Risk-Sensitive Reinforcement Learning for URLLC Traffic in Wireless Networks

Nesrine Ben Khalifa, Mohamad Assaad, Merouane Debbah

Risk-Sensitive Reinforcement Learning for URLLC Traffic in Wireless Networks

Nesrine Ben Khalifa$^\diamond$, Mohamad Assaad$^\diamond$, Mérouane Debbah*

$^\diamond$TCL chair on 5G, Laboratoire des Signaux et Systèmes (L2S), Centrale Supélec, France

*Huawei Technologies, Boulogne-Billancourt, France

Abstract—In this paper, we study the problem of dynamic channel allocation for URLLC traffic in a multi-user multi-channel wireless network where urgent packets have to be successfully received in a timely manner. We formulate the problem as a *finite-horizon* Markov Decision Process with a stochastic constraint related to the QoS requirement, defined as the *packet loss rate* for each user. We propose a novel weighted formulation that takes into account both the total expected reward (number of successfully decoded packets) and the *risk* which we define as the QoS requirement violation. First, we use the value iteration algorithm to find the optimal policy, which assumes a perfect knowledge of the controller of all the parameters, namely the channel statistics. We then propose a Q-learning algorithm where the controller learns the optimal policy without having knowledge of neither the CSI nor the channel statistics. We illustrate the performance of our algorithms with numerical studies.

Index Terms—URLLC, risk-sensitivity, resource allocation, constrained MDP, reinforcement learning

I. INTRODUCTION

In the fifth generation (5G) wireless networks, there are new service categories with heterogeneous and challenging requirements, among them the Ultra Reliable Low Latency (URLLC) traffic [1], designed for delay and reliability sensitive applications like real-time remote control, autonomous driving, and mission-critical traffic. In URLLC traffic, the End-to-End (E2E) latency defined by 3GPP is lower than 1 ms along with a reliability requirement of $1 - 10^{-5}$ to $1 - 10^{-9}$ [1], [2].

A plausible solution to address the latency requirement issue is to make transmissions without Channel State Information (CSI) knowledge at the transmitter side. To increase reliability, exploiting frequency diversity is beneficial, and occurs by making parallel transmissions of the same packet over different subcarriers in an Orthogonal Frequency Division Multiplexing (OFDM) system where each subcarrier experiences different channel characteristics.

However, this solution is costly in terms of system capacity. Therefore, the number of parallel transmissions should not be fixed in advance but should rather be variable and depending on many parameters such as the position of a user in the cell, or the statistics about his packet losses over the previous time slots. For example, if a user experienced a high number of packet losses in the previous time slots, it should be allocated a high number of subchannels to increase his success probability, whereas a user with a low number of dropped packets may be

assigned a low number of subcarriers. Hence, it is crucial to design efficient dynamic schemes able to adapt the number of parallel transmissions for each user to his experienced QoS.

In this work, we study the problem of dynamic channel allocation for URLLC traffic in a multi-user multi-channel wireless network under QoS constraints. A channel here refers to a frequency band or a subcarrier in an OFDM system, and the QoS is related to the *packet loss rate* for each user, defined as the average number of dropped packets. Besides, we introduce the notion of *risk* related to the violation of the QoS requirements; more precisely, a risk occurs or equivalently, a risk state is reached when the QoS requirement is violated for a user. Furthermore, we consider that the transmitter does not have neither the CSI nor the channel statistics at the transmission moment. In fact, due to the urgency of URLLC packets mentioned previously, there is not enough time for the BS to make channel estimation and probing techniques like in conventional wireless communications.

A. Related Work

The issue of deadline-constrained traffic scheduling has been investigated by several works including [3]–[6]. For example, in [6], the authors study the problem of dynamic channel allocation in a single user multi-channel system with service costs and deadline-constrained traffic. They propose online algorithms to enable the controller to learn the optimal policy based on Thompson sampling for multi-armed bandit problems. The MDP framework and reinforcement learning approaches for downlink packet scheduling are considered in [3], [5], [7]–[10]. In [3], the authors propose an MDP for deadline-constrained packet scheduling problem and use dynamic programming to find the optimal scheduling policies. The authors do not consider QoS constraints in the scheduling problem.

Most risk-sensitive approaches consist in analyzing higher order statistics than the average metric such as the variance of the reward [1], [11]–[13]. For instance, a risk-sensitive reinforcement learning is studied in [14] in millimeter-wave communications to optimize both the bandwidth and transmit power. The authors consider a utility (data rate) that incorporates both the average and the variance to capture the tail distribution of the rate, useful for the reliability requirement of URLLC traffic. The authors do not exploit frequency diversity.

In this work, we consider an alternative approach to the risk which consists in minimizing the risk state visitation probabil-

ity. In fact, due to the stochastic nature of the problem (time-varying channel and random arrival traffic in our context), giving a low reward to an undesirable or a risk-state may be insufficient to minimize the probability of visiting such state [15]. Therefore, in addition to the maximization of the total expected reward, we propose to consider a second criterion which consists in minimizing the probability of visiting risk states where a risk state here is related to the violation of QoS requirements.

