2018 IEEE Conference on Decision and Control (CDC)
Miami Beach, FL, USA, Dec. 17-19, 2018

Online TD(λ) for discrete-time Markov jump linear systems

R. L. Beirigo, M. G. Todorov and A. M. S. Barreto

***Abstract*— This paper proposes a new approach for the optimal quadratic control of discrete-time Markov jump linear systems (MJLS), inspired on the temporal differences (TD) concepts of reinforcement learning. The method is *online*, in the sense that it is able to simultaneously apply and refine the currently available controller, and it is *transition model-free*, because there is no need for explicit knowledge of the Markov chain transition probabilities, provided it can be sampled or simulated. The strategy builds upon a previously proposed offline method and we hope will pave the way for developing and adapting reinforcement learning techniques for MJLS. The method is experimentally evaluated in Samuelson's macroeconomic model and in the control of a faulty robotic manipulator arm, performing favorably when compared to its offline predecessor.**

***Index Terms*— Markov jump linear systems, reinforcement learning, adaptive control, robotics.**

I. Introduction

As any system may present failures, it could be expected that a satisfactory control design would take them into account. More generally, when designing a control to a system, it could be advantageous to consider its susceptibility to *abrupt changes*. This potentially switching behavior is exemplified by likely catastrophic situations, as parts malfunction, infrastructure breakdown, economic collapse, environmental disasters, but may include any change in a system's dynamics, like the effect of the seasons on the weather, the demand variation in a service, and the change in robotic dynamics due to the warming of the joints, for instance. Despite there being a vast amount of recent literature concerning these matters (see [1], [2], [3], [4], for a small sample), open problems still abound.

Our work adheres to the formalization provided by the Markov Jump Linear Systems (MJLS), which, in essence, models the system's dynamical switching by a Markov chain, where each operational mode of the system corresponds to a state of the Markov chain, and the system's stochastic switching is modeled by the Markov chain's transition dynamics. The applicability of this model is broad, encompassing flight systems, networked control, robotics, and economics, for instance. In addition, MJLS are based upon solid theoretical foundations, which provides a prolific ground for investigation of this class of systems. Comprehensive discussions on the subject can be found in the books [5], [6], [7], [8], [9], [10].

Despite the flexibility brought by the MJLS model, the assumption of perfect knowledge of the corresponding transition probabilites may present itself as rather stringent, if not impeditive, to its practical use. This drove, in a vast gamma of scenarios, substantial research endeavor in the direction of *estimating* the transition probabilities, or assuming that known bounds are available for them. For instance, *polytopic uncertainty* investigates a polytope with known vertices that contains the unknown parameters [11], [12], [13]; the *multi-simplex* setup [14]; the *partially known* case [15]; the *norm-bounded* setup [16]; a randomized Gaussian modeling [17]; *maximum-likelihood estimation* [18], possibly with *transfer* [19]. The book [20] contains a recent discussion on the subject, along with some other setups.

In this paper, we are interested in a research direction that prescinds from the transition model altogether. Here, we investigate the optimal control of MJLS when there is no knowledge, nor approximation concerns, of and about the Markov chain transition probabilities, i.e., in this work, we are interested in *model-free* techniques.

Our work is closely related to the method proposed in [21], where the authors developed a model-free technique for incrementally computing the optimal quadratic control via Monte Carlo simulation, under the hypothesis that, prior to the system operation, a batch of previously sampled or simulated transitions of the Markov chain is available. Their work applied TD(λ), a reinforcement learning[1] technique that estimates the control performance through cost samples gathered at each transition, and used this estimate to improve the control, in a policy-iteration fashion. Despite the strong appeal presented by their method, it is *offline*, meaning that potentially useful sampled data cannot be immediately used to improve the control.

We propose an alternative to [21], that, besides being *model-free* by inheritance, is *online*. Hence, the proposed method is free from the "transition model operational burden", besides being able to *immediately utilize* sampled data to potentially improve the control. Our method is applied to Samuelson's macroeconomic model and a simulator of a robotic manipulator that is subject to joint failure. The experimental results suggest that applying the online strategy may present advantages when compared to its offline predecessor.

