A Dec-POMDP Model for Congestion Avoidance and Fair Allocation of Network Bandwidth in Rate-Adaptive Video Streaming

Mahdi Hemmati, Abdulsalam Yassine, and Shervin Shirmohammadi
Distributed and Collaborative Virtual Environment Research (DISCOVER) Lab
University of Ottawa, Ottawa, Ontario, Canada
Email: mhemmati@uottawa.ca {ayassine | shervin}@discover.uottawa.ca

***Abstract*—We consider the problem of distributed rate adaptation among multiple video streaming sessions over the Internet from a decision-theoretic and computational intelligence point of view, and we design a multi-objective optimization model for network resources, seeking a fair and efficient distribution of end-users' Quality of Experience (QoE). A social welfare function is developed to capture both fairness and efficiency objectives at the same time. Then, assuming a common altruistic goal for all network users, we propose a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) model for finding the optimal network bandwidth allocation that leads to social welfare maximization. We show that the resulting optimal policy for the proposed model outperforms TCP-Friendly Rate Control (TFRC) protocol in terms of total utility and fairness.**

I. Introduction

VIDEO traffic on the Internet has been growing at a rapid pace during recent years. According to Cisco Visual Networking Index [1], video traffic will make up 80% of all consumer Internet traffic by 2019, up from 64% in 2014. An increasing fraction of this video traffic comes from video streaming services, such as live streaming, video-on-demand and over-the-top (OTT) video services. Examples are YouTube and Netflix, where the video has to be streamed in a continuous manner and without interruption in playback or degradation in quality, as much as possible. But doing so requires high bandwidth and low packet loss, which imposes new challenges to the existing best-effort Internet. These challenges are exacerbated by the fact that there are multiple video streams concurrently running on the network, potentially competing for the limited bandwidth. Hence, a fair and efficient video rate allocation model is required to 1) prevent congestion, and 2) provide a balanced video quality to all end users. Since there is no centralized authority for resource allocation in the Internet, a distributed solution for video rate adaptation is needed for congestion avoidance and bandwidth sharing among multiple video streams. The design of such a solution is a challenging problem in today's video delivery industry.

The earliest attempt to providing such a solution was TCP-Friendly Rate Control (TFRC) [2], in which the video sender infers the network condition from the estimated packet loss rates and delay metrics reported by the receiver via feedback packets. Naturally, this can only *react* to network congestion or packet loss and lacks a foresighted behavior. Some network-assisted approaches [3] have been proposed to fix these issues and improve agility in response to abrupt changes in traffic or network conditions. Although TFRC tries to provide fair bandwidth sharing to flows of different protocols, most existing congestion control solutions fail to provide a fair allocation of network bandwidth among competing video streams. Generally, fairness is not explicitly taken into account as an objective. Even when fairness is considered [4], it is about fair distribution of throughput, while the end users' quality-fairness, which is more desirable and has the ultimate impact on the human user, is typically ignored. In general, existing approaches suffer from one or more of the following four shortcomings [5]: 1) are generic protocols that are application-agnostic and do not take into account video quality, 2) do not capture the multi-agent nature of the problem, even though the problem clearly involves more than one utility-maximizing decision-maker interacting with each other, 3) do myopic adaptation only based on instantaneous rates, and 4) do not explicitly or correctly address fairness.

In this work, we address all of the above shortcomings and target a quality-driven end-to-end congestion control and bandwidth sharing mechanism. We propose a decision-theoretic model, called *Decentralized Partially Observable Markov Decision Process* (Dec-POMDP), to formulate the interaction of multiple concurrent video streaming sessions over the Internet. Aiming at maximizing the perceived quality of end-users while maintaining fairness in network bandwidth allocation, we employ a QoE model and introduce a *social welfare* function by combining the main objectives of *efficiency* and *fairness*. The solution of the proposed multi-agent decision process provides an optimal policy for all network users to adapt their streaming rates in the best interests of the entire network, leading to an optimum fair distribution of QoE among users. We evaluate the performance of this rate adaptation scheme through simulations, showing its advantages over the TFRC.

It should be mentioned that we previously presented a Dec-POMDP model for TCP-based video streaming in [5]. But that work suffered from large state-space dimension, which drastically increases the computational complexity of

978-1-4799-7560-0/15 $31.00 © 2015 IEEE
DOI 10.1109/SSCI.2015.170

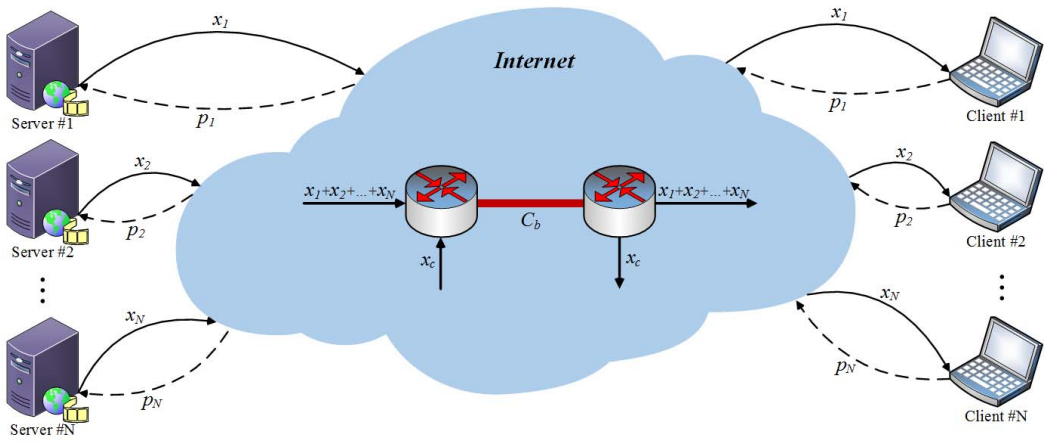

Figure 1. Schematic diagram of the problem showing N concurrent video streaming sessions