B. Addressed Issues and Contribution

In this work, we address the following issues:

- We formulate the dynamic channel allocation problem for URLLC traffic as a *finite-horizon* MDP wherein the state represents the QoS of the users, that is, the average number of dropped packets or *packet loss rate* of the users. The decision variable is the number of channels to assign to each user. We define a *risk state* as any state where the QoS requirement is violated for at least one user. Besides, we define a stochastic constraint related to the risk state visitation probability.
- Assuming the channel statistics are known to the controller, we use the finite-horizon value iteration algorithm to find the optimal policy to the weighted formulation of the problem, which takes into account both the total expected reward over the planning horizon and the *risk* criterion (QoS requirement violation probability).
- When the channel statistics are unknown to the controller, we propose a reinforcement learning algorithm (Q-learning) for the weighted formulation of the problem, which enables the controller to learn the optimal policy. We illustrate the performance of our algorithms with numerical studies.

C. Paper Structure

In Section II, we present the system model for the multi-user multi-channel wireless network with URLLC packets and time-varying channels along with the QoS definition. In Section III, we introduce the constrained MDP formulation with all its components. In Section IV, we present both the finite-horizon value iteration algorithm and the reinforcement learning algorithm. Section V is devoted to numerical results. Finally, we conclude the paper in Section VI.

II. SYSTEM MODEL

We consider a multi-user multi-channel wireless network where URLLC packets have to be transmitted over time-varying and fading channels. Due to the strict latency requirement of URLLC packets in 5G networks mentioned previously, there is not enough time for the BS to estimate the channel, and the packets are then immediately transmitted in the absence of CSI at the transmitter side. When a packet is successfully decoded, the receiver sends an acknowledgment feedback, which is assumed to be instantaneous and error-free. We consider a centralized controller which dynamically
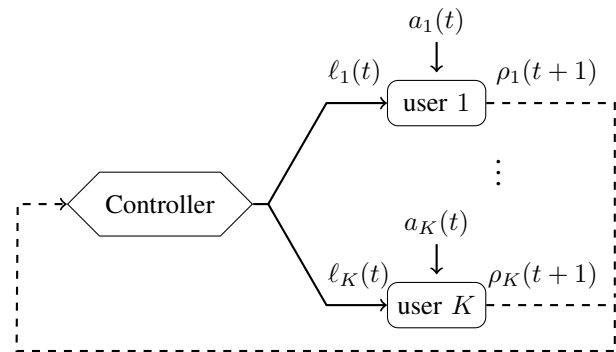
Fig. 1: Dynamic allocation of channels $(\ell_1, .., \ell_K)$ to the users based on their QoS $(\rho_1, .., \rho_K)$.

distributes the channels to the users based on their QoS (see Fig. 1).

Furthermore, we make the following assumptions:

Packet arrival process: the packet arrival process is considered as an independent and identically distributed (i.i.d.) random process over a finite set $\mathcal{I} = \{0, 1, .., A_{max}\}$, where A_{max} is a positive constant, and is identical for all the users. Let α_a denote the probability that $a \in \mathcal{I}$ packets arrive for a given user at the beginning of a time slot.

Deadline-constrained traffic: regarding the strict URLLC latency requirement specified by 3GPP (lower than 1 ms), each packet has a lifetime of one time slot and can either be served or dropped; if there are available channels, the packet will be transmitted, otherwise, it will be dropped because after one time slot it becomes outdated and useless. Furthermore, one packet is transmitted per channel.

Channel model: we consider i.i.d. Bernoulli channels with a mean $\mu \in [0, 1]$. In millimeter-wave communications, the links are characterized by their intermittence and high sensitivity, and this channel model reflects the existence of a light-of-sight (LOS) channel state [6], [16]. To increase reliability, a user can be assigned more channels than the number of waiting packets (depending on his experienced QoS). Some packets are then simultaneously sent over multiple parallel channels.

Channel split: for each user, all the packets are equally important: when the number of available channels is larger than that of waiting packets, we assume that some packets are picked uniformly at random to be replicated. A packet is obviously more likely to be successfully received when sent over many channels simultaneously. However, assigning more channels to a user will affect the QoS experienced by the other users. Note that channel split across the packets (which occurs in the same manner for all the users) should not be confused with the channel split across the users (which takes into account the QoS perceived by the users).

For user k, the distribution of available channels ℓ_k over the waiting packets a_k occurs as follows: each packet is transmitted over $(\ell_k \wedge a_k)$ channels and may be furthermore replicated once with a probability $(\frac{\ell_k \vee a_k}{a_k})$, where the symbol $\ell_k \wedge a_k$ denotes the larger integer m such that $ma_k \leqslant \ell_k$, and

$\ell_k \vee a_k$ denotes the remaining integer of the division of ℓ_k by a_k.

