Supplementary Appendix

Talha Bozkus and Urbashi Mitra

I. Proof of Proposition 5

The following expressions are valid for all (s,a) pairs; hence, we drop the (s,a) notation for simplicity.

$$\lim_{t\to\infty}\mathbb{V}[\mathcal{E}_t]=\lim_{t\to\infty}\mathbb{V}\Big[(1-u)\sum_{i=0}^{t-1}u^{t-i-1}\sum_{n=1}^{K}\mathbf{w}_i^{(n)}\mathcal{X}_i^{(n)}\Big]. \tag{1}$$

$$=\lim_{t\to\infty}(1-u)^2\big[\sum_{i=0}^{t-1}u^{2(t-i-1)}\mathbb{V}\big[\sum_{n=1}^{K}\mathbf{w}_i^{(n)}\mathcal{X}_i^{(n)}\big]+2\sum_{i=0}^{t-1}\sum_{j=i+1}^{t-1}u^{2(t-i-1)}u^{2(t-j-1)}Cov\big[\sum_{n=1}^{K}\mathbf{w}_i^{(n)}\mathcal{X}_i^{(n)},\sum_{n=1}^{K}\mathbf{w}_j^{(n)}\mathcal{X}_j^{(n)}\big]\big]. \tag{2}$$

$$\leq\lim_{t\to\infty}(1-u)^2\big[\sum_{i=0}^{t-1}u^{2(t-i-1)}\mathbb{V}\big[\sum_{n=1}^{K}\mathbf{w}_i^{(n)}\mathcal{X}_i^{(n)}\big]+2\sum_{i=0}^{t-1}\sum_{j=i+1}^{t-1}u^{2(t-i-1)}u^{2(t-j-1)}\sqrt{\mathbb{V}\big[\sum_{n=1}^{K}\mathbf{w}_i^{(n)}\mathcal{X}_i^{(n)}\big]\mathbb{V}\big[\sum_{n=1}^{K}\mathbf{w}_j^{(n)}\mathcal{X}_j^{(n)}\big]}\big]. \tag{3}$$

$$\leq\lim_{t\to\infty}(1-u)^2\big[\sum_{i=0}^{t-1}u^{2(t-i-1)}+2\sum_{i=0}^{t-1}\sum_{j=i+1}^{t-1}u^{2(t-i-1)}u^{2(t-j-1)}\big]\mathbb{V}\big[\sum_{n=1}^{K}\mathbf{w}_i^{(n)}\mathcal{X}_i^{(n)}\big]. \tag{4}$$

$$\leq\lim_{t\to\infty}(1-u)^2\big[\sum_{i=0}^{t-1}u^{2(t-i-1)}+2\sum_{i=0}^{t-1}\sum_{j=0}^{t-1}u^{2(t-i-1)}u^{2(t-j-1)}\big]\lambda^2. \tag{5}$$

$$\leq\frac{(1-u)}{(1+u)}\lambda^2+\frac{2\lambda^2}{(1+u)^2}, \tag{6}$$

where (1) follows from (19), (2) follows from the properties of the variance operator, (3) follows from the Cauchy-Schwarz inequality for the variance operator, (4) follows from the fact that variance is independent of time indices from (7), (5) follows from (24), and (6) follows from the infinite geometric sum formula.

II. The details of the 6G wireless networks

A. *MISO network with collision channels*

We consider the first wireless network model in [1]. For the parameters μ, P_{rec}, c_0, σ, d_i, the same values as in [1] are used and kept constant for different network sizes. The network size is formed by changing N, M and R as follows: the interval [0, 5000] is formed by increasing N, M comparably with $R=1$, the interval [5000, 10000] is formed similarly with $R=2$, the interval [10000, 15000] is formed similarly with $R=3$, the interval [15000, 20000] is formed similarly with $R=4$, and the interval [20000, 40000] is formed similarly with $R=5$ The values N, M, R are chosen from the following: $N\in\{2,3,...,30\}$, $M\in\{2,3,...,10\}$, $R\in\{2,3,4,5\}$. Fig.1a illustrates the same wireless network model with four transmitters.