the problem. Furthermore, the binary observations in TCP (success or failure of packet delivery) makes it difficult for the learning agents to converge to the optimal policy. By replacing TCP with TFRC in the present work, we take advantage of the observation of packet loss rate in order to improve the model's capability for inferring the unobservable network congestion.

The rest of the paper is organized as follows. Section II provides a detailed description of the multi-user video streaming problem. We also explain the QoE model employed and the rate adaptation mechanism assumed for users. The concept of social welfare will also be developed in this section. The proposed Dec-POMDP model will be presented in section III, after introducing its mathematical definition and its components. Section IV will discuss the computationally intensive task of solving Dec-POMDP, before presenting the implementation details and evaluation results in section V, which compares the performance of the proposed model with TFRC, in terms of total QoE and fairness. Final remarks and future research avenues concludes the paper.

II. Problem Description

We consider the problem of network bandwidth sharing by several video streaming sessions over the Internet, seeking a fair and efficient distribution of Quality of Experience (QoE) from the media consumers' perspective. All network users are supposed to share a common altruistic goal to maximize some notion of social welfare. A dynamic rate adaptation scheme is also assumed to be implemented by the media servers.

A. Network Model

Consider N concurrent video streaming sessions over the Internet, sharing the bandwidth of the network. Each session is composed of a sender node (media streaming server) and a receiver node (client) that establish an end-to-end transport layer connection, equipped with *TCP-Friendly Rate Control* (TFRC) protocol [2] to stream a multimedia content. We assume that the network has a single bottleneck link that causes packet loss once congested. Since the sender-side of each session cannot observe the traffic generated by other sessions, it is only able to infer the congestion status based on the feedback information received from the network or receiver-side. The major source of information about network congestion level is the receiver's estimate of packet loss rate, which is included in TFRC feedback packets.

As depicted in Figure 1, each session n chooses its transport-layer sending rate x_n at sender-side and receives an estimate of the packet loss rate p_n from the receiver-side. The bottleneck link of the network, with a capacity of C_b, should handle the total sum of sending rates plus a time-varying cross-traffic x_c. As the total traffic on bottleneck link comes close to its capacity, all sessions would experience higher rates of packet loss. Therefore, a distributed rate control and congestion avoidance scheme, capable of dealing with this dynamic and partially observable environment is required.

B. Utility Model: QoE

The utility of users is their ultimate Quality of Experience (QoE). Since QoE is a subjective measure of quality which is not readily available in real-time, an automatic prediction method is required to map the network's Quality of Service to user's QoE. We adopt the G.1070 opinion model recommended by ITU-T [6] as the video quality model for predicting the subjective quality measured in *Mean Opinion Score* (MOS). In multimedia communications, the MOS provides a numerical measure of the perceived quality from the users' perspective of received media at the destination. It is expressed as a single number in the range 1 to 5, where 1 corresponds to the lowest perceived quality, and 5 to the highest.

There are several factors affecting the perceived quality of a video: codec type and specifications, spatial resolution, key frame interval, delay, frame rate, bit rate and packet loss rate. Among these, we assume that all are fixed (or have negligible impact on video quality) during a streaming session except the last two: encoding bit rate (r) and packet loss rate (p). Under such conditions and according to G.1070 quality model, the utility (u) or QoE of the user could be summarized as:

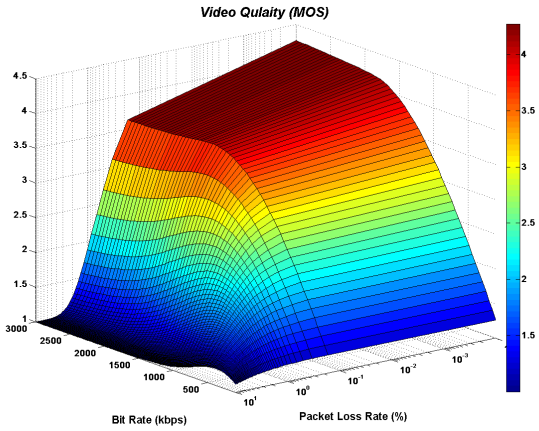

Figure 2. Video QoE based on ITU-T Recommendation G.1070

$$u = 1 + \left[c_1 \left(1 - \frac{1}{1 + (\frac{r}{c_2})^{c_3}} \right) \right] e^{-\frac{p}{d}} \quad \in [1, 5] \qquad (1)$$

where $c_1, c_2,$ and c_3 are codec-dependent constants and d is the degree of robustness against packet loss and is itself a function of bit rate, frame rate and a number of codec-dependent constants.

