POLICY RECOGNITION VIA EXPECTATION MAXIMIZATION

Adrian Šošić [1,2] *Abdelhak M. Zoubir* [1] *Heinz Koeppl* [2]

[1] Signal Processing Group, Technische Universität Darmstadt, Germany
[2] Bioinspired Communication Systems, Technische Universität Darmstadt, Germany

ABSTRACT

Learning from Demonstrations (LfD) has proven to be a powerful concept for solving optimal control problems in high-dimensional state spaces where demonstrations can be used to facilitate the search for efficient control policies. However, many existing LfD approaches suffer from either theoretical, practical, or computational drawbacks such as the need to learn a latent reward model, to monitor the expert's controls, or to repeatedly solve potentially demanding planning problems. In this work, we consider the LfD objective from a system identification perspective and propose a probabilistic policy recognition framework based on expectation maximization that operates directly on the observed expert trajectories, avoiding the aforementioned problems. Using a spatial prior over policies, we are able to make accurate predictions in regions of the state space that are scarcely explored.

Index Terms— learning from demonstrations, imitation learning, expectation maximization, system identification, Markov random fields

1. INTRODUCTION

Dealing with complex or interacting systems whose (joint) state space is large remains a challenge. For such systems, the exploration of the state space typically becomes problematic and off-the-shelf methods often fail at finding control policies that efficiently exploit the system dynamics, or involve the risk of letting the system run into undesired or unsafe states [1]. Yet, in many daily situations, we regularly observe specialized behavior that is highly optimized and focused to accomplish complex tasks. In such cases, Learning from Demonstrations (LfD) [2, 3] offers a promising alternative to classical reinforcement learning approaches.

While in principle all LfD approaches pursue the same goal, that is, building models of behavior based on demonstrations, many different ideas and concepts have been proposed to tackle the problem in the past, some of them formulated as trajectory matching problems [4, 5], others as inverse reinforcement learning problems [6, 7, 8] or supervised learning problems [9, 10, 11]. Many of these methods, however, suffer from either theoretical, practical, or computational drawbacks such as the need to learn the latent reward structure of the system [12], to monitor the expert's controls [2], or to repeatedly solve potentially demanding planning problems [7]. As we believe that LfD is a key concept to system identification, we aim at developing methods that overcome these limitations and that are able to learn by pure observation.

In this work, we specifically consider the problem of policy recognition, that is, estimating a system's policy from observed trajectory data. To this end, we frame the LfD objective as a probabilistic inference problem which we address using an expectation maximization framework. A similar approach has already been taken in [13], yet with a different intention. Here, the authors assume that the environment can be explored actively by the observer and use the gathered knowledge about the expert behavior to guide the exploration of the state space. In this paper, we not only provide a more general formulation of the inference problem than the one proposed in [13], accounting for the uncertainty inherent in the trajectory data, but also drop the assumption of having the possibility to interact with the environment, and therefore focus on the harder problem of learning solely from the available trajectory data. Along these lines, we argue (underpinned by our empirical results) that optimal policies tend to have intrinsic structures that stem from regularities of the underlying system dynamics. By implicitly encoding these structures into our model, we are able to accurately recover the system's policy even when there are only few observations available and parts of the state space are not visited by the expert.

2. METHODOLOGY

Suppose we are given a dynamic system in the form of a Markov decision process (MDP) [14] of which we can observe a noisy system trajectory of length T.[1] The state and the action space of the system shall be denoted by $\mathcal{S}$ and $\mathcal{A}$, respectively, both having finite numbers of elements $|\mathcal{S}|$ and $|\mathcal{A}|$. Our goal is to find an estimate of the system's policy π that can explain the observed expert behavior reasonably well. In particular, we focus on Markovian deterministic policies, $\pi : \mathcal{S} \to \mathcal{A}$, as we assume the expert follows an optimal deterministic control strategy (which is guaranteed to exist

[1] It is straightforward to extend our arguments to multiple trajectories as they can be treated conditionally independent given the system parameters.