B. *MISO energy harvesting network with multiple relays*

We consider the second wireless network model in [1]. The number of transmitters is chosen from $\{2,3,4,5\}$, and the number of relays is chosen from $\{1,2,3\}$. The input parameters $x_i=1$ for $i\in\{2,3,4,5\}$. The probabilities $p_i=0.2$ and $q_i=0.8$ for $i\in\{2,3,4,5\}$. The weight parameters in the cost function $\alpha_1=\frac{1}{9}$, $\alpha_2=\frac{1}{6}$ and $\alpha_3=\frac{1}{3}$, and these are chosen such that different cost components have comparable contributions on the overall cost function. These parameters are also kept constant for different network sizes. The network size is formed as follows: the interval [0, 5000] is formed by increasing N with 3 transmitters and 1 relay, [5000, 10000] is formed by increasing N with 3 transmitters and 2 relays, [10000, 15000] is formed by increasing N with 4 transmitters and 2 relays, [15000, 20000] is formed by increasing N with 4 transmitters and 3 relays, [20000, 30000] is formed by increasing N with 5 transmitters and 2 relays, and [30000, 40000] is formed by increasing N with 5 transmitters and 3 relays. Fig.1b illustrates the same wireless network model with 4 transmitters and 3 relays.

C. MIMO network with collision channels

We generalize the MISO network model in [1] to the multiple receivers case. Assume there are T transmitters and R receivers. All channels between transmitters and receivers are modeled with a standard Gilbert-Elliot model, and the channels between closer TX-RX pairs are more likely to be in a good state than those between farther TX-RX pairs due to shorter distance and less interference. For a network with $T = 3$ transmitters and $R = 4$ receivers, the channels between TX_1 and RX_1, TX_1 and RX_2, TX_2 and RX_2, TX_2 and RX_3, TX_3 and RX_3, TX_3 and RX_4 have the probability of transitioning from good to bad state $p_1 = 0.2$ and the probability of transitioning from bad to good state $1 - p_1 = 0.8$. The other channels have the probability of transitioning from good to bad state $p_2 = 0.4$ and the probability of transitioning from bad to good state $1 - p_2 = 0.6$. The channel transition probability matrix between the transmitter i and receiver j is denoted by $\mathbf{C}_{ij}$. The overall channel transition matrix $\mathbf{C}$ is formed by taking the Kronecker product between $\mathbf{C}_{ij}$ for $i \in \{1, 2, 3\}$ and $j \in \{1, 2, 3, 4\}$.

Each transmitter has a buffer with a size $N-1$ that stores the incoming data packets, which follow a Bernoulli distribution with probability p_3. There are $R + 1$ different actions each transmitter can take: remain silent or transmit j packets to j different receivers for $j = 1, 2, 3, ..., R$. Let $\mathbf{B}_{ijk}$ denote the buffer transition probability of a generic transmitter from state i to state j under action k for $i, j \in \{0, 1, ..., N-1\}$ and $k \in \{0, 1, ..., R\}$. We herein give the equations for $\mathbf{B}_{ijk}$ for $T = 3$ and $R = 4$, and one can construct similar equations for other (T, R) pairs.

$$\begin{aligned}
\mathbf{B}_{i,i,0} &= 1-p, \;\; 0 \le i < N-1, \\
\mathbf{B}_{i,i+1,0} &= p, \;\; 0 \le i < N-1, \\
\mathbf{B}_{N-1,N-1,0} &= 1, \\
\mathbf{B}_{0,0,1} &= 1, \\
\mathbf{B}_{i,i,1} &= p, \;\; 0 < i \le N-1, \\
\mathbf{B}_{i,i-1,1} &= 1-p, \;\; 0 < i \le N-1. \\
\mathbf{B}_{i,0,2} &= 1, \;\; 0 \le i \le 1, \\
\mathbf{B}_{i+1,i,2} &= p, \;\; 1 \le i \le N-2, \\
\mathbf{B}_{i+2,i,2} &= 1-p, \;\; 0 \le i \le N-3. \\
\mathbf{B}_{i,0,3} &= 1, \;\; 0 \le i \le 2, \\
\mathbf{B}_{i+2,i,3} &= p, \;\; 1 \le i \le N-3, \\
\mathbf{B}_{i+3,i,3} &= 1-p, \;\; 0 \le i \le N-4. \\
\mathbf{B}_{i,0,4} &= 1, \;\; 0 \le i \le 3, \\
\mathbf{B}_{i+3,i,4} &= p, \;\; 1 \le i \le N-4, \\
\mathbf{B}_{i+4,i,4} &= 1-p, \;\; 0 \le i \le N-5.
\end{aligned} \tag{7}$$