This paper is organized as follows. Some notation, basic definitions and relevant results from the literature are put

The authors are with the National Laboratory for Scientific Computing - LNCC/MCTIC, Av. Getúlio Vargas 333, Petrópolis, Rio de Janeiro, CEP 25651-070, Brazil. The third author is currently with Google DeepMind, London, UK. E-mails: rafaelb@lncc.br, todorov@lncc.br and amsb@lncc.br.
This work was partially supported by the Brazilian National Research Council – CNPq, Grants 421486/2016-3 and 461739/2014-3, and Coordenadoria de Aperfeiçoamento de Pessoal de Nível Superior – CAPES, Grant 1528969/2015-5.

[1]Also known as, or sharing ideas, premises, with, *neuro-dynamic programming*, *adaptive control*, encompassing as vast gamma of techniques and algorithms from a theoretical land with fuzzy borders.

together in Section II. The problem statement and proposed solution are provided in Section III. Sections IV and V feature numerical examples that illustrate the proposed approach. Some concluding remarks and directions for future work are the subject of Section VI.

II. Preliminaries

Consider a homogeneous Markov chain $\theta = \{\theta^k; k = 0,1,2,\ldots\}$ in a complete stochastic basis $(\Omega, \mathfrak{F}, \{\mathfrak{F}_k\}, \mathbb{P})$, with state space $\mathcal{S} \triangleq \{1,\ldots,N\}$, and transition matrix $P = [p_{ij}]$, i.e., $P \in \Pi \triangleq \{Q = [q_{ij}] \in \mathbb{R}^{N\times N}; q_{ij} \geq 0, \sum_{j=1}^{N} q_{ij} \equiv 1 \; \forall i,j \in \mathcal{S}\}$ is an N-by-N row-stochastic matrix. The mathematical expectation with respect to $\mathbb{P}$ will be represented as $\mathbb{E}$. For later use, consider also the following operation:

$$\mathcal{E}_i(X) \triangleq \sum_{j=1}^{N} p_{ij} X_j, \quad i \in \mathcal{S}, \tag{1}$$

for any N-sequence of matrices of the form $X = (X_1,\ldots,X_N)$, with $\mathcal{E}(X) \triangleq (\mathcal{E}_1(X),\ldots,\mathcal{E}_N(X))$. The set of all N-sequences of positive semidefinite n-by-n matrices, i.e., objects of the form $X = (X_1,\ldots,X_N)$ such that $0 \leq X_i \in \mathbb{R}^{n\times n}$ for all $i \in \mathcal{S}$, will be denoted $\mathbb{H}_N^{n+}$. Throughout the paper, $\|\cdot\|$ will stand for either the euclidean norm of vectors or the spectral norm of matrices.

The subject of our study is the following discrete-time control system:

$$\begin{cases} x^{k+1} = A_{\theta^k} x^k + B_{\theta^k} u^k \\ z^k = C_{\theta^k} x^k + D_{\theta^k} u^k \\ x^0 = \tilde{x}, \; \theta^0 = \tilde{\theta}, \end{cases} \tag{2}$$

where $x = \{x^k \in \mathbb{R}^{n_x}; k = 0,1,2,\ldots\}$ is the state, $u = \{u^k \in \mathbb{R}^{n_u}; k = 0,1,2,\ldots\}$ is the control, and $z = \{z^k \in \mathbb{R}^{n_z}; k = 0,1,2,\ldots\}$ is a controlled output. We shall use the shorthand notation $(\cdot)_{\theta^k} \equiv (\cdot)_i$ for $\theta^k = i$ (e.g., if $\theta^k = i$, then $A_{\theta^k} \equiv A_i$).

A. Jump linear quadratic optimal control

Defining

$$\begin{aligned} \mathbf{A}_i &\triangleq A_i + B_i F_i, \\ \mathbf{C}_i &\triangleq C_i + D_i F_i, \end{aligned} \quad i \in \mathcal{S}, \tag{3}$$

with the assumptions[2]

$$\begin{aligned} C_i' D_i &\equiv 0, \\ D_i' D_i &> 0, \end{aligned} \quad i \in \mathcal{S}, \tag{4}$$

and concentrating our attention on *state-feedback* controllers of the form

$$u = \{u^k = F_{\theta^k} x^k; \; k = 0,1,2,\ldots\}, \tag{5}$$

we have the *closed-loop* variant of system (2), given by

$$\begin{cases} x^{k+1} = \mathbf{A}_{\theta^k} x^k \\ z^k = \mathbf{C}_{\theta^k} x^k \\ x^0 = \tilde{x}, \; \theta^0 = \tilde{\theta}. \end{cases} \tag{6}$$

[2] As shown in [5, Chapter 4], the orthogonality between C and D is without loss of generality, and has the advantage of simplifying the subsequent derivations. The other assumption rules out singular controls by guaranteeing that every control action is penalized.