[15]). Accordingly, we can express the policy as a collection of action assignments, $\pi = (\pi_1, \dots, \pi_{|\mathcal{S}|}) \in \mathcal{A}^{|\mathcal{S}|}$, and may write the joint distribution over the true state sequence $s = (s_1, \dots, s_T)$ and observations $y = (y_1, \dots, y_T)$ as

$$p(s, y \mid \pi) = p(s_1) \prod_{t=1}^{T-1} p(s_{t+1} \mid s_t, \pi_{s_t}) \prod_{t=1}^{T} p(y_t \mid s_t). \quad (1)$$

Although it is generally possible to extend our reasoning about the system parameters beyond the policy (see e.g. [13]), we assume that both the transition model $p(s_{t+1} \mid s_t, a_t)$ and the observation model $p(y_t \mid s_t)$ are known. The reason is that, in a real scenario, we often have a very precise idea about the physical capabilities of a system, allowing us to judge which state transitions are feasible and which are not.[2] In the same way, we are typically aware of the physical limitations of our measurement devices that provide us with trajectory data. However, a similar argument does not hold for the policy since, in general, policies can differ drastically depending on the particular task being performed by the agent (thinking of everyday human behavior, for example). Nevertheless, we will shortly see that many optimal policies follow a common pattern and that it is possible to capture their structure in our model.

2.1. Maximum likelihood estimation

One straightforward approach to solve the policy recognition problem is via *maximum likelihood* (ML) estimation from the trajectory data, i.e.

$$\pi^{\mathrm{ML}} = \arg\max_{\pi} p(y \mid \pi),$$

which can be performed using the expectation maximization algorithm [16]. For this purpose, we first compute the conditional expectation of the log of the joint density in Eq. (1) for an initial guess π' of the policy (E-step), i.e.

$$\begin{aligned} Q^{\mathrm{ML}}(\pi, \pi') &= \sum_{s \in \mathcal{S}^T} p(s \mid y, \pi') \log p(s, y \mid \pi) \\ &\stackrel{\mathrm{c}}{=} \sum_{t=1}^{T-1} \sum_{s_{t+1} \in \mathcal{S}} \sum_{s_t \in \mathcal{S}} p(s_{t+1}, s_t \mid y, \pi') \log p(s_{t+1} \mid s_t, \pi_{s_t}). \end{aligned}$$

Herein, $\stackrel{\mathrm{c}}{=}$ indicates equality up to an additive constant. The posterior distribution $p(s_{t+1}, s_t \mid y, \pi')$ can be efficiently computed using the Baum-Welch algorithm [17]. Maximizing the above function w.r.t. π (M-step) is guaranteed to monotonically increase the likelihood of the estimate such that we converge to a local maximum of the likelihood function by iterating between both steps. Reordering the summations, we can highlight the individual contributions of each parameter π_i,

$$\begin{aligned} Q^{\mathrm{ML}}(\pi, \pi') &= \sum_{i=1}^{|\mathcal{S}|} Q_i^{\mathrm{ML}}(\pi, \pi') \\ &= \sum_{i=1}^{|\mathcal{S}|} \sum_{t=1}^{T-1} \sum_{s_{t+1} \in \mathcal{S}} p(s_{t+1}, s_t = i \mid y, \pi') \log p(s_{t+1} \mid s_t = i, \pi_i). \end{aligned}$$

In particular, we notice that the function decouples into distinct terms, $Q_i^{\mathrm{ML}}(\pi, \pi')$, $i \in \{1, \dots, |\mathcal{S}|\}$, that can be optimized independently. Unfortunately, we can also see that the maximum likelihood approach offers no means to reason about the policy at states that are far from the given trajectories. More precisely, if $p(s_t = i \mid y)$ is zero (meaning that state i is incompatible with the observed trajectory data under the assumed observation model), then also $p(s_{t+1}, s_t = i \mid y, \pi')$ is zero and so is the corresponding value Q_i^{ML}, irrespective of the particular assignment of π_i. In other words, the ML solution can not extrapolate to regions of the states space where we have observed no data.