A visualization of the above QoE model and its variations with respect to bit rate and packet loss is shown in Figure 2. It is clearly observed that a significant degradation of video quality occurs at packet loss rates above 0.05%, even with a pretty high source bit rate. Therefore, a QoE-aware rate adaptation scheme at the sender-side should not only try to maximize its bandwidth utilization, but also be able to proactively avoid congestion leading to higher packet loss rates.

C. Rate Adaptation Mechanism

State-of-the-art adaptive video streaming [7] [8] embed the rate adaptation algorithm inside the client application. This allows the client to independently choose the playback quality without any need for intelligent components inside the network. However, both industry [9] and academia [10] are showing interest in server-side or network-assisted adaptive streaming. In this study, we are considering a *server-based adaptive streaming*.

The adaptation mechanism is to be done jointly at both the application and the transport layers. Video source rate control would be carried out by the application layer, while a TFRC-like congestion control scheme is performed at the transport layer to take care of the packet loss rate. In TFRC, the sending rate x reacts to variation of packet loss rate p using the following formula:

$$x = g(p) = \frac{\ell}{RTT} \sqrt{\frac{3}{2p}} \qquad (2)$$

where ℓ represents the packet size and RTT is the round-trip time.

Similar to TFRC, our server-based adaptation mechanism includes a rate switching scheme at transport layer, which uses a mapping from observed packet loss rates to optimal sending rates. But unlike TFRC, this adaptation is not tied to instantaneous values; rather it takes into account the history of observations and tries to do a foresighted optimization.

At the application layer, the source rate r (the target rate of the live video encoder or the source rate of pre-encoded videos) is adjusted according to the chosen sending rate using the sending buffer level:

$$r = x + \delta \left(B - \frac{B_u + B_l}{2} \right) \qquad (3)$$

where B is the current buffer level, B_u and B_l are the upper and lower limits of the buffer and δ is the adjustment rate. The buffer's input and output rates are r and x, respectively. Using this rate adjustment method, the source rate would closely follow the sending rate, especially at steady state. Therefore, we might use them interchangeably in some approximations later on.

D. Social Welfare: Efficiency + Fairness

We assume that all users of the network are programmed to behave altruistically, as opposed to selfishly, although they all operate independently without any type of communications or information exchange. In other words, the geographically distributed users of the network not only share common resources, but also share a common objective function, called *social welfare*, which captures both *efficiency* and *fairness*.

By efficiency, we mean maximizing total utility of all users. Fairness, on the other hand, could have many different interpretations and criteria [11]. Various fairness measures have been proposed across different scientific disciplines, ranging from the simple ratio between the smallest and the largest entries, to more sophisticated functions like Jain's index [12], which is very popular in network resource allocation. An axiomatic theory of fairness was constructed by [13] in an attempt to formalize the notion of fairness. This work showed that all fairness measures satisfying five basic axioms, form a unique family of functions $f_\beta(\cdot)$ parametrized by β, where the Jain's index corresponds to the special case of $\beta = -1$.

We define the social welfare function as a weighted sum of the network's total utility and some fairness measure of the distribution of utility, namely Jain's index. Let $\mathbf{u} = [u_1, u_2, \ldots, u_N]^T$ be the vector of utilities (*i.e.* QoE's) achieved for all N users of the network. We apply Jain's fairness function to the allocation vector and combine it with the sum $\sum_{n=1}^{N} u_n$ as a measure of efficiency in order to construct a scalar metric for maximization of both objectives:

$$\Phi(\mathbf{u}) = \log \left(\sum_{n=1}^{N} u_n \right) + \lambda \log \left(J(\mathbf{u}) \right) \qquad (4)$$

where

$$J(\mathbf{x}) = \frac{\left(\sum_{i=1}^{N} x_i \right)^2}{N . \sum_{i=1}^{N} x_i^2} \in [0, 1] \qquad (5)$$

is the well-known Jain's index for distributive fairness [12] and λ serves as the relative importance of our two objectives.

It could be shown [13] that there is an upper bound for λ in order for the welfare function to preserve the common sense of Pareto dominance. It turns out that λ should be less than or equal to $\bar{\lambda} = \left|\frac{\beta}{1-\beta}\right|$, which would be equal to $1/2$ for the case of Jain's index. Replacing $\lambda = 1/2$ in equation 4, our social welfare function could be written as

$$\Phi(\mathbf{u}) = \log\left(\frac{\left(\sum_{n=1}^{N} u_n\right)^2}{\sqrt{N \cdot \sum_{n=1}^{N} u_n^2}}\right). \tag{6}$$

III. Dec-POMDP Model

Optimal sequential decision making under uncertainty have been extensively studied in artificial intelligence [14] [15] and stochastic control [16] literature. The basic theoretical foundations of this area are the concept of *state* and the *Markov property* –postulating that the future states of the stochastic process depend only on the present state, not on the past history of events. *Markov Decision Process* (MDP) [16] models decision problems under uncertainty when the full state information is available. In many real world problems this is not the case and only incomplete state information might be observable. *Partially Observable Markov Decision Process* (POMDP) [17] provides a powerful modeling framework for such problems.

Our adaptive video streaming problem is a decision making problem with Markov property and partially observable information about the network state. However, since there are several active decision-makers interacting with the network, a decentralized or multi-agent modeling tool would be required. In this section, we first introduce the Dec-POMDP framework and then present the proposed decision process model for rate-adaptive video streaming.