We are interested in optimizing the controlled output z^k of (6), with the corresponding control performance being measured by the *infinite horizon quadratic cost*

$$\mathfrak{J}(\tilde{\theta}, \tilde{x}, u) \triangleq \sum_{k=0}^{\infty} \mathbb{E}\left(\|z^k\|^2\right). \tag{7}$$

In this paper, we use the notion of stability for the system (6) as formalized in Definition 1.

Definition 1 (mean square stability): A control $u = \{u^k; k = 0,1,2,\ldots\}$ is said to *stabilize system* (2) *in the mean square sense* if, regardless of $x^0 \in \mathbb{R}^n$ and $\theta^0 \in \mathcal{S}$, the application of u in (6), yields

$$\lim_{k\to\infty} \mathbb{E}\left(\|x^k\|^2\right) = 0. \tag{8}$$

In such case, we say that (6) is *mean square stable* (MSS). ▽

In this work, we are interested in solving the following problem.

Problem 1 (Jump linear quadratic (JLQ)): Find a control $\hat{u}$ that satisfies

$$\mathfrak{J}(\theta^0, x^0, \hat{u}) \leq \mathfrak{J}(\theta^0, x^0, u), \forall u. \tag{9}$$

for system (6), and has the form (5). ▽

As the following lemma shows, an algebraic characterization of the closed-loop cost is possible, for mean square stabilizing controls of the form (5).

Lemma 1: If a given controller u of the form (5) stabilizes system (6) in the mean square sense, then the corresponding cost in (7) is given by

$$\mathfrak{J}(\theta^0, x^0, u) = x^{0\prime} X_{\theta^0} x^0, \tag{10}$$

where $\mathbb{H}_N^{n+} \ni X = (X_1,\ldots,X_N)$ is the unique solution of the following Lyapunov-like equation

$$X_i = \mathbf{A}_i' \mathcal{E}_i(X) \mathbf{A}_i + \mathbf{G}_i, \quad i \in \mathcal{S}, \tag{11}$$

where

$$\mathbf{G}_i \triangleq \mathbf{C}_i' \mathbf{C}_i, \quad i \in \mathcal{S}. \tag{12}$$

Proof: See [5, Chapter 4]. ■

As evidenced in Lemma 1, in order to solve Problem 1, we must be able to find *control gains*

$$F \triangleq (F_1,\ldots,F_N), \tag{13}$$

which give us a control of the form (5) that minimize (10), under the constraint (11), for each $i \in \mathcal{S}$. However, a quick examination of this setup clearly reveals two severe complications, which will be addressed in the sequel:

(i) The underlying optimization problem of minimizing (10) subject to (11) carries a great deal of implicit dependence upon the controller gains;

(ii) In order to solve (11) directly, we need to know the transition probabilities in (1).

III. Problem Statement and Main Result

In this paper we are interested in solving the JLQ problem in a *model-free*, *online* basis. For a technique to be qualified as *model-free*, it should be able to solve Problem 1 without the need of the probability matrix P (*transition* model). A model-free, *offline* method was proposed in [21], that applies *policy iteration* to refine an approximation of $\mathcal{E}(X)$, where $\mathbb{H}_N^{n+} \ni X = (X_1, \ldots, X_N)$ is the unique solution of (11), and then use this approximation to calculate a better policy. Each element Y^t of the approximation sequence $\{Y^t;\, t = 1, 2, \ldots\}$ is calculated by

$$Y_i^{t+1} = Y_i^t + \gamma_t \sum_{k=0}^{\infty} e_i^{t,k} \mathcal{D}_i^{t,k}(Y^t). \tag{14}$$