2.2. Maximum a posteriori estimation

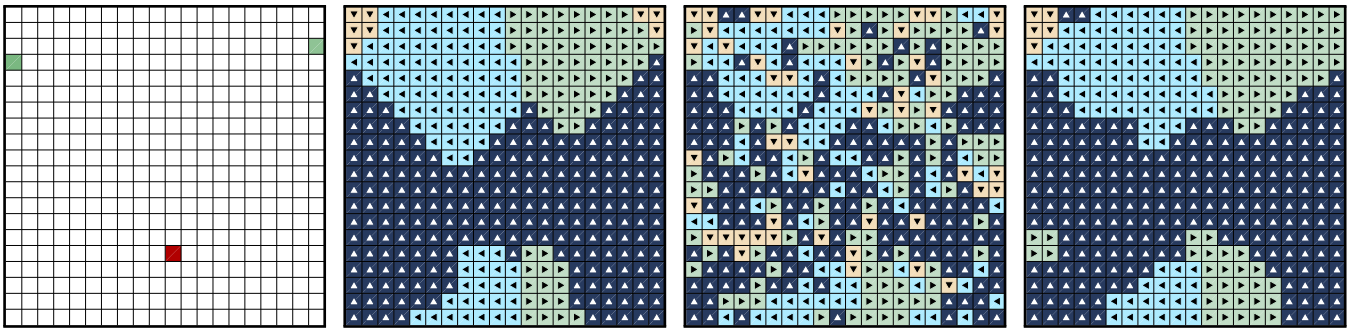
While the ML approach can solve the policy recognition problem only partially, we shall now see that it is possible to make accurate predictions if we account for certain properties of the expert policy. In particular, we argue that in most realistic scenarios optimal policies show pronounced correlation structures that can be exploited in order to enhance the prediction quality. The reason for this can be explained as follows: as actions, states and the transition model of a system are often linked to physical processes (e.g. when controlling a robot), they typically give rise to smooth dynamics in the sense that (physically) similar actions will result in similar state transitions. This is true for most systems with continuous state or action spaces where the smoothness is given naturally, but also for discrete systems that approximate continuous ones or that show certain regularities in their dynamics (see an example in Section 3). Consequently, for such systems there is a high chance that two nearby[3] states are assigned the same action under an optimal policy that is related to a certain task.

We can easily incorporate this property into our framework by using an appropriate spatial prior model over policies, such as a Markov random field (MRF) [18, 19] (e.g. a Potts model [20]), encoding the spatial smoothness, i.e.

$$p(\pi) \propto \prod_{i=1}^{|\mathcal{S}|} \exp\left(\frac{\beta}{2} \sum_{j \in \mathcal{N}_i} J_{\pi_i, \pi_j} \delta(\pi_i, \pi_j) \right). \quad (2)$$

[2] In fact, it could be the task of the inference algorithm to figure out which of the hypothetical actions are actually played by the expert, yet we assume the action space to be known. Also, as we do not need to solve the underlying control problem (where errors in the transition model usually accumulate), we can tolerate deviations from the true system dynamics as long as we can generally discriminate between actions by observed one-step transitions.

[3] The term "nearby" should here be understood in the context of the particular problem at hand and the underlying system dynamics.



(a) latent reward signal (b) true expert policy (c) ML estimate (d) MAP estimate ($\beta = 2$)

Fig. 1: Example result of the proposed algorithm. Both policy estimates are obtained from 200 expert trajectories, each consisting of four state transitions. (a) Latent reward signal to the agent (green color indicates positive reward, red color stands for negative reward). (b) True policy of the agent computed from the reward model in (a). (c) ML estimate of the policy based on the observed expert trajectories. States for which there is no evidence are assigned random actions (corresponding to the MAP solution under a uniform prior, i.e. $\beta = 0$). (d) MAP estimate resulting from the proposed Markov random field model.