The overall buffer transition probability tensor is obtained by taking the Kronecker product of $\mathbf{B}$ with itself T times. Let s_t be the state at time t defined as a tuple: $s_t = \{b_t, h_t\}$, where b_t and h_t are the buffer and channel state of the overall system defined for $T = 3$ and $R = 4$ as follows: $b_t = [b_{t,1}, b_{t,2}, b_{t,3}]$, where $b_{t,i} \in \{0, 1, ..., N-1\}$ is the buffer state of the i^{th} transmitter for $i = 1, 2, 3$, and $h_t = [h_{t,11}, h_{t,12}, h_{t,13}, h_{t,21}, h_{t,22}, h_{t,23}, h_{t,31}, h_{t,32}, h_{t,33}, h_{t,41}, h_{t,42}, h_{t,43}]$, where $h_{t,ij} \in \{0, 1, ..., M-1\}$ is the channel state of the channel between the i^{th} transmitter and j^{th} receiver.

Given the state s_t and the set of actions $a_{t,1}, a_{t,2}, a_{t,3}$ for 3 transmitters, the single-stage cost function in time slot t is defined by the following equation:

$$\mathbf{c}(s_t, a_{t,1}, a_{t,2}, a_{t,3}) = \eta_1 \sum_{i=1}^{3} \frac{b_{t,i}}{N-1} + \eta_2 \sum_{i=1}^{3} \sum_{j=1}^{4} f(h_{t,ij}) a_{t,i} + \eta_3 \sum_{i=1}^{3} a_{t,i} + \eta_4 \sigma\{r_{t,1}, r_{t,2}, r_{t,3}, r_{t,4}\}, \tag{8}$$

where $\mathbf{1}(\cdot)$ is the indicator function defined as:

$$f(h_{t,ij}) = \frac{P_{j,rec}}{c_0 d_{ij}^{-\sigma} h_{t,ij}}, \tag{9}$$

which denotes the power threshold at the transmitter for successful transmission as a function of channel gain $h_{t,ij}$ between the i^{th} transmitter and j^{th} receiver at time t, $P_{j,rec}$ is the power threshold at the j^{th} receiver, d_{ij} is the distance between the j^{th} receiver and the i^{th} transmitter, σ is the path loss component, and c_0 is a constant, $r_{t,j}$ is the number of data packets that the j^{th} receiver receivers, and $\sigma\{x_1, ..., x_N\}$ is the sample standard deviation of the given list of numbers.