A. Overview of Dec-POMDP Framework

Here we provide a formal definition of the employed modeling framework, which is an extension of single-agent POMDP to a multi-agent cooperative setting. A *Decentralized Partially Observable Markov Decision Process (Dec-POMDP)* is defined as a tuple $\langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{O}, T, O, R, h, I \rangle$, where

- $\mathcal{N} = \{1, 2, \ldots, N\}$ is the set of N agents.
- $\mathcal{S}$ is the finite set of states $\mathbf{s}$ in which the environment can be.
- $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$ is the finite set of joint actions of all agents $\mathbf{a} = \langle a_1, \ldots, a_N \rangle$, where an individual action of an agent $n \in \mathcal{N}$ is denoted by $a_n \in \mathcal{A}_n$.
- $\mathcal{O} = \mathcal{O}_1 \times \cdots \times \mathcal{O}_N$ is the finite set of joint observations $\mathbf{o} = \langle o_1, \ldots, o_N \rangle$, where an individual observation of an agent $n \in \mathcal{N}$ is denoted by $o_n \in \mathcal{O}_n$.
- T is the transition function that provides $P(\mathbf{s}'|\mathbf{s}, \mathbf{a})$, the probability of transition to a next state $\mathbf{s}'$ given that joint action $\mathbf{a}$ is executed at state $\mathbf{s}$.
- O is the observation function that specifies $P(\mathbf{o}|\mathbf{a}, \mathbf{s}')$, the probability that the agents receive joint observation $\mathbf{o}$ of state $\mathbf{s}'$, when they reached this state through joint action $\mathbf{a}$.
- R is the common immediate reward function which depends on the state of the environment and actions of all agents. $R(\mathbf{s}, \mathbf{a})$ specifies a real number as the common reward for all agents.
- h is the time horizon of the problem, which could be either finite or infinite.
- $I \in \mathcal{P}(\mathcal{S})$ is the initial probability distribution of the state, where $\mathcal{P}(\cdot)$ denotes the set of probability distributions over its argument.

At each time step, also known as *stage*, the agents simultaneously take an action. The resulting joint action provides a common reward based on the current state. It also causes a stochastic transition to the next state, of which a joint observation is emitted by the environment and each agent observes its own component.

B. Proposed Model for Rate-Adaptive Video Streaming

Our proposed Dec-POMDP model is specified by the following components:

- **Agents**
  The dynamic interaction is taking place among a finite number of video streaming sessions, regarded as agents or users, indexed by $n \in \mathcal{N} = \{1, 2, \ldots, N\}$. Each streaming session comprises a sender and a receiver node in the network. However, since we are considering a sender-based adaptive streaming, the agents of the model are actually the sending entities of the video streaming process. The receiver nodes help provide the observations to the senders by sending back TFRC-like feedback packets. This multi-agent decision process runs over the course of a sequence of discrete time steps indexed by $k = 0, 1, 2, \ldots$. We assume a fixed *Round-Trip Time* (RTT) for the network, and set the time step of Dec-POMDP to be equal to one RTT.
- **Actions**
  We take the *sending rate* (x) of each video stream at the transport layer as the action of the agent. Note that the *source encoding rate* (r) of the video streams are closely tied to the chosen sending rate. We assume that in live video streaming, the target encoding rate of the video stream is adjusted at the application layer, according to the sending rate, using the sending buffer level. For the case of on-demand streaming, the video is supposed to be pre-encoded at corresponding source rates as well.
  Formally speaking, the action taken by user n at a given time step k is to choose the appropriate sending rate $a_n^k = x_n^k \in \mathcal{A}_n$, where $\mathcal{A}_n$ is the set of discrete pre-defined sending rates available for user n. The joint action of all users would be denoted by $\mathbf{a}^k = \langle a_1^k, \ldots, a_N^k \rangle \in \mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$.
- **States**
  The state of the proposed Dec-POMDP is set to be the

unobservable *congestion level* of the network. As mentioned before, we assume that there is a single bottleneck link within the network that determines the congestion status of the network for the video streaming scenario under study.

The congestion level, denoted by C_g, takes values in the interval of $[0, 1]$ and is defined as the ratio of total traffic on the bottleneck link to its capacity:

$$C_g = \min\{\frac{x_c + \sum_{n=1}^{N} x_n}{C_b}, 1\}, \quad (7)$$

where C_b is the capacity of the bottleneck link, x_n's are the sending rates of the concurrent video streaming sessions, and x_c is the rate of the cross-traffic on the bottleneck link.

Since the state space is a finite set in the formalism of MDP, we define the state of our model at time step k, as $\mathbf{s}^k = C_g^k \in \mathcal{S}$, where $\mathcal{S}$ is a discretization of the interval of $[0, 1]$ into a finite number of possibilities.

It is clear that the congestion level of the network is not directly observed by the users. Therefore, we are dealing with a Partially Observable MDP (POMDP) from the viewpoint of each user, where it has to infer the state of the decision process using its limited observations.

