is a pricing coefficient that depends on the preference of the user in spending cost for receiving higher video quality. In fact, a lower value for λ_i states that having higher QoE is more important for the user than the cost it should pay. Likewise, β_i in (17) reflects node i's willingness for contribution in the network. More precisely, higher values of β_i show that node i prefers to forward the video to others in order to get the reward from its CNs.

An important parameter that affects a node's decision, especially in user centric networks, is how the cost is assigned to a user. Indeed, having a fair cost allocation among the nodes of a multicast receiving group is of high importance. To this end, we use the Shapley value cost sharing rule [9] to determine the cost of each node in a multicast group of receiving nodes. The cost of each CN in $\mathcal{M}_j^l$ is defined based on the energy consumed by the PN j. Assuming that the required energy for every unicast link from the PN j to the multicast receiving nodes in $\mathcal{M}_j^l$ with M_j^l CNs are sorted as $0 \leq e_{j,1}^{l,\mathrm{uni}} \leq \cdots \leq e_{j,M_j^l}^{l,\mathrm{uni}}$ such that $e_j^l = e_{j,M_j^l}^{l,\mathrm{uni}}$, then the cost that CN i pays to PN j, i.e., $c_{j,i}^l$, is obtained by

$$c_{j,i}^l = c_i(j, \mathbf{a}_{-i}^l) = \sum_{h=1}^{i} \frac{e_{j,h}^{l,\mathrm{uni}} - e_{j,h-1}^{l,\mathrm{uni}}}{M_j^l + 1 - h}. \tag{18}$$

When the action set of node i for a certain layer is not empty, that means it is possible for node i to get this layer, the cost that node i pays in case of receiving this layer is the minimum possible cost. Otherwise, when there is no transmitting node to provide the layer for node i, the cost of the layer is set to infinity for node i, i.e.,

$$c_i^l = \begin{cases} \min\limits_{j \in \mathcal{A}_i^l} \quad c_{j,i}^l & \text{if } \mathcal{A}_i^l \neq \emptyset \\ \infty & \text{otherwise.} \end{cases} \tag{19}$$

Consequently, the action of node i is to choose the PN of each layer as

$$a_i^l = \begin{cases} \underset{j \in \mathcal{A}_i^l}{\operatorname{argmin}} \quad c_{j,i}^l & \text{if } b_i^{l,(\mathrm{r})} = 1 \\ \emptyset & \text{if } b_i^{l,(\mathrm{r})} = 0 \end{cases} \tag{20}$$

in which $b_i^{l,(\mathrm{r})}$ is the objective parameter determined by node i that shows whether it receives layer l or not.

So far, we have defined the profit function of a node based on its QoE, the cost it pays and the reward it receives. In defining the utility function of the node, whose maximization determines the node's action, we take the incentives into account. It is proposed that every node transfers a portion θ of its virtual income to its PNs as an incentive for forwarding the video to them. The value of θ is a design parameter and assumed to be fixed for all the nodes in the network independent of the user preference. More precisely, the incentive received by node i due to *transmission* of layer l of video to its CN m is given by

$$I_{i,m}^l = \theta V_m \tag{21}$$

Thus, the total incentive received by node i from its *CNs* for forwarding layer l is obtained by

$$I_i^{l,(\mathrm{c})} = \sum_{m \in \mathcal{M}_i^l} \theta V_m. \tag{22}$$

and $\mathbf{I}_i^{(\mathrm{c})} = [I_i^{1,(\mathrm{c})}, \ldots, I_i^{L,(\mathrm{c})}]^{\mathrm{T}}$ is defined as an $L \times 1$ vector containing the incentives node i receives for each of the layers $l \in \mathcal{L}$ from the CNs in $\mathcal{M}_i^l$. The total incentive received by node i from its CNs is then given by

$$I_i^{(\mathrm{c})} = \mathbf{b}_i^{(\mathrm{t})} \mathbf{I}_i^{(\mathrm{c})}. \tag{23}$$