Herein, δ stands for Kronecker's delta and $\beta \in [0, \infty)$ is called the inverse temperature, controlling the strength of the prior. $J_{m,n}$ denotes the $(m,n)^{\text{th}}$ element of the matrix J, quantifying the (physical) "similarity" between actions m and n at neighboring states. More specifically, any two actions are considered to be similar whenever their corresponding value in J is large. As this relationship is undirected, J is a symmetric matrix. The neighborhood $\mathcal{N}_i$ further determines the range of the spatial influence of parameter π_i on the remaining parameters, which are in the sequel denoted as $\pi_{\setminus i}$. In particular, the neighborhood structure is symmetric, too, meaning that if state i is a neighbor of state j, then also j is a neighbor of i.

Using the above prior model, we reformulate our goal and accordingly aim for the *maximum a posteriori* (MAP) estimate of the policy given the observed trajectory data, i.e.

$$\begin{aligned}\pi^{\text{MAP}} &= \arg\max_{\pi} p(\pi \mid y) \\ &= \arg\max_{\pi} \big(\log p(y \mid \pi) + \log p(\pi) \big).\end{aligned}$$

Here, the first term represents the likelihood of the observed trajectory data and the second term corresponds to the MRF prior. Again, the estimate can be obtained via the EM algorithm by simply replacing the objective function in the E-step with the following one [16],

$$Q^{\text{MAP}}(\pi, \pi') = Q^{\text{ML}}(\pi, \pi') + \log p(\pi). \tag{3}$$

Unfortunately, the resulting M-step is generally infeasible under the MRF prior as the function no longer decouples into individual terms, requiring an optimization over an exponentially large space [19]. However, we can still arrive at a local maximum by applying the iterated conditional modes (ICM) algorithm [21] which optimizes one variable at a time, giving rise to a coordinate ascent type of procedure. In particular, we can define a local optimization function Q_i^{MAP} for each variable π_i, whose value depends on the current assignment of the remaining variables $\pi_{\setminus i}$ via the neighborhood $\mathcal{N}_i$,

$$\begin{aligned}Q_i^{\text{MAP}}(\pi_i, \pi_{\setminus i}, \pi') &= Q_i^{\text{ML}}(\pi, \pi') + \log p(\pi_i \mid \pi_{\setminus i}) \\ &\overset{c}{=} Q_i^{\text{ML}}(\pi, \pi') + \beta \sum_{j \in \mathcal{N}_i} J_{\pi_i, \pi_j} \delta(\pi_i, \pi_j).\end{aligned}$$

Notice that the scaling factor $\frac{1}{2}$ in Eq. (2) has vanished here due to the symmetry properties of $\mathcal{N}_i$ and J. Optimizing these functions one after another w.r.t. their individual policy parameters π_i will in turn increase the value of the target function in Eq. (3), allowing us to gradually refine our estimate.

3. SIMULATION RESULTS

To test our model, we adopt the traditional Gridworld problem [14], which mimics the motion of a robot through a two-dimensional space (an example scenario is shown in Fig. 1). Here, the state space consists of 20×20 distinct states arranged on a grid. The transition model of the system is defined as follows: an agent living in the Gridworld can choose among four actions which correspond to the motions *up*, *left*, *down* and *right*. Taking a certain action, the agent will move in the corresponding direction with a probability of 60%. With a chance of 40%, however, it will randomly move in the other three directions or not move at all. If the resulting move makes the agent hit the boundary of the world, it will stay at its current position. At any point in time we observe the agent's true location with a probability of 60%, and with 40% chance we mistakenly spot him at either of the neighboring four states (with probability mass "lying outside" the world being shifted to the true location of the agent).