- **Observations**

  As mentioned above, each user observes neither the traffic generated by other users, nor the congestion level of the network, which is impacted by the overall traffic transmitted over the bottleneck link. The only available piece of information about the network condition is the TFRC-like feedback packets received per RTT. By means of each feedback packet, the sender observes the receiver's estimate of the *Packet Loss Rate (PLR)*, denoted by $\hat{p}$, which is an indication of how congested the bottleneck router of the network is.

  We define the observation of user n at time step k as $o_n^k = \hat{p}_n^k \in \mathcal{O}_n$, where $\mathcal{O}_n$ is a discretized set of possible values for *PLR* estimates.

- **Transition Function**

  In order to specify the transition function: $T : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$, which provides a probability distribution of the new state given the current state and the joint action of the users, we need a probabilistic model for the cross-traffic over the bottleneck link. If the joint action of the users at time step k is $\mathbf{a} = \langle x_1^k, x_2^k, \ldots, x_N^k \rangle$, the probability that the new state C_g^k falls into the interval $[\alpha, \beta]$, denoted by:

  $$P\left(\alpha \leq C_g^k \leq \beta\right) = P(\alpha \leq \frac{x_c^k + \sum_{n=1}^{N} x_n^k}{C_b} \leq \beta), \quad (8)$$

  would translate to the probability that the cross-traffic fall into a corresponding interval:

  $$P\left(\alpha \cdot C_b - \sum_{n=1}^{N} x_n^k \leq \ x_c^k \ \leq \beta \cdot C_b - \sum_{n=1}^{N} x_n^k\right) = P\left(\tilde{\alpha} \leq x_c^k \leq \tilde{\beta}\right). \quad (9)$$

  We use the results of [18] for modeling the Internet backbone traffic at the flow level. According to their model, since the total cross-traffic is the result of a number of flows with independent rates, the central limit theorem asserts that the distribution of the cross-traffic tends to be Gaussian at high loads, which is typical of backbone links. The mean and variance of the rate of the cross-traffic are also calculated in this model in terms of the average size and duration of the contributing flows. Having specified the probability distribution function of the cross-traffic, we can calculate the transition probabilities for each joint action using equation (9).

  The transition probabilities depend on not only the joint action of the users, but also on the current state of the network. The auto-correlation function of the cross-traffic stochastic process induces this dependency since it restrains abrupt changes in the rate of the cross-traffic and accordingly the congestion level.

- **Observation Function**

  The observation function: $O : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{O})$ determines the probability of each joint observation $\mathbf{o}$ if a particular joint action $\mathbf{a}$ is taken that leads to a new state $\mathbf{s}'$. We assume that the probability of observing a certain rate of packet loss by each user is solely dependent on the network's state (congestion level) and independent of the users' actions (sending rates). This is justified by noting that the congestion level itself is caused by and derived from total sending rate generated by users plus cross-traffic rate. We also assume that the observation of each agent is statistically independent of other's observations. Therefore the probability of each joint observation would be equal to the product of individual probabilities.

- **Reward Function**

  The common goal of all agents is to achieve and maintain an optimal allocation of the network bandwidth leading to an efficient and fair distribution of end-users' QoE. Using the social welfare function defined in equation (6), we can specify the common reward function of our Dec-POMDP model as:

  $$R = \mathbb{E}\left[\sum_{k=1}^{h} \gamma^k \ \Phi(\mathbf{u}^k)\right] \quad (10)$$

  where γ is the discount factor (set to one in this model) and $\mathbf{u}^k = [u_1^k, u_2^k, \ldots, u_N^k]^T$ is the vector of utilities (*i.e.* QoE's) achieved for all N users at time step k as defined by equation (1). Note that the QoE of the n^{th} user u_n^k is in turn a function of its source rate r_n^k and experienced packet loss rate p_n^k. Since the video source rate r_n^k is adjusted to the sending rate x_n^k, and the packet loss rate p_n^k is determined by the network congestion level C_g^k, the common reward function of equation (10) would reduce to a function of joint action and state: $R(\mathbf{a}, \mathbf{s})$.

- **Horizon**

  The choice of time horizon in sequential decision problems is very critical. On one hand, it has to be large enough in order to fulfill the objective of foresighted (as

opposed to myopic) optimization. On the other hand, it has to be chosen small enough for the problem to remain computationally solvable. We chose $h = 5$ for our model.

- **Initial State Distribution**
  We assume that network congestion is not at the extreme low or extreme high levels at the beginning of the video streaming service, and is uniformly distributed over the rest of the interval.

IV. Optimal Solution of Dec-POMDP

A policy in a fully observable MDP is a mapping from states to actions. The policy of a user is comprised of a sequence of actions selected by the user at every time step based on the state of the environment. In selecting the actions, the agent can ignore the history because of the Markov property. In a POMDP [17], the agent does not observe the state, but it can compute a *belief* that summarizes the history and works as a Markovian signal.

In a Dec-POMDP [19], however, each agent will only have access to its individual actions and observations during execution and there is no method known to summarize this individual history. It is not possible to maintain and update an individual belief in the same way as in a POMDP, because the transition and observation functions are specified in terms of joint actions and observations. The consequence of this lack of access to a Markovian signal is that planning for Dec-POMDPs involves searching the space of tuples of individual policies that map full-length individual histories to actions.