Note that, a CN sends incentive just to the PNs of the layers which have been forwarded to others by it. For the layers that it does not forward, it just pays its cost. Similar to node i that receives reward and incentive from its CNs, node j, as the PN of node i, does so. The vector containing the incentives sent by node i to the PN of each layer is denoted $\mathbf{I}_i^{(\mathrm{p})} = [I_i^{1,(\mathrm{p})}, \ldots, I_i^{L,(\mathrm{p})}]^{\mathrm{T}}$ in which $I_i^{l,(\mathrm{p})} = \theta V_i$ if $b_i^{l,(\mathrm{r})} = 1$. The total incentive sent by node i is then given by

$$I_i^{(\mathrm{p})} = \mathbf{b}_i^{(\mathrm{r})} \mathbf{I}_i^{(\mathrm{p})} = \theta L_i^{(\mathrm{t})} V_i \tag{24}$$

Now we define the utility function of node $i \in \mathcal{P}$ as

$$U_i := U_i(\mathbf{a}_i, \mathbf{a}_{-i}) = \Pi_i - I_i^{(\mathrm{p})} + I_i^{(\mathrm{c})}. \tag{25}$$

Definition 1: Budget-balanced cost sharing rule: In game theory, a cost sharing rule is budget balanced if the sum of the cost allocated to the players of the game are equal to the budget spent by the seller. Shapley value is a budget-balanced cost allocation rule [11], i.e., in our case

$$e_i^l = \sum_{m \in \mathcal{M}_i^l} c_{i,m}^l. \tag{26}$$

Observation 1: The utility function of node $i \in \mathcal{P}$ can be written as

$$U_i = \sum_{l \in \mathcal{L}} b_i^{l,(\mathrm{r})} \left(\mathrm{q}^l - \lambda_i c_i^l \right) + b_i^{l,(\mathrm{t})} \left((\beta_i - \theta) e_i^l + (1-\theta) I_i^{l,(\mathrm{c})} \right). \tag{27}$$

Proof: Using the definition of Π_i and I_i in (17) and (24), the utility function of (25) can be written as

$$\begin{aligned} U_i &= \mathbf{b}_i^{(\mathrm{r})} \mathbf{q} + \beta_i \mathbf{b}_i^{(\mathrm{t})} \mathbf{r}_i - \mathbf{b}_i^{(\mathrm{r})} \lambda_i \mathbf{c}_i - \mathbf{b}_i^{(\mathrm{t})} \mathbf{I}_i^{(\mathrm{p})} + \mathbf{b}_i^{(\mathrm{t})} \mathbf{I}_i^{(\mathrm{c})} \\ &= \mathbf{b}_i^{(\mathrm{r})} \left(\mathbf{q} - \lambda_i \mathbf{c}_i \right) + \mathbf{b}_i^{(\mathrm{t})} \left(\beta_i \mathbf{r}_i + \mathbf{I}_i^{(\mathrm{c})} - \mathbf{I}_i^{(\mathrm{p})} \right). \end{aligned} \tag{28}$$

Using (24) and (14) into (28) and by expanding it we have

$$U_i = \sum_{l \in \mathcal{L}} b_i^{l,(\mathrm{r})} \left(\mathrm{q}^l - \lambda_i c_i^l \right) + + b_i^{l,(\mathrm{r})} \left(\beta_i r_i^l + I_i^{l,(\mathrm{c})} + \theta \left(r_i^l + I_i^{l,(\mathrm{c})} \right) \right). \tag{29}$$

Finally, as the cost sharing rule used in the network is budget-balanced, using (26) and (11) we have $r_i^l = e_i^l$ and thus, (29) can be written as

$$U_i = \sum_{l \in \mathcal{L}} b_i^{l,(\mathrm{r})} \left(\mathrm{q}^l - \lambda_i c_i^l \right) + b_i^{l,(\mathrm{t})} \left((\beta_i - \theta) e_i^l + (1-\theta) I_i^{l,(\mathrm{c})} \right)$$

■

C. Decision making by the players in two steps

Every node $i \in \mathcal{P}$ finds $b_i^{l,(\mathrm{r})}$ and $b_i^{l,(\mathrm{t})}$ of (27) in a way to maximize its own utility function. Using Fig. 1, we explain how the algorithm works based on the decisions of nodes i and j.

Every node solves its utility maximization problem two times with different constraints. At first, node i maximizes its utility function by finding the best PNs $j \in \mathcal{A}_i^l, \forall l \in \mathcal{L}$ based on the layers that are *currently available* at the its neighboring nodes. After finding the best PNs, node i joins the chosen PNs to receive the video from them. Then, node i assumes that all the layers of the video are available at all of its neighboring nodes, i.e, $b_j^{l,(\mathrm{r})} = 1, \forall j \in \mathcal{N}_i$ and solves the utility maximization problem again with this new constraint.