As can be seen from (14), Offline TD(λ) must wait until all the terms of the sum are processed before being able to update Y^t, which entails simulating or visiting the entire trajectory, before applying the corresponding information to improve the policy. We propose applying the *online incremental form* of (14), given by

$$\bar{Y}_i^{t,k+1} = \bar{Y}_i^{t,k} + \gamma_t e_i^{t,k} \mathcal{D}_i^{t,k}(\bar{Y}^{t,k}), \tag{15a}$$

$$\text{with} \quad \bar{Y}_i^{t,0} = \bar{Y}_i^t, \quad Y_i^{t+1} = Y_i^{t,N_t}, \tag{15b}$$

where N_t is the episode length,[3], the *stepsize* γ_t is assumed to satisfy the usual conditions

$$\sum_{t=0}^{\infty} \gamma_t = \infty \quad \text{and} \quad \sum_{t=0}^{\infty} \gamma_t^2 < \infty, \tag{16}$$

whereas $\mathcal{D}_i^{t,k}(\cdot)$, the *temporal difference*, is given by

$$\mathcal{D}_i^{t,k}(\cdot) = \Upsilon_i'^{t,k}(\mathbf{G}_{\theta_t^{k+1}} + \mathbf{A}'_{\theta_t^{k+1}}(\cdot)_{\theta_t^{k+1}}\mathbf{A}_{\theta_t^{k+1}} - (\cdot)_{\theta_t^k})\Upsilon_i^{t,k}, \tag{17}$$

with

$$\Upsilon_i^{t,k} = \begin{cases} I, & \text{if } k = 0, \\ \mathbf{A}_{\theta_t^k}\Upsilon_i^{t,k-1}, & \text{if } k > 0, \end{cases} \tag{18}$$

the *cost* is given by

$$\mathbf{G}_i \triangleq \mathbf{C}_i'\mathbf{C}_i, \quad i \in \mathcal{S}, \tag{19}$$

and the *eligibility coefficients* by

$$e_i^{t,k} = \begin{cases} 0, & k < k_i^t, \\ \lambda^{k-k_i^t}, & k \geq k_i^t, \end{cases} \tag{20}$$

where k_i^t is the first time that state i is visited in trajectory t, i.e.,

$$k_i^t = \inf_k \{\theta_t^k = i\}, \quad i \in \mathcal{S}. \tag{21}$$

At the *policy improvement* step, a new policy F is calculated by applying the current approximation $\bar{Y}$ of $\mathcal{E}(X)$ in the equation

$$F_i = -\left(B_i'\bar{Y}_i B_i + D_i'D_i\right)^{-1} B_i'\bar{Y}_i A_i, \quad i \in \mathcal{S}. \tag{22}$$

[3] N_t can be, for instance: the length of the trajectory θ^t, the number of iterates that the algorithm takes to meet some convergence criterion, or simply a prespecified number, such as a maximum number of iterates.

Algorithm 1 Online TD(λ)

Require: F is stabilizing

for $\ell = 1, \ldots, L$ **do** ▷ episode batch
  for $t = 1, \ldots, T$ **do** ▷ episode
    Initialize θ^0
    $e \leftarrow 0e$ ▷ "restart" eligibility coefficients
    for $k = 1, \ldots, K$ **do** ▷ step
      Perform $u^k \equiv F_{\theta^k} x^k$; observe $\theta^k \to \theta^{k+1}$
      if $e_{\theta^k} = 0$ **then**
        $e_{\theta^k} \leftarrow 1$ ▷ first visit to $i = \theta^k$
      end if
      $\bar{Y}_i \leftarrow \bar{Y}_i + \gamma_t e_i \mathcal{D}_i^{t,k}(\bar{Y}), \forall i \in \mathcal{S}$
      $e \leftarrow \lambda e$
    end for
  end for
  $F \leftarrow \mathcal{F}(\bar{Y})$
end for

Ensure: $\bar{Y} \approx \mathcal{E}(X)$

For this process to be consistent, the initial policy must be stabilizing, but Y may be initialized with arbitrary values [21].