The first component is the buffer cost, which is proportional to the sum of the occupancy levels of the buffers. The second component is the transmission cost, as a function of the channel conditions, and incurs when the transmitters transmit under unfavorable channel conditions. The third component is the collision cost, which is proportional to the number of data packets being transmitted simultaneously. The fourth component is the receiver load cost. It occurs when the receivers receive a relatively unequal number of data packets. For example, if $r_{t,1} = r_{t,2} = 3$, $r_{t,3} = r_{t,4} = 1$, then there will be an unbalanced load on the receivers ($\sigma\{3, 3, 1, 1\} = 1$) than the case where $r_{t,1} = r_{t,2} = r_{t,3} = r_{t,4} = 2$ ($\sigma\{2, 2, 2, 2\} = 0$). We herein measure how unbalanced the receiver's loads are using the sample standard deviation. Finally, the constants $\eta_1, \eta_2, \eta_3, \eta_4$ determine the weight of each component and are chosen such that different cost components have comparable contributions to the overall cost function. We use the following numerical parameters: $p_3 = 0.8$, $P_{1,rec} = P_{2,rec} = P_{3,rec} = 10^{-9.7}, c_0 = 10^{0.17}, \sigma = 4.7$, and $\eta_1 = 0.3, \eta_2 = 0.2, \eta_3 = 0.3, \eta_4 = 0.2$. For settings with different $R = 4$ and $T = 3$, the following distances are used: $d_{11} = d_{12} = d_{22} = d_{23} = d_{33} = d_{34} = 100$ and $d_{13} = d_{14} = d_{21} = d_{24} = d_{31} = d_{32} = 200$. For the other cases, the closer TX-RX pairs have a distance of 100, while the others have a distance of 200. The network size is formed as follows: the interval [0, 5000] is formed by increasing N with 2 transmitters and 2 receivers, [5000, 10000] is formed by increasing N with 2 transmitters and 3 receivers, [10000, 15000] is formed by increasing N with 3 transmitters and 3 receivers, [15000, 20000] is formed by increasing N with 3 transmitters and 4 receivers, [20000, 30000] is formed by increasing N with 4 transmitters and 4 receivers, and [30000, 40000] is formed by increasing N with 4 transmitters and 5 receivers. Fig.1c illustrates the same wireless network model with 2 transmitters and 3 receivers.

D. MIMO network with mobile channels

We generalize the MIMO network to the case where the transmitters are mobile. Assume the transmitters are constrained within L x L area, where L is an integer multiple of 10. (The boundary coordinates are (0,0), (0,L), (L,0), (L,L)). Each transmitter can be present in the coordinates that are multiples of 10s, such as (60,30) or (0, 60), and they can move multiple units of 10s each time. Assume there are $T = 4$ transmitters and $R = 3$ receivers. The transmitters TX$_1$ and TX$_2$ are faster and can move either 10m or 20m each time or remain fixed, whereas the transmitters TX$_3$ and TX$_4$ are slower and can only move 10m each time or remain fixed. Assume the three receivers are fixed, and their locations are (0,0), (0.6L,0.2L), and (0.2L,0.6L). The initial location of each transmitter is uniformly randomly assigned with the constraint that each has a different initial location and is not allowed to occupy the same location as receivers.

The state is defined by the explicit locations of each transmitter at time t as $s_t = [x_{t,1}, y_{t,1}, x_{t,2}, y_{t,2}, x_{t,3}, y_{t,3}, x_{t,4}, y_{t,4}]$, where $(x_{t,i}, y_{t,i})$ are the location of the i^{th} transmitter for $i = 1, 2, 3, 4$ and $x_{t,i}, y_{t,i} \in \{0, 10, ..., L\}$ at time t. The action space contains the indices of the receivers: $\mathcal{A} = \{1, 2, ..., R\}$. The probability transition tensor of the overall system is constructed by concatenating the probability that the transmitters move from the locations at time t (defined by s_t) to their next locations at time $t + 1$ (defined by s_{t+1}) under the action k, which represents a specific transmitter and receiver association for all different scenarios. Each transmitter has an equal probability of either moving in any direction 10m or 20m (for fast transmitters such as TX$_1$ and TX$_2$) or remaining stationary.

For example, the following equation describes the probability of transitioning from s_t to s_{t+1} for TX$_1$:

$$p(s_{t+1} = [x_{t,1} + 20, y_{t,1}, x_{t,2}, y_{t,2}, x_{t,3} + 10, y_{t,3}, x_{t,4} - 10, y_{t,4}] \Big| s_t = [x_{t,1}, y_{t,1}, x_{t,2}, y_{t,2}, x_{t,3}, y_{t,3}, x_{t,4}, y_{t,4}]) = (\frac{1}{9})^8, \tag{10}$$

whereas the following equation describes the probability from transitioning from s_t to s_{t+1} for TX$_3$:

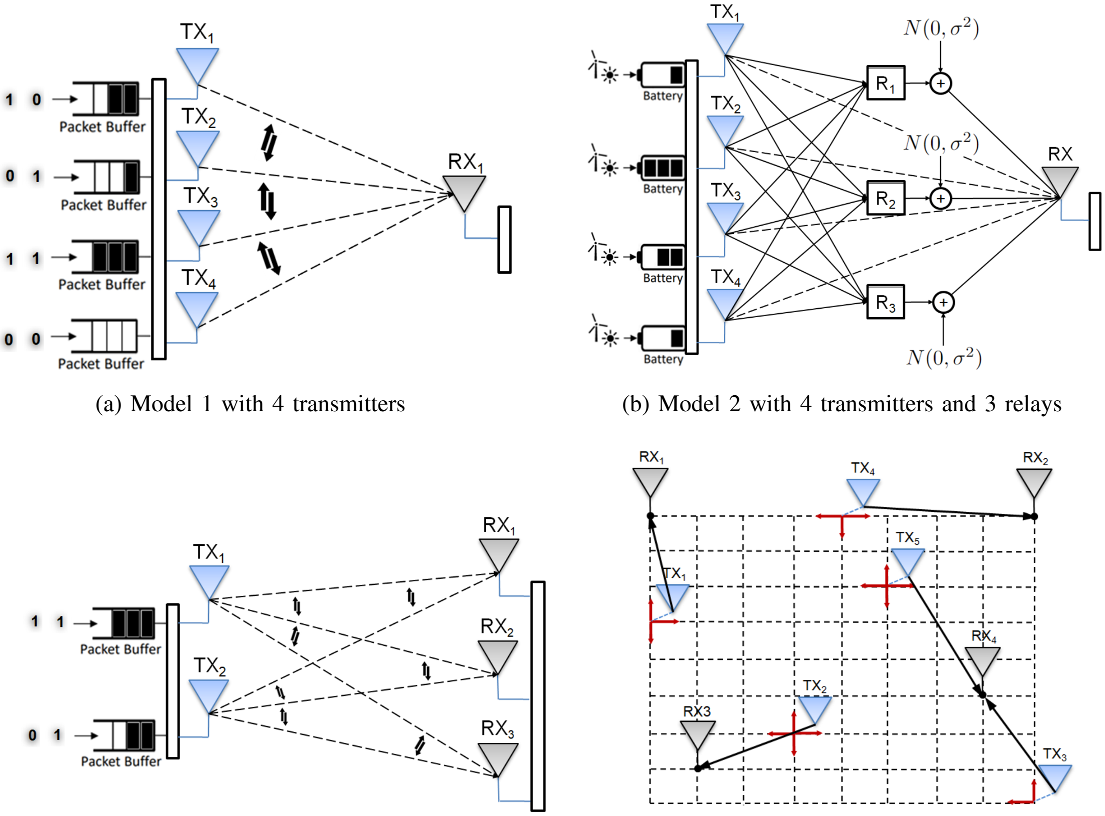
$$p(s_{t+1} = [x_{t,1} + 10, y_{t,1}, x_{t,2} - 10, y_{t,2} - 10, x_{t,3}, y_{t,3}, x_{t,4} - 10, y_{t,4}] \Big| s_t = [x_{t,1}, y_{t,1}, x_{t,2}, y_{t,2}, x_{t,3}, y_{t,3}, x_{t,4}, y_{t,4}]) = (\frac{1}{5})^8, \tag{11}$$

Given the state s_t and the set of actions $a_{t,i}$ for $i = 1, 2, 3, 4$ for $R = 4$ transmitters, the single-stage cost function in time slot t is defined by the following equation:

$$\mathbf{c}(s_t, a_{t,1}, a_{t,2}, a_{t,3}, a_{t,4}) = -\eta_1 \sum_{i=1}^{4} log_2(1 + \frac{\alpha_i}{d_{t,i}^2}) + \eta_2 \sigma\{r_{t,1}, r_{t,2}, r_{t,3}\} + \eta_3 \sum_{i,j:a_{t,i}=a_{t,j}} \frac{P_{i,rec} + P_{j,rec}}{d_{t,i,j}^2} \tag{12}$$