Solving a Dec-POMDP is a really challenging task. In fact, it is known that the problem of finding the optimal solution for a finite-horizon Dec-POMDP with even only two agents is NEXP-complete [20]. In practice, this means that solving a Dec-POMDP takes doubly exponential time in the worst case. Moreover, efficient approximation of Dec-POMDP is not easily possible, and even finding an ϵ-approximate solution is NEXP-complete [21].

Since the number of joint policies in a Dec-POMDP grows exponentially with the number of possible observations, a brute force search would only be suitable for very small problems. Therefore, so much effort has been spent by researchers during last decade to create efficient methods for finding exact or approximate solution of Dec-POMDP. [22] provides a recent survey of the existing methods.

For our work, we use the *Joint Equilibrium based Search for Policies* (JESP) [23], which is guaranteed to find a locally optimal joint policy. It relies on a procedure called *alternating maximization*, that computes a maximizing policy for one agent at a time, while keeping the policies of the other agents fixed. This process is repeated until the joint policy converges to a Nash equilibrium: a tuple of policies such that for each agent's policy is a best response to the policies employed by the other agents. JESP uses a dynamic programming approach to compute the best-response policy for a selected agent, using a reformulation of the problem as an augmented POMDP by fixing the other agent's policies.

```
# Dec-POMDP Model for Adaptive Video Streaming
#-------------------------------------------
#Agents
#------
agents: 2
#Discount factor
#---------------
discount: 1.0
#Type of Values
#--------------
values: reward
#States (Congestion Level)
#-------------------------
states: CgLL CgL CgM CgH CgHH
#Initial state distribution
#--------------------------
start exclude: CgLL CgHH
#Actions (Sending Rate kbps)
#---------------------------
actions:
R1M R2M R3M
R1M R2M R3M
#Observations (Packet Loss Rate)
#-------------------------------
observations:
pLL pL pM pH pHH
pLL pL pM pH pHH
#Transition Probabilities
#------------------------
# T: <a1 a2> : matrix of %f for all <s> & <s'>
#   CgLL    CgL     CgM     CgH     CgHH
T: R1M R1M :
   0.3845  0.5515  0.0635  0.0005  0.0000
   0.0671  0.8661  0.0665  0.0003  0.0000
   0.0642  0.5527  0.3820  0.0011  0.0000
   0.1363  0.5864  0.2703  0.0070  0.0000
   0.1584  0.6818  0.1571  0.0027  0.0000
...
T: R3M R3M :
   0.0000  0.0027  0.1571  0.6818  0.1584
   0.0000  0.0070  0.2702  0.5865  0.1363
   0.0000  0.0011  0.3820  0.5527  0.0642
   0.0000  0.0003  0.0665  0.8661  0.0671
   0.0000  0.0005  0.0635  0.5515  0.3845
#Observation Probabilities
#-------------------------
# O: <a1 a2> : <s'> : <o1 o2> : %f
O: * * : CgLL : pLL pLL : 0.8191
O: * * : CgLL : pLL pL : 0.0814
O: * * : CgLL : pLL pM : 0.0045
O: * * : CgLL : pLL pH : 0.0000
O: * * : CgLL : pLL pHH : 0.0000
...
O: * * : CgHH : pHH pHH : 0.8191
#Rewards
#-------
# R: <a1 a2> : <s> : <s'> : <o1 o2> : %f
R: R1M R1M : CgLL : * : * : 1.8855
R: R1M R1M : CgL : * : * : 1.8831
R: R1M R1M : CgM : * : * : 1.8591
R: R1M R1M : CgH : * : * : 1.6343
R: R1M R1M : CgHH : * : * : 0.7411
...
R: R3M R3M : CgHH : * : * : 0.6932
```

Listing 1. Description of Dec-POMDP Model in a Text Format

V. Implementation and Evaluation

In order to find the optimal policy for our proposed Dec-POMDP model, we used *Multi-Agent Decision Process* (MADP) Toolbox [24], which provides software tools for modeling, specifying, planning and learning a variety of decision-theoretic problems in multi-agent systems. This toolbox includes a *parser* for reading text descriptions in Tony Cassandra's POMDP file format (`.pomdp`) [25] and its Dec-POMDP extension (`.dpomdp`).

The description of our Dec-POMDP model for two agents is shown in Listing 1. The state space (interval of $[0, 1]$ for congestion level) is discretized into five bands from very low to very high, namely `CgLL`, `CgL`, `CgM`, `CgH`, `CgHH`. Initial distribution of the state is assumed to be uniform over all states excluding the extreme cases: `CgLL`, `CgHH`. Both agents, assumed to be homogeneous in terms of their actions and observations, have three choices for their actions at each time step: sending video packets at 1, 2, or 3 Mbps rates, denoted by `R1M, R2M, R3M`, respectively.

The observation space (PLR estimates) is also discretized into five levels in logarithmic scale: $\texttt{pLL} \approx 10^{-5}$, $\texttt{pL} \approx 10^{-4}$, $\texttt{pM} \approx 10^{-3}$, $\texttt{pH} \approx 10^{-2}$, and $\texttt{pHH} \approx 10^{-1}$.

For each pair of joint action $\mathbf{a}$, a $|\mathcal{S}|\times|\mathcal{S}|$ matrix specifies the transition probabilities from start sate $\mathbf{s}$ to end state $\mathbf{s}'$. The probability values are calculated as described in section III. For the sake of brevity, not all matrices are shown in Listing 1; only the first and the last cases. As mentioned before, the probability of each observation only depends on the network state and is independent of the actions taken by the users and observations of other agents. Based on this assumption, the probability of occurrence of all possible joint observations are calculated and specified in the description of the model.