The main result of the paper is the following theorem, which states that the online and offline algorithms converge to the same response when $t \to \infty$, as long as a relation is preserved between λ and the spectral radius r_σ of the linear operator $\mathbb{H}^{n^2} \ni \mathcal{H} = (\mathcal{H}_1, \ldots, \mathcal{H}_N)$, defined by

$$\mathcal{H}_j(\cdot) \triangleq \sum_{i=1}^{N} p_{ij}(\mathbf{A}_i \otimes \mathbf{A}_i)(\cdot)(\mathbf{A}_i' \otimes \mathbf{A}_i'), \tag{23}$$

where $\otimes$ stands for the Kronecker product [22].

Theorem 1: If $\lambda^2 r_\sigma(\mathcal{H}) < 1$ and, for some constants Ξ and $0 \leq \rho < 1$, we have $\mathbb{P}(N_t = k) \leq \Xi\rho^{-k}$, then with probability one,

$$\lim_{t\to\infty} \bar{Y}^t = \mathcal{E}(X). \tag{24}$$

Proof: Due to space restrictions, we refer the reader to [23] for the complete proof of the theorem. ■

Remark 1: The spectral radius condition of the preceding theorem is inherited from [21] and, as in that reference, is only a sufficient (but not necessary) condition for convergence, so the algorithms might even converge for larger values of λ. The hypothesis that the distribution of the episode length has exponentially vanishing tails, on the other hand, is borrowed from [24], and is quite natural if we bear in mind that the state of a MSS system of the form (6) tends to the origin exponentially fast [5, Theorem 3.9] in the mean square sense[4].

IV. Example: Samuelson's Macroeconomic Model

The macroeconomic model proposed by Samuelson [25] analyzes the business cycle by focusing on the *consumption*

[4] This condition is satisfied, for instance, if, for fixed $\varepsilon > 0$ we define $N_t = \inf_k \left\{k;\, \max_i \|\lambda^k \mathcal{D}_i^{t,k}(\bar{Y}^t)\| \leq \varepsilon\right\}$, which could be natural if we thought of ε as a machine precision-related parameter.

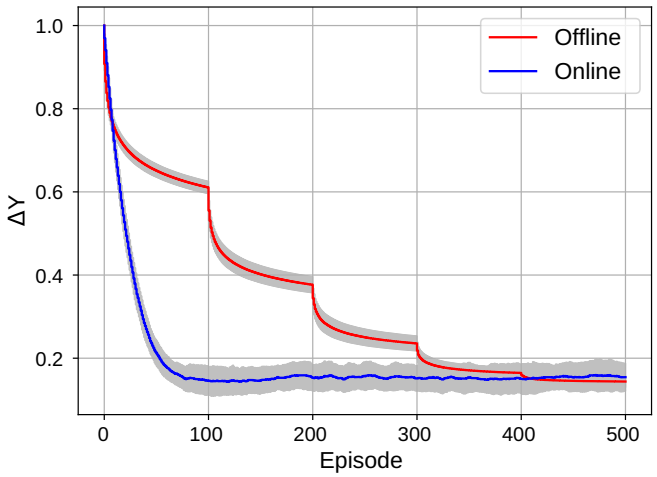

Fig. 1. Evolution of the approximation error through the episodes for the offline and online variants of TD(λ), normalized with respect to the approximand. Each curve corresponds to the mean over 100 repetitions, and the grey area shows the standard deviation. These results suggest that the *online* strategy of *immediately applying* sampled data in the approximation process may allow for a faster convergence, when compared with the *offline* variant, that accumulates sampled data and *applies them in batches*.