The probability that a packet is successfully received given that there are a_k waiting packets at the transmitter and ℓ_k assigned channels can then be expressed by

$$\nu_k(a_k, \ell_k) = \left(1 - \frac{\ell_k \vee a_k}{a_k}\right)\left(1 - (1-\mu)^{(\ell_k \wedge a_k)}\right)$$
$$+ \left(\frac{\ell_k \vee a_k}{a_k}\right)\left(1 - (1-\mu)^{1+(\ell_k \wedge a_k)}\right). \quad (1)$$

The expected number of successfully received packets for user k is then given by

$$\mathbb{E}\left[N_k(\ell_k)\right] = \sum_{a_k \in \mathcal{I}} a_k \alpha_{a_k} \nu_k(a_k, \ell_k). \quad (2)$$

QoS criterion: for each user k, we define the packet loss rate at time slot t, $\rho_k(t)$, as follows

$$\rho_k(t) = \frac{1}{t}\sum_{i=0}^{t-1} \frac{n_k(i)}{a_k(i)}, \quad t \geqslant 1, \quad (3)$$

where $n_k(t)$ denotes the number of lost packets for user k at time slot t. Note that $\rho_k \in [0,1]$ ($n_k(t) \leqslant a_k(t)$). A packet is lost when either of the two following events occurs:

(i) it is not transmitted because of insufficient available channels,

(ii) is transmitted but ACK feedback is not received.

The parameter ρ_k reflects the QoS perceived by user k: higher values of ρ_k mean a higher number of lost packets and poor QoS whereas lower values of ρ_k mean good QoS. To ensure good QoS for the users, the resource allocation scheme should take account of their experienced QoS and keep this parameter values for all users within an acceptable range.

Finally, the decision variable is the number of channels associated to each user k at each time slot, denoted by ℓ_k, which satisfies

$$\sum_{k=1}^{K} \ell_k(t) = L, \quad (4)$$

where L denotes the number of available channels.

III. Constrained MDP Framework

The stochastic nature of the wireless channel incites us to consider an MDP framework to solve the decision problem. In this section, we first introduce the constrained MDP formulation along with its components. We then derive the optimality equations.

A. Model Formulation

We define the following *finite-horizon* MDP

- *State Space*: is the finite set $\mathcal{T} \times \mathcal{S}$ where $\mathcal{T} = \{0, .., T\}$, $\mathcal{S} = \{\rho_1 \times .. \times \rho_K\}$, ρ_k for $k = 1, .., K$ is defined in (3), and the symbol $\times$ stands for the Cartesian product.
- *Action Space*: is the finite set $\mathcal{L} = \{(\ell_1, .., \ell_K)$ satisfying (4)$\}$, where ℓ_k denotes the number of channels assigned to user k.

- *Reward*: we define the reward r at time slot t, when the controller chooses action $\ell \in \mathcal{L}$ in state s_t, as the expected total number of successfully received packets over all the users, that is,

$$r(s_t, \ell) = \mathbb{E}\left[\sum_{k=1}^{K} N_k(\ell_k)\right]. \quad (5)$$

Note that the reward depends only on the number of channels allocated for each user (the action), and not on the current state s_t. Besides, the reward is a non-linear function of the action.

- *Transition Probabilities:*
First, we define the probability that n packets are lost for user k as a function of the number of waiting packets a_k and the number of assigned channels ℓ_k at a given time slot as follows

$$\sigma_k(n, a_k, \ell_k) = \binom{a_k}{n}\left(1 - \nu_k(a_k, \ell_k)\right)^n \nu_k(a_k, \ell_k)^{a_k - n},$$

where $n \leqslant a_k$ and $\binom{a_k}{n}$ denotes the binomial coefficient. The state transition probability for user k is given by

$$p(\rho_k' \,|\, \rho_k(t), \ell_k) = \alpha_{a_k} \sigma_k(n, a_k, \ell_k), \quad (6)$$

where

$$\rho_k' = \frac{t}{t+1}\rho_k + \frac{1}{t+1}\frac{n}{a_k}. \quad (7)$$

Finally, let $s_{t+1} = \rho_1' \times .. \times \rho_K'$ and $s_t = \rho_1 \times .. \times \rho_K$, the transition probability from state s_t to state s_{t+1} given when action l is taken, is then given by

$$p(s_{t+1} \,|\, s_t, \ell) = \prod_{k=1}^{K} p(\rho_k' \,|\, \rho_k(t), \ell_k). \quad (8)$$

Regarding the strict requirements of URLLC packets described earlier, we introduce in the following the notion of a *risk-state*.

Definition 1. *We define a risk state any state where $\rho_k > \rho_{max}$ for any $k \in \{1, .., K\}$ with $\rho_{max} > 0$ is constant fixed by the controller. The set of risk states Φ is then,*

$\Phi = \{\rho_1 \times .. \times \rho_K$ *where there* $\exists$ *k such that* $\rho_k > \rho_{max}\}$.

Besides, a risk-state is an absorbing state, that is, the process ends when it reaches a risk state [15].

