Supplementary Appendix

Talha Bozkus and Urbashi Mitra

I. Proof of Corollary 1

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 on the network models

A. SISO wireless network model with independent channels

For the parameters C_0, η, l, P_{rcv}, p, the same values as in [1] are used. The buffer size Q and the number of channel partitions H are chosen close to each other. The network size is formed by increasing Q and H comparably.

B. MISO energy harvesting wireless network with Gaussian interference channels and multiple relays

We firstly explain the setting given in Fig.3. Let h_{ij} be the channel gain between the i^{th} transmitter and j^{th} relay for i=1,2,3 and j=1,2. Let h_{Rk} be the channel gain between the receiver and k^{th} relay for k=1,2. Let h_{lR} be the channel gain between the l^{th} transmitter and receiver. Let the inputs to the transmitters be x_1,x_2,x_3, and the receiver output Y. The overall system is governed by the following equations:

$$Y_{R_1}=h_{11}x_1\mathbf{1}\{a_1=1\}+h_{21}x_2\mathbf{1}\{a_2=1\}+h_{31}x_3\mathbf{1}\{a_3=1\}+N(0,\sigma^2). \tag{7}$$

$$Y_{R_2}=h_{12}x_1\mathbf{1}\{a_1=2\}+h_{22}x_2\mathbf{1}\{a_2=2\}+h_{32}x_3\mathbf{1}\{a_3=2\}+N(0,\sigma^2). \tag{8}$$

$$Y=h_{R1}Y_{R_1}+h_{R2}Y_{R_2}+h_{1R}x_1\mathbf{1}\{a_1=0\}+h_{2R}x_2\mathbf{1}\{a_2=0\}+h_{3R}x_3\mathbf{1}\{a_3=0\}. \tag{9}$$

The individual state of the i^{th} transmitter at time t is defined as a tuple $s_{t,i}=(b_{t,i},h_{t,i})$, where $b_{t,i}$ is the battery state and $h_{t,i}$ is the channel state with $b_{t,i}\in\{0,1,...,N-1\}\ \forall i$. We define the battery probability transition tensor (PTT) of each transmitter $\mathbf{B}$ with dimensions $N\times N\times 3$, which stores the probability of transitioning between battery states under different actions. The structure of the battery PTT is similar to the buffer PTT in [2] except that there are now three actions.

Then, the joint battery PTT $\bar{\mathbf{B}}$ is obtained as $\bar{\mathbf{B}} = \mathbf{B} \otimes \mathbf{B} \otimes \mathbf{B}$. On the other hand, the evolution of different channel gains is characterized by the standard Gilbert-Elliot channel model with different probabilities as follows:

$$\begin{aligned} h_{11}, h_{12}, h_{21}, h_{22}, h_{31}, h_{32} &\in \{0,1\} \text{ with } (p_1, q_1), \\ h_{1R}, h_{2R}, h_{3R} &\in \{0,1\} \text{ with } (p_2, q_2), \\ h_{R1}, h_{R2} &\in \{0,1\} \text{ with } (p_3, q_3), \end{aligned}$$

where p_1, p_2, p_3 are the probabilities of transitioning from state 0 (good) to 1 (bad), and q_1, q_2, q_3 are the probabilities of transitioning from state 1 (bad) to 0 (good). Different (p, q) can fully characterize the different channel gains. For example, the channel conditions between the transmitter and relay, as well as the relay and receiver are likely to be better than that of direct channel between the transmitter and receivers (due to shorter distance and less interference) Hence, we can assume that $p_1 = p_3 \ll p_2$ and $q_2 \ll q_1 = q_3$ (*i.e.* the channel is more likely to be bad for the direct channel). We then construct three channel transition probability matrices: $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ for each three different distributions, and obtain the joint probability transition matrix $\bar{\mathbf{C}}$ as $\bar{\mathbf{C}} = \mathbf{C}_1 \otimes \mathbf{C}_2 \otimes \mathbf{C}_3$.