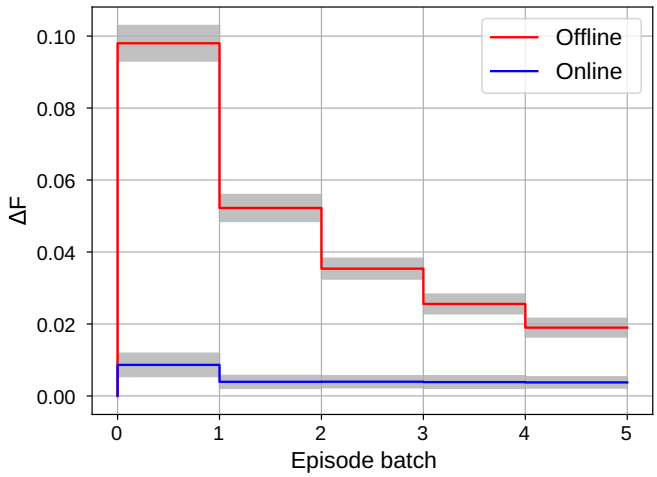

Fig. 2. Normalized error in the approximation of $\mathcal{F}(X)$ *versus* episodes, where X corresponds to the CARE solution being sought, F is the approximation of $\mathcal{F}(X)$ calculated as in (22), and ΔF is the approximation error represented by $\Delta F \equiv \sum_{i,j,k} \frac{\left|\left[\mathcal{F}_i(X)\right]_{jk} - \left[F_i\right]_{jk}\right|}{\left|\left[\mathcal{F}_i(X)\right]_{jk}\right|}$. These results correspond to the mean over 100 repetitions, where the standard deviation is depicted by the grey area.

and *investment* intentions, assuming they depend on the *level* and *pace* of growth of the economic activity, respectively. As it applies the *Keynesian multiplier* and the *accelerator theory of investment* to model the consumption and investment intentions, respectively, it is also known as *multiplier-accelerator* model.

In its classical form, Samuelson's macroeconomic model has the following realization of its state-space version:

$$x(k+1) = \begin{bmatrix} 0 & 1 \\ -\alpha & 1-s-\alpha \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k),$$

with s^{-1} corresponding to the *multiplier*, and α to the *accelerator* coefficients, respectively. By analysing historical data from the United States Department of Commerce, s and α were estimated by [25], grouping them into three main classes, each corresponding to a *modus operandi* of the system:

- $\theta^k = 1$, when both s and α are in mid-range;
- $\theta^k = 2$, when s is in low, or α is in high range; and
- $\theta^k = 3$, when s is in high, or α is in low range.

The model has the transition probability matrix given by

$$P = \begin{bmatrix} 0.67 & 0.17 & 0.16 \\ 0.30 & 0.47 & 0.23 \\ 0.26 & 0.10 & 0.64 \end{bmatrix}, \tag{25}$$

and, for this example, we applied the following system parameters:

$$A_1 = \begin{bmatrix} 0 & 1 \\ -2.5 & 3.2 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 1 \\ -4.3 & 4.5 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & 1 \\ 5.3 & -5.2 \end{bmatrix}$$

$$C_1'C_1 = \begin{bmatrix} 3.6 & -3.8 \\ -3.8 & 4.87 \end{bmatrix}, \quad C_2'C_2 = \begin{bmatrix} 10 & -3 \\ -3 & 8 \end{bmatrix}, \quad C_3'C_3 = \begin{bmatrix} 5 & -4.5 \\ -4.5 & 4.5 \end{bmatrix}$$

$$B_i \equiv \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad D_1'D_1 = 2.6, \quad D_2'D_2 = 1.165, \quad D_3'D_3 = 1.111.$$

Online TD(λ) was applied to solve this problem with the following settings: $L = 5$, $T = 100$, $K = 10$, $\lambda = 0.1$, and $\gamma_k = 0.1/k$. To account for the inherent randomness, the experiments were repeated, with the same settings, for 100 randomly generated realizations of the Markov chain, with the algorithm's performance being evaluated by comparison with Offline TD(λ) [21], with the same parameters. $\bar{Y}^{t=0}$ and $Y^{t=0}$ were initialized with zeros, and, as we were interested in evaluating both algorithms' performances when approximating $\mathcal{E}(X)$, $F^{\ell=0}$ was initialized with the optimal control gains in both algorithms.

Figure 1 shows the evolution of the approximation error through the episodes with respect to $\mathcal{E}(X)$ for both algorithms. We applied the normalized error formula given by

$$\Delta(\cdot) \triangleq \sum_{i,j,k} \frac{\left|\left[\mathcal{E}_i(X)\right]_{jk} - \left[(\cdot)_i\right]_{jk}\right|}{\left|\left[\mathcal{E}_i(X)\right]_{jk}\right|}, \tag{26}$$

with $i \in \mathcal{S}$, and j and k corresponding to the matrices' rows and columns, respectively, obtaining the approximation errors $\Delta(\bar{Y}^t)$ and $\Delta(Y^t)$ for the online and offline algorithms, respectively. Each curve in the graph corresponds to the mean over 100 repetitions, with the grey area corresponding to the standard deviation.