If the utility of node i in the latter case is higher than the former one, node i can increase its utility by receiving additional layers that are currently not available at its neighboring nodes. Hence, node i incentivizes the nodes which currently do not possess the layers it prefers to receive. More precisely, node i proposes an incentive equal to θV_i to the PN that it prefers to receive a video layers from. In such a case, we call node i a *potential CN* of PN j for layer l. Hence, PN j is motivated by node i to get the layer l from its candidate parents in $\mathcal{A}_j^l$ and serve CN i if this action increase node j's utility.

When it comes to node j to play the game, it first finds $I_i^{l,(\mathrm{c})}$ based on (22) for all the layers it *currently transmits* and the layers it does not currently transmit but there is a potential CN for them. By doing so, PN j knows that in case of receiving a certain video layer from one of its neighboring nodes, there are potential CNs that may get this layer from it and it can increase its virtual income in exchange. Then PN j, like node i, solves its optimization problem in two steps to find which of the video layers it should get and the same procedure is performed at every node.

The optimization problem at a node can be formulated as an integer programming problem as:

$$\max_{\mathbf{b}_i^{(\mathrm{r})},\mathbf{b}_i^{(\mathrm{t})}} \quad \sum_{l\in\mathcal{L}} b_i^{l,(\mathrm{r})}\left(\mathrm{q}^l - \lambda_i c_i^l\right) + b_i^{l,(\mathrm{t})}\left((\beta_i-\theta)e_i^l + (1-\theta)I_i^{l,(\mathrm{c})}\right) \quad (30a)$$

subject to:

$$b_i^{l,(\mathrm{r})} \le b_i^{l-1,(\mathrm{r})}, \qquad \forall l\in\mathcal{L} \quad (30b)$$

$$b_i^{l,(\mathrm{t})} \le b_i^{l,(\mathrm{r})}, \qquad \forall l\in\mathcal{L} \quad (30c)$$

$$\sum_{l\in\mathcal{L}} p_i^l \le P_i^{\max} \quad (30d)$$

$$b_i^{l,(\mathrm{r})}, b_i^{l,(\mathrm{t})} \in \{0,1\}. \quad (30e)$$

Equation (30b) indicates that to get a specific video layer, receiving the previous layers are necessary. Eq. (30c) ensures that node i receives a layer in order to transmit it to others and finally, Eq. (30d) indicates the transmit power constraint at node i.

A Nash equilibrium point is a solution concept of games at which none of the players can increase its utility by changing its decision unilaterally [11]. It is shown in [9] that the non-cooperative cost sharing game with Shapley value rule is a potential game for which the existence of a pure NE is guaranteed. In such games, the best response dynamics technique [11] can be exploited in order to reach to an NE point, such that, at every iteration of the game one of the nodes of the networks updates its action based on the action of other nodes taken on previous iteration.

IV. Simulation Results

A. Simulation Setup

In this section, we present the performance of the proposed algorithm. The nodes of the network are randomly distributed in a 500m $\times$ 500m squared area. The number of nodes varies between 10 and 30 and in each realization of the network, one of the nodes is randomly chosen as the source. The simple path-loss channel model is considered where the channel gain between the nodes i and j is obtained by $g_{i,j} = 1/(x_{i,j})^\alpha$ in which $x_{i,j}$ is the distance between the nodes i and j and α is the path loss exponent, set to 3. The maximum transmit power at the nodes is assumed to be $P_j^{\max} = 150$ mW, $\forall j \in \mathcal{W}$ and the noise power is equal to -90 dBm. The minimum required SNR at the receiving nodes is considered as $\gamma^{\mathrm{th}} = 10$ dB. The number of bits per symbol is set to be $n_b = 2$ with symbol duration T_s normalized to 1. The pricing values λ and β are set to 1, 2, respectively, and the the incentive parameter θ is $\theta \in \{0, 0.3, 0.5\}$ where $\theta = 0$ represents the non-incentive case. The energy required for the links are normalized to the maximum energy that can be spent for one symbol, i.e., $T_s P^{\max}$. The optimization problem of (30a) is solved using CVX[1] along with Gurobi[2] in MATLAB environment.