The reward model, as described in section III-B, only depends on the joint action and state: $R(\mathbf{a}, \mathbf{s})$. The calculated reward values for all combinations of actions and states constitute the last part of our model description. Note that only a few lines of the observation probabilities and rewards specification are included in Listing 1 due to limited space.

We used the JESP algorithm implemented in MADP Toolbox to solve our Dec-POMDP model. The result of JESP planning is a deterministic policy which maps every possible sequence of observations (PLR) with different lengths to an optimal action (sending rate). This policy is guaranteed to yield a local maximum of expected common reward, which provides both efficiency and fairness according to equation (6). Using a simple look-up table, the resulting optimal policy could be hard-coded into the transport layer protocol of the media streaming servers to replace/augment TFRC.

In order to evaluate our proposed rate adaption scheme, we compare its performance with that of TFRC in terms of total utility of the users as well as the fairness index of the distribution of QoE. Since there are stochastic components in our problem formulation and proposed model, we conducted 50 rounds of simulations and calculated the mean values to cross out the random effects. Figure 3 illustrates one sample of these simulations for each of the methods. We can see that the sending rates chosen by TFRC tend to oscillate between minimum and maximum values due to the reactive nature of this mechanism, whereas the foresighted optimization of Dec-POMDP model provides less switching in sending rate, lower congestion levels and higher common reward at the same time.

Table I
Comparison of the proposed solution with TFRC based on the average results of 50 runs of simulations

Metric	Dec-POMDP	TFRC	Improvement
Total QoE	7.452	5.975	24.7%
Fairness Index	0.881	0.780	12.9%
Social Welfare	1.946	1.663	17.0%

Table I shows the average results of 50 runs of simulations for three main metrics: total utility of users, fairness of users' QoE, and the social welfare function (equation (6)). Simulation results confirm that the optimal solution of our proposed Dec-POMDP model outperforms TFRC congestion control mechanism both in terms of efficiency and fairness.

VI. Conclusion

The problem of network bandwidth sharing among multiple video streaming sessions was considered from a decision-theoretic and computational intelligence point of view. A Dec-POMDP model was proposed to capture the multi-agent aspects of the dynamic interaction between network users. A common objective function, called social welfare, which incorporates maximization of total utility while achieving a fair distribution of QoE, was designed to be collectively optimized by different users. The solution of this sequential decision-making process provides an optimal policy for each agent to adapt its sending rate based on the sequence of packet loss rate observations. The optimal solution of Dec-POMDP induces an implicit cooperation among non-communicating network users, resulting in a much higher total QoE for users as well as improved fairness, in contrast to the popular TFRC.

For the next steps, we are currently working on developing a model-free *Multi-Agent Reinforcement Learning* (MARL) [26] algorithm for finding the optimal policy of users on-the-go. *Learning*, as opposed to *planning*, refers to the process of acquiring knowledge about optimal policy in a decision process, without having access to the model of the underlying dynamics. In the present study, we used JESP planning for computing the optimal policy, assuming that the Dec-POMDP model is known to the network protocol designer. We are investigating the possibility for the individual network users to learn the optimal policy on their own through interacting with the network.

References

[1] "Cisco Visual Networking Index: Forecast and Methodology, 2014-2019," 2015.

[2] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification (Proposed Standard)," 2008. [Online]. Available: https://tools.ietf.org/html/rfc5348