A deterministic policy π assigns at each time step and for each state an action. Our goal is to find an optimal deterministic policy π^* which maximizes the total expected reward $\mathcal{V}_T^\pi(s)$ given by

$$\mathcal{V}_T^\pi(s) = \mathbb{E}^\pi\left[\sum_{t=0}^{T} r(s_t, \pi(s_t)) \,|\, s_0 = s\right], \quad (9)$$

with the reward r is defined in (5), while satisfying the QoS constraint given by

$$\eta^\pi(s) < w, \quad (10)$$

where $\eta^\pi(s)$ denotes the probability of visiting a risk state over the planning horizon, given that the initial state (at time slot

0) is s and policy π is followed, and w is a positive constant. Formally,

$$\eta^{\pi}(s) = \mathbb{P}^{\pi} \left(\exists t \text{ such that } s_t \in \Phi | s_0 = s \right). \tag{11}$$

In order to explicitly characterize $\eta^{\pi}(s)$, we introduce in the following the risk signal $\overline{r}$.

Definition 2. *We define a risk signal $\overline{r}$ as follows*

$$\overline{r}(s_t, \ell_t, s_{t+1}) = \begin{cases} 1 & \text{if } s_{t+1} \in \Phi \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

where s_t and ℓ_t denote the state and action at time slot t, respectively, and s_{t+1} denotes the subsequent state.

Proposition 1. *The probability of visiting a risk-state, $\eta^{\pi}(s)$, is given by*

$$\eta^{\pi}(s) = \overline{\mathcal{V}}_T^{\pi}(s), \tag{13}$$

where we set

$$\overline{\mathcal{V}}_T^{\pi}(s) = \mathbb{E}^{\pi} \left[\sum_{t=0}^{T} \overline{r}(s_t, \pi(s_t), s_{t+1}) \mid s_0 = s \right]. \tag{14}$$

Proof. The random sequence $\overline{r}(t = 0)$, $\overline{r}(t = 1)$,.., $\overline{r}(t = T)$ may contain 1 if a risk state is visited, otherwise all its components are equal to zero (recall that a risk state is an absorbing state). Therefore, $\sum_{t=0}^{T} \overline{r}(t)$ is a Bernoulli random variable with a mean equal to the probability of reaching a risk state, that is, relation (13) holds. $\qquad\square$

B. Optimality Equations

By virtue of Proposition 1, we associate a state value function $\overline{\mathcal{V}}_T^{\pi}$ to the probability of visiting a risk state. Now, we define a new weighted value function $\mathcal{V}_{\xi,T}^{\pi}$, which incorporates both the reward and the risk, as follows

$$\mathcal{V}_{\xi,T}^{\pi}(s) = \xi \mathcal{V}_T^{\pi}(s) - \overline{\mathcal{V}}_T^{\pi}(s), \tag{15}$$

where $\xi > 0$ is the weighting parameter, determined by the risk level the controller is willing to tolerate. The function $\mathcal{V}_{\xi,T}^{\pi}$ can be seen as a standard value function associated to the reward $\xi r - \overline{r}$. The case $\xi = 0$ corresponds to a *minimum-risk* policy whereas the case $\xi \to \infty$ corresponds to a *maximum-value* policy.

Let Π denote the set of deterministic policies, and define

$$\mathcal{V}_T^*(s) = \max_{\pi \in \Pi} \mathcal{V}_T^{\pi}(s), \quad \overline{\mathcal{V}}_T^*(s) = \min_{\pi \in \Pi} \overline{\mathcal{V}}_T^{\pi}(s),$$

$$\mathcal{V}_{\xi,T}^*(s) = \max_{\pi \in \Pi} \mathcal{V}_{\xi,T}^{\pi}(s).$$

Besides, we define u_t^{π}, $\overline{u}_t^{\pi}$, and $u_{\xi,t}^{\pi}$ for $0 \leqslant t \leqslant T$ respectively by

$$u_t^{\pi}(s) = \mathbb{E}^{\pi} \left[\sum_{i=t}^{T} r(s_i, \pi(s_i)) | s_t = s \right], \tag{16}$$

$$\overline{u}_t^{\pi}(s) = \mathbb{E}^{\pi} \left[\sum_{i=t}^{T} \overline{r}(s_i, \pi(s_i), s_{i+1}) | s_t = s \right], \tag{17}$$

$$u_{\xi,t}^{\pi}(s) = \xi u_t^{\pi}(s) - \overline{u}_t^{\pi}(s). \tag{18}$$

Note that $\mathcal{V}_T^{\pi}$ incorporates the total expected reward over the entire planning horizon whereas u_t incorporates the rewards from decision epoch t to the end of the planning horizon only. Besides, $\overline{u}_t(s)$ is the probability of visiting a risk state given that at time t the system is in state $s \in \{\mathcal{S}/\Phi\}$, and is thus a measure of the risk.

The optimality equations are given by (the proof is similar to that in [17], chap. 4 and skipped here for brevity)

$$u_t^*(s) = \max_{\ell \in \mathcal{L}} \left\{ r(s_t, \ell) + \sum_{j \in \mathcal{S}} p(j|s_t, \ell) u_{t+1}^*(j) \right\} \tag{19}$$

$$\overline{u}_t^*(s) = \min_{\ell \in \mathcal{L}} \left\{ \sum_{j \in \mathcal{S}} p(j|s_t, \ell) \left(\overline{r}(s_t, \ell, j) + \overline{u}_{t+1}^*(j) \right) \right\} \tag{20}$$

$$u_{\xi,t}^*(s) = \max_{\ell \in \mathcal{L}} \left\{ \sum_{j \in \mathcal{S}} p(j|s_t, \ell) \left(\xi r(s_t, \ell) - \overline{r}(s_t, \ell, j) \right. \right.$$
$$\left. \left. + u_{\xi,t+1}^*(j) \right) \right\}, \tag{21}$$

for $t = 0, .., T - 1$. For the boundary conditions, that is at time slot T, $u_T^*(s)$, $\overline{u}_T^*(s)$, and $u_{\xi,T}^*(s)$ are set to zero for each $s \in \mathcal{S}$.

