In the second exploration strategy, $\epsilon$ decreases linearly between the values $\epsilon_{t=1} = 2\bar{\epsilon}$ and $\epsilon_{t=K} = 0$:

$$\epsilon_t = 2\bar{\epsilon} \left( \frac{K - t}{K - 1} \right)$$ (32)

In the third exploration strategy, the algorithm does pure exploration during the $\epsilon f$ first learning iterations of each TDMA time slot, then pure exploitation during the remaining $(1 - \epsilon)f$ last learning iterations of the time slot (see Figure 4):

$$\epsilon_t = \begin{cases} 1 & \text{if } t - \left\lfloor \frac{t}{f} \right\rfloor < \epsilon f \\ 0 & \text{otherwise} \end{cases}$$ (33)

Note that for both the sensing time and power allocation problems, the agents have an imperfect knowledge of the state of the environment. The state represented by an agent at each iteration of the Q-learning algorithm is actually an imperfect estimation of the environment state. In this case, the convergence demonstration of single agent Q-learning [21] does not hold. However, multi-agent Q-learning algorithms have been successfully applied in multiple scenarios [14] and in particular to cognitive radios [13], [15], [22]. Numerical results will show that both Q-learning algorithms presented in this paper converge as well.

V. NUMERICAL RESULTS

A. Sensing Time Allocation Algorithm

Unless otherwise specified, the following simulation parameters are used: we consider $N = 2$ nodes able to transmit at a maximum data rate $C_{H_{0,1}} = C_{H_{0,2}} = 0.6$. They each require a data rate $\hat{R}_1 = \hat{R}_2 = 0.1$. One node has a sensing channel characterized by $\gamma_1 = 0$ dB and the second one has a poorer sensing channel characterized by $\gamma_2 = -10$ dB.

It is assumed that the primary network transition probabilities are $p_{00} = 0.9$, $p_{01} = 0.1$, $p_{10} = 0.2$ and $p_{11} = 0.8$. The target detection probability is $P_D = 0.95$.

We consider $s = 10$ samples per time slot and $r = 100$ time slots per learning periods. The Q-learning algorithm is implemented with a learning rate $\alpha = 0.5$ and a discount rate $\gamma = 0.7$. The chosen exploration strategy consists in using $\epsilon = 0.1$ during the first $K/2$ iterations and then $\epsilon = 0$ during the remaining $K/2$ iterations.

Figure 5 gives the result of the Q-learning algorithms when no exploration strategy is used ($\epsilon = 0$). It is observed that after 430 iterations, the algorithm converges to $M_1 = M_2 = 4$ which is a sub-optimal solution. The optimal solution obtained by minimizing Equation (11) is $M_1 = 4, M_2 = 1$ (as the second node has a low sensing SNR, the first node has to contribute more to the sensing of the primary signal). After convergence, the normalized average throughputs are $\hat{R}_{1,C_{H_{0,1}}} = \hat{R}_{2,C_{H_{0,2}}} = 0.108$ whereas the optimal normalized average throughputs are $\hat{R}_{1,\text{opt},C_{H_{0,1}}} = 0.096$ and $\hat{R}_{2,\text{opt},C_{H_{0,2}}} = 0.144$ and lead to an inferior global cost in Equation (11).

Figure 6 gives the result of the Q-learning algorithms when the exploration strategy described at the beginning of this Section is used. It is observed that the algorithm converges to the optimal solution defined in the previous paragraph.

Table I compares the performance of the sensing time allocation algorithm implementation based on the cooperative cost function defined by Equation (23) with the one based on the competitive cost function defined by Equation (20). The cooperative cost function penalizes the actions that lead to a higher than required throughput and as a result performs better (i.e. gives higher realized average throughputs $\hat{R}_j$) than the competitive cost function, in different scenarios. In particular, it helps achieve fairness among the nodes when one of the nodes has a lower sensing SNR (in which case the other nodes tend to contribute more to the sensing) or when one of the nodes has an inferior channel capacity (in which case this node...