In order to connect the model with a policy, each state is assigned a reward with probability τ whose value is then

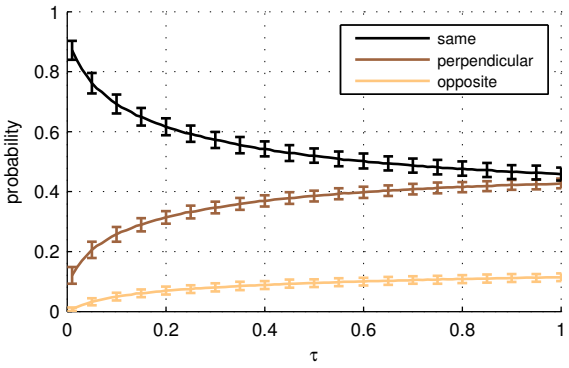


Fig. 2: Spatial smoothness of the optimal policies in the Grid-world scenario for different "densities" τ of the reward signal. The graphs show the estimated probabilities that two adjacent states are assigned the same action, actions resulting in perpendicular movements, or actions corresponding to opposite directions. Plotted are the mean values and standard deviations estimated from 1000 Monte Carlo runs.

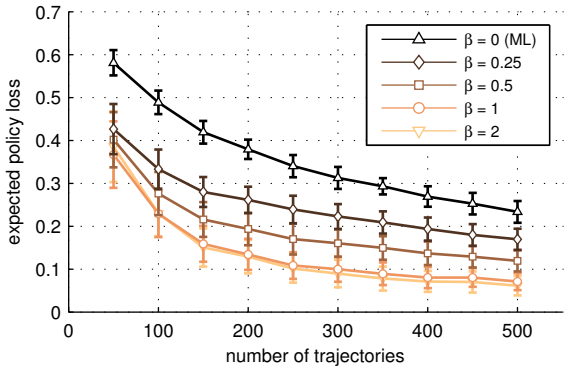


Fig. 3: Expected policy loss $\mathcal{L}(\pi, \hat{\pi})$ resulting from our policy estimates for different prior strengths β of the MRF as a function of the number of observed trajectories. Each trajectory consists of four state transitions. Plotted are the mean values and standard deviations estimated from 100 Monte Carlo runs. The results for $\beta = 1$ and $\beta = 2$ are barely distinguishable, indicating a suitable range for β in this setting.

drawn from a standard normal distribution so that finally rewards are distributed randomly across the state space. In case no state gets assigned any reward, the procedure is repeated. The expert policy is then found as the solution to the optimal control problem arising from the associated MDP. For our example, we used the infinite-horizon discounted model [14] with a discount factor of $\gamma = 0.9$. Having defined the model, we can generate the required trajectory data by executing the learned expert policy. In our example, the initial distribution $p(s_1)$ is chosen as the uniform distribution on the state space.

As a first result we observe that, depending on the constellation of rewards, an optimal policy for this setting is most likely going to be highly structured (see Fig. 1b as an example). This observation is perfectly in line with our reasoning in the last section. In fact, we can observe that neighboring states are assigned the same (or a similar) action with high probability. This is particularly true if the reward signal is sparse, but still holds in situations with dense rewards structures (see Fig. 2), justifying the use of the MRF prior as introduced in Eq. (2). In order to demonstrate that already a crude prior model can significantly improve the quality of the estimate, we adopt a simple four-state neighborhood structure according to which any two states are neighbors if their Manhattan distance on the grid is 1. Since we conduct the simulations in a sparse reward regime of $\tau = 0.01$, it is sufficient to consider only correlations between identical actions as they apparently capture most of the structure that is inherent in the expert policies (see again Fig. 2). For this reason, we choose the similarity matrix J as the identity matrix, ignoring all other dependencies across different actions.

The performance of both the ML and the MAP approach are now compared in terms of the following policy loss function,

$$\mathcal{L}(\pi, \hat{\pi}) = \frac{1}{|\mathcal{S}|} \sum_{i=1}^{|\mathcal{S}|} \delta(\pi_i, \hat{\pi}_i),$$

which measures the percentage of mismatch between the true policy π and the estimated policy $\hat{\pi}$. Fig. 3 shows the evolution of the expected policy loss over the amount of trajectory data provided to the algorithm for different prior strengths β, estimated from 100 Monte Carlo runs. Each of the trajectories consists of four state transitions. We can see that the proposed MRF approach clearly outperforms the ML solution with an average reduction in loss of about 65% caused by the MRF prior. Moreover, we notice a considerable improvement of the prediction quality when only little data is available.