In a non-risk state, the reward r is given in (5) and the risk signal is equal to zero whereas in a risk state the reward r is set to zero and the risk signal $\overline{r}$ is set to one.

IV. ALGORITHM DESIGN

In this section, we present two algorithms: (i) finite-horizon value iteration algorithm which assumes that all the model parameters are known to the controller, namely the channel statistics (model-based algorithm), and (ii) reinforcement learning algorithm which does not require the controller's knowledge of channel statistics (model-free algorithm).

A. Value Iteration Algorithm

In order to find a policy that maximizes the weighted value function defined in (15), we use the value iteration algorithm [17]. In this algorithm, we proceed backwards: we start by determining the optimal action at time slot T for each state, and successively consider the previous stages, until reaching time slot 0 (see Algorithm 1).

Algorithm 1 Finite-Horizon Value Iteration Algorithm

1: Initialization: **for each s**
2: $u_T^*(s) \leftarrow 0$, $\overline{u}_T^*(s) \leftarrow 0$, $u_{\xi,T}^*(s) \leftarrow 0$
3: **Endfor**
4: $t \leftarrow T - 1$
5: **while** $t \geqslant 0$
6: **For each s**
7: **update** $u_t^*(s)$, $\overline{u}_t^*(s)$, and $u_{\xi,t}^*(s)$ according to (19), (20), and (21), respectively
8: **EndFor**
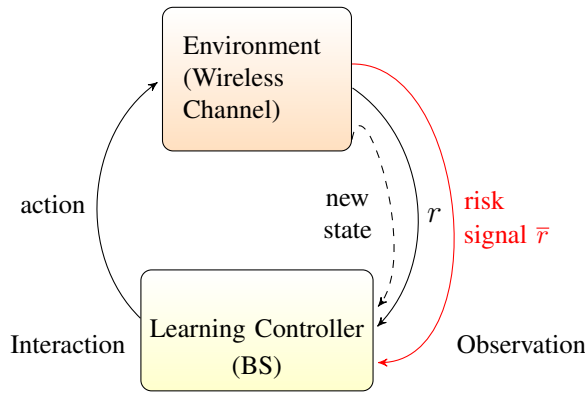9: $t \leftarrow t - 1$
10: **EndWhile**

Fig. 2: Reinforcement Learning Model.

B. Risk-Sensitive Reinforcement Learning Algorithm

During the learning phase, the controller gets estimates of the value of each state-action pair. It updates its estimates through the interaction with the environment where at each iteration it performs an action and then observes the reward r, the risk signal $\overline{r}$, and the next state (see Fig. 2).

The learning controller chooses an action at each learning step following the ε-greedy policy, that is, it selects an action that maximizes its current estimate with probability $1 - \varepsilon$, or a random action with probability ε. The parameter ε captures the exploration-and-exploitation trade-off: when $\varepsilon \to 0$, the controller tends to choose an action that maximizes its current state's estimated value; whereas when $\varepsilon \to 1$, the controller tends to choose randomly an action and to favor the exploration for optimality.

The state-action value function is given by [18], [19]

$$Q^\pi(s_t, \ell) = r(s_t, \ell) + \sum_{j \in \mathcal{S}} p(j|s_t, \ell)\, u_{t+1}^\pi(j),$$

where the first term denotes the immediate reward, that is the number of successfully received packets over all the users, when the action l is performed in state s_t; and the second term denotes the expected reward when the policy π is followed in the subsequent decision stages. Similarly to the state-action value function associated to the reward, we define the state-action value function associated to the risk $\overline{Q}^\pi$ as

$$\overline{Q}^\pi(s_t, \ell) = \sum_{j \in \mathcal{S}} p(j|s_t, \ell)\left(\overline{r}(s_t, \ell, j) + \overline{u}_{t+1}^\pi(j)\right).$$

Note that the introduction of the signal risk $\overline{r}$ enabled us to define a state-action value function, $\overline{Q}$ to the risk.