TABLE I: Video properties used for the simulation

layer l	Data rate: d^l (Mbps)	VQM: q^l	agg. Data rate (Mbps)	agg. VQM: Q
1	0.6980	0.2585	0.6980	0.2585
2	0.3508	0.1616	1.0487	0.4200
3	0.3829	0.1513	1.4316	0.5713
4	0.1969	0.1556	1.6285	0.7269
5	0.0784	0.1632	1.7069	0.8901
6	2.4739	0.0789	4.1808	0.9690
7	3.8541	0.0185	8.0349	0.9875
8	5.9385	0.0125	13.9733	1.0000

Regarding to measuring the QoE, the VQM values are used such that the VQM value of layer l, i.e., q^l is normalized between 0 and 1 and receiving all the the layers results in $\sum_{l\in\mathcal{L}} q^l = 1$, that is, the maximum possible video quality. The VQM values and the corresponding required data rate required for each layer used throughout the simulations are shown in Table. I. The values in Table I are average over the values of three videos encoded by scalable video Coding H.264/SVC [12]. The measurements are provided by xiph.org[3] and the videos are called *CrowdRun*, *BlueSky* and *ParkJoy*.

B. Results

We first depict the social welfare of the network. The social welfare is defined as the aggregated utility of the nodes in the network as $\mathrm{SW} = \sum_{i\in\mathcal{W}} U_i$. In Fig. 2, the social welfare is depicted for different number of nodes. It can be observed that by increasing the density of the network, the social welfare of the networks improves. This is because in a dense network, the distances between the transmitting and receiving nodes become shorter than the ones in sparse networks. With a shorter distance, the energy required at a transmitting node as well as the cost paid by the receiving node are lower. Therefore, lower cost paid to get the video increases the social welfare. For instance when there are 30 nodes in the network, about 10% improvement is achievable in the network with the proposed incentive mechanism compared to the case of non-incentive mechanism. It can also be observed in Fig. 2 that, by increasing the incentive coefficient θ, a higher social welfare is obtained. As discussed earlier, this is due to the fact that with higher incentive, more video enhancement layers are obtained by the nodes which are located relatively close to the source. In such a case, the social welfare improves as more nodes benefit from the enhancement layers and perceive higher quality of experience.

To have a better insight about how the proposed game-theoretic algorithm works, Fig. 3 shows the convergence of the algorithm over different iterations when there are 25 nodes in the network. Using the proposed incentive-based method,

[1] http://cvxr.com/cvx/
[2] http://www.gurobi.com/
[3] https://media.xiph.org/video/derf/

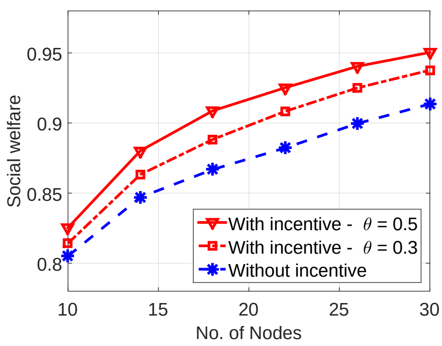

Fig. 2: Social welfare of the network for different number of nodes.

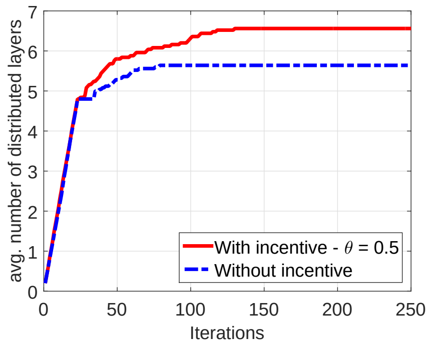

Fig. 3: Convergence of the mechanism to the NE. There are 25 nodes in the network.

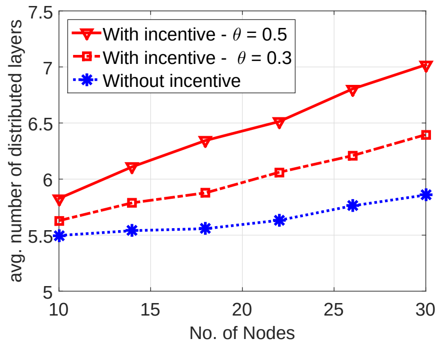

Fig. 4: Average number of video layers received by the nodes of the network.

the algorithm requires more iterations to converge. The game is played iteratively and the nodes update their actions until reaching the NE point at which no improvement is possible.