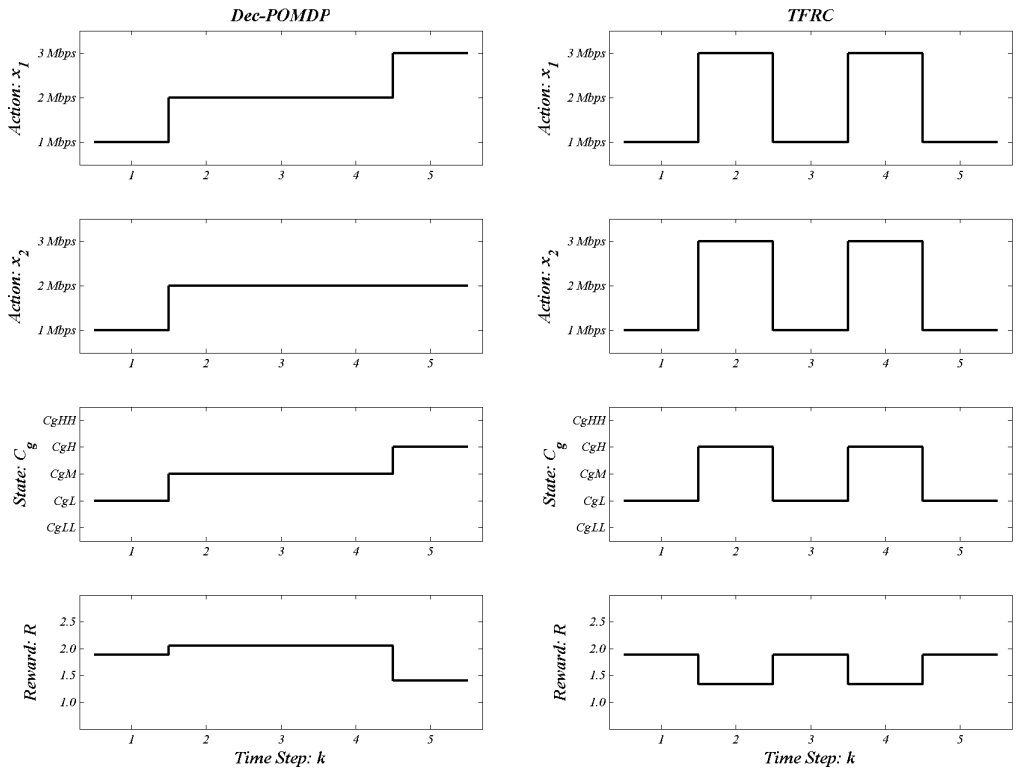

Figure 3. Samples of optimal Dec-POMDP solution and TFRC sending rates and the resulting network congestion and social welfare

[3] X. Zhu, R. Pan, M. S. Prabhu, N. Dukkipati, V. Subramanian, and F. Bonomi, "Layered Internet video adaptation (LIVA): Network-assisted bandwidth sharing and transient loss protection for video streaming," *IEEE Transactions on Multimedia*, vol. 13, no. 4, pp. 720–732, 2011.

[4] J. Jiang, V. Sekar, and H. Zhang, "Improving Fairness, Efficiency, and Stability in HTTP-Based Adaptive Video Streaming With FESTIVE," *IEEE/ACM Transactions on Networking*, vol. 22, no. 1, pp. 326–340, 2014.

[5] M. Hemmati, A. Yassine, and S. Shirmohammadi, "An Online Learning Approach to QoE-Fair Distributed Rate Allocation in Multi-User Video Streaming," in *Proc. Int. Conf. on Signal Processing and Communication Systems (ICSPCS), Australia*, 2014.

[6] "ITU-T Recommendation G.1070 Opinion model for video-telephony applications," 2012.

[7] A. C. Begen, T. Akgul, and M. Baugher, "Watching video over the web: Part 1: Streaming protocols," *IEEE Internet Computing*, vol. 15, no. 2, pp. 54–63, 2011.

[8] Q. I. Stockhammer, Thomas, "Dynamic Adaptive Streaming over HTTP – Design Principles and Standards," in *Proc. ACM Conf. on Multimedia Systems*, 2011, pp. 133–134.

[9] Huawei Technologies Co., "Server Management in Adaptive Streaming on the Internet," in *Proc. 4th W3C Web TV Workshop*, 2014, pp. 1–2.

[10] N. Bouten, S. Latré, and J. Famaey, "In-Network Quality Optimization for Adaptive Video Streaming Services," *IEEE Transactions on Multimedia*, vol. 16, no. 8, pp. 2281–2293, 2014.

[11] C. Joe-Wong, S. Sen, T. Lan, and M. Chiang, "Multi-Resource Allocation: Fairness-Efficiency Tradeoffs in a Unifying Framework," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1206–1214.

[12] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System," Tech. Rep., 1984.

[13] T. Lan, D. Kao, M. Chiang, A. Sabharwal, and M. Hiang, "An Axiomatic Theory of Fairness in Network Resource Allocation," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[14] A. G. Barto, R. S. Sutton, and C. Watkins, "Learning and sequential decision making," University of Massachusetts, Tech. Rep., 1989.

[15] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. The MIT Press, 1998.

[16] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995, vol. I-II.

[17] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, May 1998.

[18] C. Barakat, P. Thiran, G. Iannaccone, C. Diot, and P. Owezarski, "Modeling Internet backbone traffic at the flow level," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2111–2124, 2003.

[19] F. a. Oliehoek, "Decentralized POMDPs," in *Reinforcement Learning: State-of-the-Art*, M. Wiering and M. V. Otterlo, Eds. Springer, 2012, pp. 471–503.

[20] D. S. Bernstein, S. Zilberstein, and N. Immerman, "The Complexity of Decentralized Control of Markov Decision Processes," in *Proc. Uncertainty in Artifical Intelligence*, 2000, pp. 32–37.

[21] Z. Rabinovich, C. V. Goldman, and J. S. Rosenschein, "The Complexity of Multiagent Systems: The Price of Silence," in *Proc. AAMAS*, 2003, pp. 1102–1103.

[22] C. Amato, G. Chowdhary, A. Geramifard, N. K. Ure, and M. J. Kochenderfer, "Decentralized Control of Partially Observable Markov Decision Processes," in *Proc. IEEE CDC*, 2013, pp. 2398–2405.

[23] R. Nair, M. Tambe, M. Yokoo, D. Pynadath, and S. Marsella, "Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings," in *Proc. IJCAI*, 2003, pp. 705–711.

[24] F. A. Oliehoek, M. T. J. Spaan, and P. Robbel, "MultiAgent Decision Process (MADP) Toolbox 0.3," 2014.

[25] A. R. Cassandra, "Exact and Approximate Algorithms for Partially Observable Markov Decision Processes," Ph.D. dissertation, 1998.

[26] L. Busoniu, R. Babuška, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, vol. 38, no. 2, pp. 156–172, 2008.