Besides, the state-action value function associated to the weighted formulation, Q_ξ^π, is given by

$$Q_\xi^\pi(s_t, \ell) = \xi Q^\pi(s_t, \ell) - \overline{Q}^\pi(s_t, \ell).$$

Finally, the Q-function updates at the learning step n (which should not be confused with the decision epoch t) are given by [18]

$$Q^{(n+1)}(s_t, \ell) \leftarrow [1 - \alpha_n(s_t, \ell)]Q^{(n)}(s_t, \ell) + \\ \alpha_n(s_t, \ell)[r + \max_{\ell \in \mathcal{L}} \{Q^{(n)}(s_{t+1}, \ell)\}], \quad (22)$$

$$\overline{Q}^{(n+1)}(s_t, \ell) \leftarrow [1 - \alpha_n(s_t, \ell)]\overline{Q}^{(n)}(s_t, \ell) + \\ \alpha_n(s_t, \ell)[\overline{r} + \min_{\ell \in \mathcal{L}} \{\overline{Q}^{(n)}(s_{t+1}, \ell)\}], \quad (23)$$

and,

$$Q_\xi^{(n+1)}(s_t, \ell) \leftarrow [1 - \alpha_n(s_t, \ell)]Q_\xi^{(n)}(s_t, \ell) + \\ \alpha_n(s_t, \ell)[\xi r - \overline{r} + \max_{\ell \in \mathcal{L}} \{Q_\xi^{(n)}(s_{t+1}, \ell)\}], \quad (24)$$

where $\alpha_n(s_t, \ell)$ denotes the learning rate parameter at step n when the state s_t and action ℓ are visited.

The learning algorithm converges to the optimal state-action value function when each state-action pair is performed infinitely often and when the learning rate parameter satisfies for each (s_t, ℓ) pair (the proof is given in [18], [20] and skipped here for brevity),

$$\sum_{n=1}^\infty \alpha_n(s_t, \ell) = \infty, \text{ (a)} \quad \text{and} \quad \sum_{n=1}^\infty \alpha_n^2(s_t, \ell) < \infty \text{ (b)}. \quad (25)$$

In this case, the Q-functions are related to the value functions as follows

$$\max_{\ell \in \mathcal{L}} \{Q(s_t, \ell)\} = u_t^*(s_t), \quad \min_{\ell \in \mathcal{L}} \{\overline{Q}(s_t, \ell)\} = \overline{u}_t^*(s_t),$$

$$\max_{\ell \in \mathcal{L}} \{Q_\xi(s_t, \ell)\} = u_{\xi,t}^*(s_t).$$

When a risk state is reached during the learning phase, the system is restarted according to the uniform distribution to a non-risk state. In addition, when $t \geq T$, we consider that an artificial absorbing state is reached and we reinitialize t (see Algorithm 2).

Algorithm 2 Q-learning Algorithm

1: Initialization $t \leftarrow 0$, $s_0 \leftarrow s$, $n \leftarrow 1$,
2:   **for each** $\ell \in \mathcal{L}$
3:   $Q(s_0, \ell) \leftarrow 0$, $\overline{Q}(s_0, \ell) \leftarrow 0$, $Q_\xi(s_0, \ell) \leftarrow 0$
4:   **End for**
5: **Repeat**
6: observe current state s_t
7: select and perform action ℓ in state s_t
8: observe the new state s_{t+1}, reward r and the risk $\overline{r}$
9: **update the Q-functions** $Q(s_t, l)$, $\overline{Q}(s_t, l)$, $Q_\xi(s_t, \ell)$
   according to (22), (23), (24) respectively
10: $t \leftarrow t + 1$
11: $n \leftarrow n + 1$
12: **update** α_n
13: **if** $t = T$, **then** $t \leftarrow 0$ *artificial absorbing state reached*
14: **if** $s_t \in \Phi$, **then** $s_t \sim \text{Unif}\{\mathcal{S}/\Phi\}$ *absorbing state reached*
15: **until** convergence

V. PERFORMANCE EVALUATION

In this section, we present the numerical results obtained with the value iteration and the learning algorithms in a variety of scenarios. We consider the setting of two users along with a number of channels $L = 5$. For the arrival traffic, we consider the following truncated Poisson distribution

$$\text{Prob}(a = m) = \begin{cases} \frac{\lambda^m/m!}{\sum_{i=0}^{A_{max}} \lambda^i/i!} & \text{if } m \leqslant A_{max} \\ \text{zero} & \text{otherwise,} \end{cases} \quad (26)$$

where $\lambda = 3$ and $A_{max} = 6$. The mean of the Bernoulli channel μ and the value of the parameter ρ_{max} throughout this section are fixed to 0.6 and 0.55 respectively.

A. Minimum-risk vs maximum-value policy

First, we compare the performance of the minimum-risk policy (obtained when $\xi = 0$), maximum-value policy (obtained when $\xi \to \infty$), weighted policy (when $\xi > 0$), and the fixed policy which consists is assigning the same number of channels for each user at each time slot ($\ell_1 = 2$ and $\ell_2 = 3$).

We depict in Fig. 3-*top* the reward $u_t(s)$ given in (19) as a function of time when $s = 0.3 \times 0$ and different policies are followed. We observe that the maximum-value policy clearly outperforms the fixed and the minimum-risk policy. In Fig. 3-*bottom* showing $\overline{u}_t(s)$ given in (20), we observe that the probability of visiting a risk-state when the fixed policy is followed is much higher than that obtained when the minimum-risk policy $\overline{\pi}^*$ is performed. For example, at the time step $t = 5$, $\overline{u}_t(s)$ is equal to 0.42 when the policy π_f is performed whereas this value reduces to 0.02 when the policy $\overline{\pi}^*$ is followed. In fact, the fixed policy does not take account of the experienced QoS of the users, and therefore, it is the policy which results in the highest risk-state visitation probability. Besides, this probability decreases over time for all the policies. In fact, as time goes on, the probability of entering a risk-state over the remaining time steps decreases.