In Fig. 3, the first 25 iterations show the iterations in which the nodes join the network one after another. At the time of joining the network, a node has no CN and no incentive and thus, based on Fig. 3, the nodes receive less than 5 layers on average. After finding the initial PNs, the nodes update their decision to lower their own cost and get higher layers of video if possible. Therefore, the average number of received layers by nodes increases over the iterations until the convergence.

In Fig. 4, the average number of layers received by the nodes of the network is depicted. As can be seen, with the proposed algorithm, the average number of layers distributed in the network significantly increases. When there are 30 nodes in the network and $\theta = 0.5$, seven layers out of 8 layers are received by the nodes of the network. The average number of distributed layers increases when the network density increases, that confirms the result depicted in Fig. 2.

V. Conclusion

In this paper, we proposed an incentive mechanism for video dissemination in multi-hop wireless networks. The mechanism is based on a cross-layer model that takes the subjective QoE of the users from overlay and the energy consumption of transmission from underlay. The distributed video is encoded into multiple layers such that receiving higher video layers results in a higher video quality. In the proposed mechanism, the contributing node of the network will be paid by its respective receivers in case of forwarding the video to them. We suggested a game theoretic framework by which a node achieves its preferred video quality by maximizing its own utility function selfishly. We showed that the proposed algorithm motivates the nodes to actively contribute in the network that results in a higher quality of experience and social welfare for the whole network.

VI. Acknowledgment

This work has been funded by the German Research Foundation (DFG) as part of project B3 within the Collaborative Research Center (CRC) 1053 – MAKI. The authors would like to thank Dr. Matthias Wichtlhuber for helpful discussions.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2014–2019," *Cisco, Tech. Report*, 2015.

[2] M. Wichtlhuber, N. Aleksandrov, M. Franz, O. Hinz, and D. Hausheer, "Are incentive schemes needed for webrtc based distributed streaming?: A crowdsourced study on the relation of user motivation and quality of experience," in *Proceedings of the 7th International Conference on Multimedia Systems*, ser. MMSys '16, May 2016, pp. 19:1–19:12.

[3] A. Sinha, G. Paschos, and E. Modiano, "Throughput-optimal multi-hop broadcast algorithms," in *in Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '16. New York, NY, USA: ACM, 2016, pp. 51–60. [Online]. Available: http://doi.acm.org/10.1145/2942358.2942390

[4] M. Mousavi, H. Al-Shatri, M. Wichtlhuber, D. Hausheer, and A. Klein, "Energy-efficient data dissemination in ad hoc networks: Mechanism design with potential game,," in *IEEE 12th International Symposium on Wireless Communication Systems (ISWCS)*, August 2015.

[5] "Methods for objective and subjective assessment of quality," *ITU-T P.800*, Aug 1996.

[6] M. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting,*, vol. 50, no. 3, pp. 312–322, Sept 2004.

[7] K. D. Singh, K. Piamrat, H. Park, C. Viho, and J. M. Bonnin, "Optimising qoe for scalable video multicast over wlan," in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sept 2013, pp. 2131–2136.

[8] P. T. A. Quang, K. Piamrat, K. D. Singh, and C. Viho, "Video streaming over ad-hoc networks: a qoe-based optimal routing solution," *IEEE Transactions on Vehicular Technology*, vol. PP, no. 99, pp. 1–1, April 2016.

[9] M. Mousavi, S. Müller, H. Al-Shatri, B. Freisleben, and A. Klein, "Multi-hop data dissemination with selfish nodes: Optimal decision and fair cost allocation based on the shapley value," in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6.

[10] A. Bogliolo, P. Polidori, A. Aldini, W. Moreira, P. Mendes, M. Yildiz, C. Ballester, and J. Seigneur, "Virtual currency and reputation-based cooperation incentives in user-centric networks," in *Proc. 8th International Wireless Communications and Mobile Computing Conference (IWCMC),*, Aug 2012, pp. 895–900.

[11] Y. Shoham and K. Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008.

[12] M. Wichtlhuber, S. Dargutev, S. Müller, A. Klein, and D. Hausheer, "Qtrade: a quality of experience based peercasting trading scheme," in *2015 IEEE International Conference on Peer-to-Peer Computing (P2P)*, Sept 2015, pp. 1–10.

-