4. CONCLUSION

Based on an expectation maximization framework, we presented a probabilistic approach to the policy recognition problem which allows us to infer a system's unknown policy from observed expert behavior. We showed that one can easily improve upon existing maximum likelihood approaches by exploiting the correlation structure of the expert policy using a spatial prior model over policies, especially when only a small amount of data is available. As our methodology is purely based on observations, it is particularly suited for building models of behavior in scenarios where there is no possibility to interact with the target system, which often appear in cognitive systems (e.g. cognitive radio [22, 23]), behavioral analysis and swarm dynamics. While the framework has proven to be efficient for discrete state and action spaces, future research will concentrate on extensions to continuous spaces where only a vanishing subset of the states can be observed.

5. REFERENCES

[1] P. Abbeel and A. Y. Ng, "Exploration and apprenticeship learning in reinforcement learning," in *Proc. 22nd International Conference on Machine Learning (ICML)*, 2005, pp. 1–8.

[2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469 – 483, 2009.

[3] S. Schaal, "Is imitation learning the route to humanoid robots?," *Trends in Cognitive Sciences*, vol. 3, no. 6, pp. 233 – 242, 1999.

[4] P. Englert, A. Paraschos, J. Peters, and M. P. Deisenroth, "Model-based imitation learning by probabilistic trajectory matching," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, May 2013, pp. 1922–1927.

[5] G. Maeda, M. Ewerton, R. Lioutikov, H. Ben Amor, J. Peters, and G. Neumann, "Learning interaction for collaborative tasks with probabilistic movement primitives," in *Proc. 14th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Nov 2014, pp. 527–534.

[6] S. Zhifei and E. M. Joo, "A review of inverse reinforcement learning theory and recent advances," in *Proc. IEEE Congress on Evolutionary Computation (CEC)*, Jun 2012, pp. 1–8.

[7] A. Y. Ng and S. J. Russell, "Algorithms for inverse reinforcement learning," in *Proc. 17th International Conference on Machine Learning (ICML)*, 2000, pp. 663–670.

[8] J. Hahn and A. M. Zoubir, "Inverse reinforcement learning using expectation maximization in mixture models," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 3721–3725.

[9] D. A. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *Neural Computation*, vol. 3, no. 1, pp. 88–97, Mar 1991.

[10] C. G. Atkeson and S. Schaal, "Robot learning from demonstration," in *Proc. 14th International Conference on Machine Learning (ICML)*, 1997, pp. 12–20.

[11] U. Syed and R. E. Schapire, "A reduction from apprenticeship learning to classification," in *Advances in Neural Information Processing Systems*, 2010, pp. 2253–2261.

[12] B. Piot, M. Geist, and O. Pietquin, "Learning from demonstrations: Is it worth estimating a reward function?," in *Machine Learning and Knowledge Discovery in Databases*, pp. 17–32. Springer, 2013.

[13] D. Verma and R. P. N. Rao, "Imitation learning using graphical models," in *Machine Learning: ECML 2007*, pp. 757–764. Springer, 2007.

[14] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[15] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994.

[16] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[17] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *The Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, Feb 1970.

[18] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, , no. 6, pp. 721–741, 1984.

[19] J. Konrad and E. Dubois, "Estimation of image motion fields: Bayesian formulation and stochastic solution," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Apr 1988, pp. 1072–1075.

[20] R. B. Potts, "Some generalized order-disorder transformations," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 48, pp. 106–109, Jan 1952.

[21] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 48, no. 3, pp. 259–302, 1986.

[22] J. Lunden, S. R. Kulkarni, V. Koivunen, and H. V. Poor, "Multiagent reinforcement learning based spectrum sensing policies for cognitive radio networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 858–868, Oct 2013.

[23] N. Mastronarde and M. van der Schaar, "Fast reinforcement learning for energy-efficient wireless communication," *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 6262–6266, Dec 2011.