The reward $u_t(s)$ increases for the lower values of t until reaching a maximum value and then it decreases, for all the policies. In fact, for the lower values of t, the probability of visiting a risk-state is high, and this affects the expected value of the reward (recall that in the risk state, the reward is equal to zero). As time goes on, this probability decreases, and thus the expected reward increases. However, at the further time steps, the number of remaining decision stages is low and hence the expected reward (total number of successfully received packets over the remaining time slots) decreases.

B. Learning

In the learning algorithm, we simulate the wireless channel with a Bernoulli random variable with a number of trials equal to the number of channels associated to each packet for each user. For the learning rate parameter α_n, we considered the following expression [21]:

$$\alpha_n = \frac{1}{(1 + n(s_t, \ell))^\gamma}, \quad (27)$$
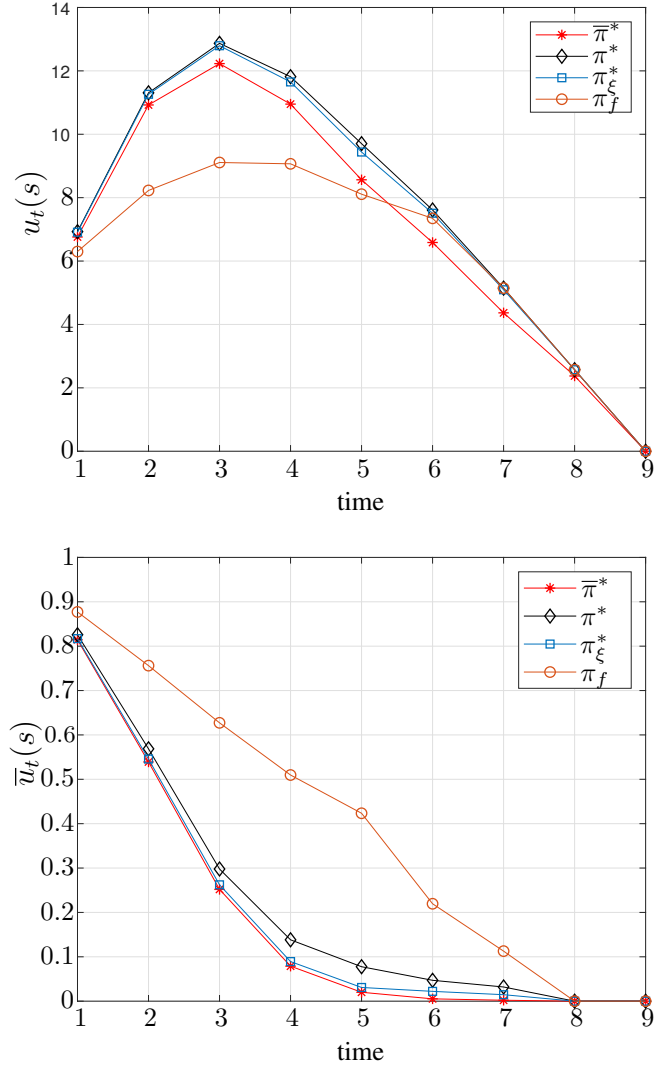
Fig. 3: Performance of the minimum-risk policy $\overline{\pi}^*$, the maximum-value policy π^*, the weighted-policy π_ξ^* with $\xi = 0.1$, and the fixed policy π_f. On the *top*, $u_t(s)$, on the *bottom*, $\overline{u}_t(s)$ where $s = 0.3 \times 0$ and $T = 9$.

where $n(s_t, \ell)$ denotes the number of times the state-action pair (s_t, ℓ) was visited until iteration n, and γ is a positive parameter $\in [0.5, 1]$ [21].

We depict in Fig. 4 the optimal (minimum-risk) policy (number of channels to assign to user 1, $\ell_1 \in [0, .., 5]$) computed by the learning algorithm, as a function of time steps (decision epochs) and ρ_1, when ρ_2 is fixed to 0. The figure shows a monotony property: the number of channels to assign to user 1 increases with time and with ρ_1. In fact, as the QoS of user 1 degrades (ρ_1 increases), more channels are assigned to it to compensate for this degradation; and as time goes on, this policy is more sensitive to this degradation as more channels are assigned for the same values of ρ_1, but at further time steps.
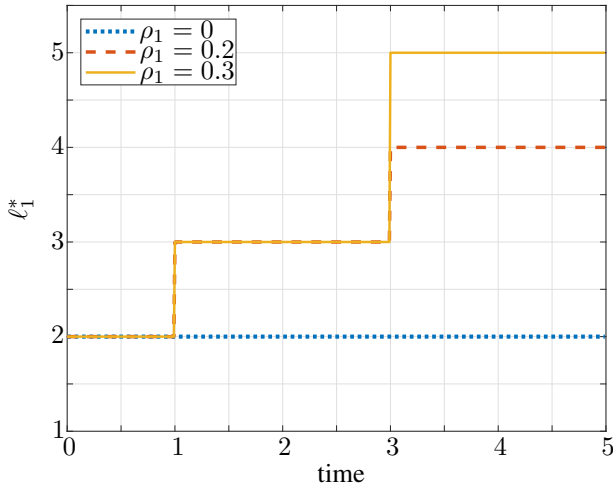
Fig. 4: Optimal policy ℓ_1^* as a function of time steps and ρ_1 with $\rho_2 = 0$ and $T = 5$.