where $d_{t,i}$ is the distance between the TX$_i$ and RX$_{a_{t,i}}$ (i.e. the distance between the i^{th} transmitter and the receiver it is transmitting to at time t), α_i is the path loss coefficient, $r_{t,i}$ is the number of transmitters transmitting to the i^{th} receiver at time t, $P_{i,rec}$ is the received power of the i^{th} transmitter, and $d_{t,i,j}$ is the distance between the i^{th} and j^{th} transmitter at time t, and η_1, η_2 and η_3 are the weighting coefficients. The first component is the negative of total throughput, which is proportional to the squared distance between the pair of each connected transmitter and receiver, discouraging transmitters from connecting to the far receivers. The second component is the receiver load cost, which prevents the overload on receivers by encouraging transmitters to connect to different receivers. The third component is the transmitter interference cost, which is inversely proportional to the distance between transmitters transmitting to the same receiver.

Let $\alpha_1 = \alpha_2 = 1$ and $\alpha_3 = \alpha_4 = 10$. Herein, a larger α reduces the effect of path loss, and we assume that the characteristics of the fast transmitters TX$_1$ and TX$_2$ allow them to handle the path loss easier than TX$_3$ and TX$_4$. Let $\eta_1 = \eta_2 = 0.4$ and $\eta_3 = 0.3$, $L = 40$ and $P_{i,rec} = 10^{-9.7}$ for all i. The given setting illustrates the MIMO system with 4 transmitters and 3 receivers. The numerical settings for the other cases are constructed similarly. The network size is formed as follows: the interval [0, 5000] is formed by increasing N with 2 transmitters and 2 receivers [5000, 10000] is formed by increasing N with 2 transmitters and 3 receivers, [10000, 20000] is formed by increasing N with 3 transmitters and 3 receivers, [20000, 30000] is formed by increasing N with 3 transmitters and 4 receivers, and [30000, 40000] is formed by increasing N with 4 transmitters and 4 receivers with slowly increasing the size of the predefined area L. Fig.1d illustrates the same wireless network model with 5 transmitters, 4 receivers and a larger area.



(a) Model 1 with 4 transmitters

(b) Model 2 with 4 transmitters and 3 relays

(c) Model 3 with 2 transmitters and 3 receivers

(d) Model 4 with 5 transmitters and 4 receivers

Fig. 1: Wireless network models with different settings

III. The details on the parameter optimization

For all algorithms, the same structure for the learning rate (α_t) and epsilon-probability (ϵ_t) are used, and the parameters c_1, c_2, c_3 are optimized. For MaxMin Q-learning, Ensemble Bootstrapped Q-learning and Averaged DQN, the number of estimators is selected via cross-validation for different network sizes as follows: for small networks: $\{2, 3, 4\}$, for modest-sized networks: $\{3, 4, 5, 6\}$, and for large networks: $\{5, 6, 7, 8, 9, 10\}$. For ADQN, the following parameters are selected via cross-validation from the following sets: batch size: $\{16, 32, 64, 128\}$, replay buffer memory: $\{5 \cdot 10^3, 10^4, 2 \cdot 10^4\}$. In addition, the following implementations are used: for small networks: 3-layer fully connected NN with 48 nodes in each layer followed by ReLU, for modest-sized networks: 4-layer fully connected NN with 48 nodes in each layer followed by ReLU, for large networks: 4-layer fully connected NN with 96 nodes in each layer followed by ReLU. For model-3, the pair of the normalized buffer and channel states is used as input (*i.e.* the input size is 2). For model-4, the pair of the normalized

Params	Small networks	Modest-sized networks	Large networks
l	$l \in [5, 10]$	$l \in [10, 15]$	$l \in [15, 30]$
K	$K \in \{2, 3\}$	$K \in \{3, 4, 5\}$	$K \in \{5, 6, 7, 8\}$
α_t	$c_1 \in \{10^2, 5 \cdot 10^2\}$	$c_1 \in \{10^2, 10^3\}$	$c_1 \in \{10^3, 10^4\}$
ϵ_t	$c_2 \in \{0.9, 0.95\}$	$c_2 \in \{0.95, 0.99\}$	$c_2 \in \{0.99, 0.999\}$
	$c_3 \in \{0.01, 0.1\}$	$c_3 \in \{0.01, 0.05\}$	$c_3 \in \{0.005, 0.01\}$

TABLE I: Near-optimal hyper-parameters of Algorithm 1

battery and channel states for different transmitters are concatenated and used as input (*i.e.* the input size is two times the number of transmitters). The output layer has $|\mathcal{A}|$ nodes, followed by a softmax function, where each node represents the probability of a specific action being the optimal one. We use a dropout with a probability of 0.2 after each layer except the final layer. The l_2 regularization is employed with the corresponding weight chosen from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. The Adam optimizer is used to update the weight of neural networks.