In the end, the probability transition tensor of the overall system $\mathbf{P}$ is obtained as $\mathbf{P} = \bar{\mathbf{B}} \otimes \bar{\mathbf{C}}$. The optimization problem is to choose optimal actions a_1, a_2, a_3 in order to minimize the overall cost function defined as follows:

$$\mathbf{c}(\{s_{t,i}\}_{i=1}^{3}) = -\alpha_1 Y + \alpha_2 \sum_{i=1}^{3} \sum_{j=1}^{2} x_i \mathbf{1}\{a_i = j\} + \alpha_3 \sum_{i=1}^{3} (1 - \frac{b_{t,i}}{N}), \tag{10}$$

where the first term is the negative throughput, the second term is the drop cost that balances the load on the relays (*i.e.* if multiple transmitters choose to use the same path through relay-1 or relay-2, there may be performance degradation), and the third term is the total amount of battery consumed (*i.e.* if $b_{t,i} = 0$, it means the battery is empty, and the corresponding cost is very large), with $\alpha_1, \alpha_2, \alpha_3$ being the weights.

For the case illustrated in Fig.3, we use the following numerical parameters: $\sigma^2 = 1, x_1 = x_2 = x_3 = 1, p_1 = q_2 = p_3 = 0.2, q_1 = q_3 = p_2 = 0.8, \alpha_1 = \frac{1}{9}, \alpha_2 = \frac{1}{6}, \alpha_3 = \frac{1}{3}$.

The network size of this network is formed as follows: the interval [0, 2000] is formed by increasing N with 2 transmitters and 1 relay, [2000, 4000] is formed by increasing N with 3 transmitters and 1 relay, [4000, 8000] is formed by increasing N with 3 transmitters and 2 relay. For each different setting, the same methodology as above is followed.

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 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 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. The performance of different distance metrics in Algorithm 1

We firstly use the l_2 distance between the Q-functions as a similarity measure, and update the line-8 of Algorithm as follows:

$$\mathbf{w}_t^{(n)} \leftarrow -\frac{1}{|\mathcal{S}|} \sum_{s=1}^{\mathcal{S}} \|\mathbf{Q}_t^{(1)}(s,:) - \mathbf{Q}_t^{(n)}(s,:)\|^2 \tag{11}$$

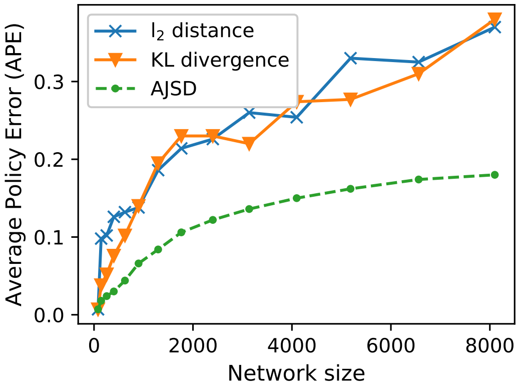

Fig. 1: APE vs network size across different similarity metrics

Herein, a larger distance leads to a smaller weight.

We also use the KL divergence between the Q-functions as a similarity measure, and update the line-8 of Algorithm as follows:

$$\mathbf{w}_t^{(n)} \leftarrow -\frac{1}{|\mathcal{S}|}\sum_{s=1}^{\mathcal{S}} \mathrm{KL}\left(\hat{\mathbf{Q}}_t^{(1)}(s,:)\|\hat{\mathbf{Q}}_t^{(n)}(s,:)\right) \tag{12}$$

Herein, a larger KL divergence leads to a smaller weight.

We carry out the simulations with the same settings in Section IV-B. The APE of the proposed algorithm across network size is shown in Fig.1. Clearly, using AJSD as a distance metric to compare the Q-functions in Algorithm 1 results in $\%50$ less APE. Moreover, we observed that slight changes in network size can cause significant variations in APE when l_2 distance or KL divergence is used, whereas AJSD provides a more robust distance measure.

References

[1] Libin Liu, Arpan Chattopadhyay, and Urbashi Mitra. On solving mdps with large state space: Exploitation of policy structures and spectral properties. *IEEE Transactions on Communications*, 67(6):4151–4165, 2019.

[2] 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.