VI. Conclusion

In this work, we studied the problem of dynamic channel allocation for URLLC traffic in a multi-user multi-channel wireless network within a novel framework. Due to the stochastic nature of the problem related to time-varying, fading channels and random arrival traffic, we considered a finite-horizon MDP framework. We determined explicitly the probability of visiting a risk state and we wrote it as a cumulative return (risk signal). We then introduced a weighted global value function which incorporates two criteria: reward and risk. By virtue of the value iteration algorithm, we determined the optimal policy. Furthermore, we used a Q-learning algorithm to enable the controller to learn the optimal policy in the absence of channel statistics. Besides, we examined the impact of the learning rate parameter on the accuracy of the learning algorithm.

In the future work, we would like to take account of spatial diversity in the dynamic allocation scheme where both the BS and the user terminals can be equipped with multiple antennas to enhance the system performance.

References

[1] M. Bennis, M. Debbah, and H. V. Poor, "Ultra-reliable and low-latency wireless communication: tail, risk and scale," *Technical Report, ArXiv*, may 2018.

[2] H. Ji, S. Park, J. Yeo, Y. Kim, J. Lee, and B. Shim, "Ultra-reliable and low-latency communications in 5G downlink: Physical layer aspects," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 124–130, June 2018.

[3] A. Dua and N. Bambos, "Downlink wireless packet scheduling with deadlines," *IEEE Transactions on Mobile Computing*, vol. 6, no. 12, pp. 1410–1425, Dec 2007.

[4] N. Nomikos, N. Pappas, T. Charalambous, and Y. Pignolet, "Deadline-constrained bursty traffic in random access wireless networks," in *SPAWC*, May 2018.

[5] A. Destounis, G. S. Paschos, J. Arnau, and M. Kountouris, "Scheduling URLLC users with reliable latency guarantees," in *Proc. of the 16th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2018, pp. 1–8.

[6] S. Cayci and A. Eryilmaz, "Learning for serving deadline-constrained traffic in multi-channel wireless networks," in *Proc. of the 15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2017, pp. 1–8.

[7] R. Aggarwal, M. Assaad, C. E. Koksal, and P. Schniter, "OFDMA downlink resource allocation via ARQ feedback," in *Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Nov 2009, pp. 1493–1497.

[8] N. U. Hassan and M. Assaad, "Dynamic resource allocation in multi-service OFDMA systems with dynamic queue control," *IEEE Transactions on Communications*, vol. 59, no. 6, pp. 1664–1674, June 2011.

[9] A. Ahmad and M. Assaad, "Optimal resource allocation framework for downlink OFDMA system with channel estimation error," in *Proc. of IEEE Wireless Communication and Networking Conference*, Sydney, April 2010, pp. 1–5.

[10] ——, "Joint resource optimization and relay selection in cooperative cellular networks with imperfect channel knowledge," in *Proc. of IEEE 11th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*.

[11] A. Gosavi, "Finite horizon Markov control with one-step variance penalties," in *Proc. of the 48th Annual Allerton Conference on Communication, Control, and Computing*, Sept 2010, pp. 1355–1359.

[12] A. Kumar, V. Kavitha, and N. Hemachandra, "Finite horizon risk sensitive MDP and linear programming," in *Proc. of the 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 7826–7831.

[13] M. Assaad, A. Ahmad, and H. Tembine, "Risk sensitive resource control approach for delay limited traffic in wireless networks," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, Kathmandu, Dec 2011, pp. 1–5.

[14] T. K. Vu, M. Bennis, M. Debbah, M. Latva-aho, and C. S. Hong, "Ultra-reliable communication in 5G mmWave networks: A risk-sensitive approach," *IEEE Communications Letters*, vol. 22, no. 4, pp. 708–711, April 2018.

[15] P. Geibel and F. Wysotzki, "Risk-sensitive reinforcement learning applied to control under constraints," *J. Artif. Int. Res.*, vol. 24, no. 1, pp. 81–108, jul 2005.

[16] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164–1179, June 2014.

[17] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2005.

[18] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, May 1992.

[19] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, Massachusetts: MIT Press, 1998.

[20] D. P. Bertsekas and J. Tsitsiklis, *Neuro-dynamics Programming*. Cambridge, Massachusetts: MIT Press, 1996.

[21] E. Even-Dar and Y. Mansour, "Learning rates for Q-learning," *J. Mach. Learn. Res.*, vol. 5, pp. 1–25, 2004.