IV. Parameter Tuning

We herein discuss how to choose the parameters v, l, K, α_t and ϵ_t for small ($|\mathcal{S}| \leq 10000$), modest ($|\mathcal{S}| \in [10000, 40000]$), and large networks ($|\mathcal{S}| \geq 40000$). Table I summarizes the parameters that yield near-optimal APE and can be used for different simulations. However, an optimal parameter selection requires cross-validation.

The number of visits (v) needed for each state-action pair is independent of network size, and near-optimal performance can be achieved with a small v value, around $v \approx 50$. Likewise, the proposed algorithm allows us to have short trajectories (l) while ensuring several key factors: (i) achieving near-optimal APE performance, (ii) minimizing runtime and computational requirements per trajectory, (iii) capturing sufficient samples from neighboring states, (iv) preserving the significance of the initial state despite discounting in longer trajectories, and (v) avoiding redundant paths and loops that provide no new samples.

Increasing the number of Markovian environments (K) reduces the average runtime complexity of the algorithm [2]. However, there are several points to consider: (i) Emphasizing high-order node relationships may lead to the loss of low-order node relationships. (ii) $\mathbf{L}^{(n)}$ converges to a fixed tensor as n increases, rendering samples from corresponding SMEs redundant and potentially degrading performance. (iii) Memory requirements increase. Hence, K should be small enough to avoid these drawbacks while also increasing with the network size to prevent an increase in other parameters (specifically v, l) and reduce runtime complexity.

The learning rate (α_t) must adhere to the convergence conditions of Q-learning [3] and have a suitable decay to adjust the learning speed. We assume the form $\alpha_t = \frac{1}{1+\frac{t}{c_1}}$, where $c_1 > 0$ determines the decay rate and should increase with $|\mathcal{S}|$, v, l, and K. On the other hand, the parameter ϵ_t is essential for balancing exploration and exploitation. We use the form $\epsilon_t = \max((c_2)^t, c_3)$, where $c_2 > 0$ adjusts the decay rate, and $0 < c_3 \ll 1$ determines the minimum exploration probability. As the system parameters increase, c_2 should also increase to ensure sufficient exploration. Furthermore, c_3 should be small and positive, allowing for exploration with a low probability when the policy is nearly converged.

In general, a small u encourages the algorithm to rely on the Q-functions of different Markovian environments, promoting extensive exploration of the state-action space. This not only accelerates learning in the early stages by promoting exploration and yielding rapid initial progress but also facilitates a better trade-off between exploration and exploitation. In contrast, a large u prioritizes the ensemble Q-function output, enabling the algorithm to leverage the knowledge acquired from prior experiences, thereby expediting the overall convergence process. The emphasis on the ensemble output also contributes to stability throughout the learning process, resulting in more consistent Q-functions. Although different choices are possible, we find that a constant update ratio $u_t = 0.5$ gives a favorable balance between performance and complexity, facilitating convergence of Algorithm 1 across different network sizes. Our proposed approach is easier to tune compared to the algorithms presented in our prior work, particularly regarding the parameter u_t, which does not require extensive fine-tuning.

References

[1] Talha Bozkus and Urbashi Mitra. Link analysis for solving multiple-access mdps with large state spaces. *IEEE Transactions on Signal Processing*, 71:947–962, 2023.

[2] Talha Bozkus and Urbashi Mitra. Ensemble graph Q-learning for large scale networks. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[3] Francisco S Melo. Convergence of Q-learning: A simple proof. *Institute Of Systems and Robotics, Tech. Rep*, pages 1–4, 2001.