As it can be seen from Figure 1, both controls were able to increasingly refine the approximation of $\mathcal{E}(X)$. This result corroborates [21]'s, suggesting that the *sampling* processes were able to enable the *model-free* algorithms to completely prescind from the transition model when approximating $\mathcal{E}(X)$. We also see that Online TD(λ) presented a faster convergence, markedly at the initial time steps. This suggests that immediately applying the gathered data to update the estimate of $\mathcal{E}(X)$ at each transition may be beneficial to the

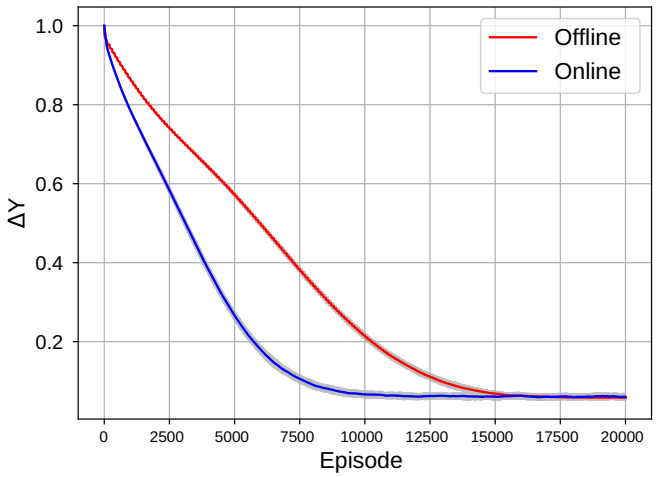

Fig. 3. Evolution of the approximation error through the episodes for the offline and online variants of TD(λ), normalized with respect to the approximand. Each curve corresponds to the mean over 100 repetitions, and the grey area shows the standard deviation. These results suggest that the *online* strategy of *immediately applying* sampled data in the approximation process may allow for a faster convergence, when compared with the *offline* variant, that accumulates sampled data and *applies them in batches*.

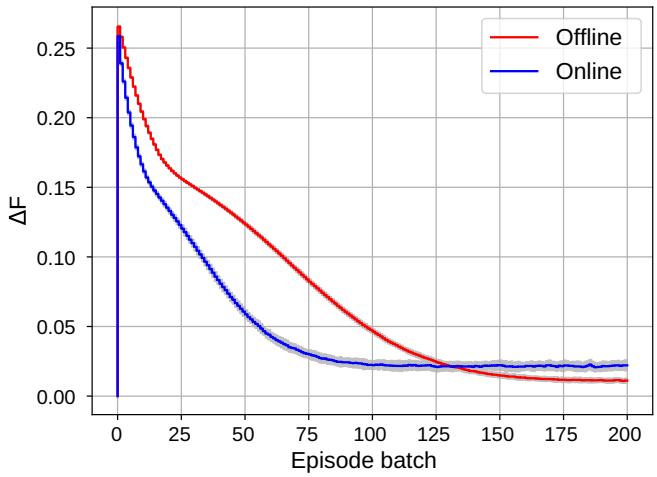

Fig. 4. Normalized error in the approximation of $\mathcal{F}(X)$ *versus* episodes, where X corresponds to the CARE solution being sought, F is the approximation of $\mathcal{F}(X)$ calculated as in (22), and ΔF is the approximation error represented by $\Delta F \equiv \sum_{i,j,k} \frac{\left|\left[\mathcal{F}_i(X)\right]_{jk} - \left[F_i\right]_{jk}\right|}{\left|\left[\mathcal{F}_i(X)\right]_{jk}\right|}$. These results correspond to the mean over 100 repetitions, where the standard deviation is depicted by the grey area.

approximation process.

The curve corresponding to the offline version shows a "scale-shaped" outline, and we hypothesize that this result could be due to intermittent local convergence, as we elaborate below. We note that Online TD(λ) updates its estimate of $\mathcal{E}(X)$ at each step of the episode, while Offline TD(λ) only updates it at the end of the episode. This could contribute for a *bias* in the approximation process, that could possibly explain the observed result of local convergence-like behavior.

Figure 2 shows evolution of the error in the control gains throughout the episode batches. We applied a normalized error formula similar to (26), given by

$$\Delta(\cdot) \triangleq \sum_{i,j,k} \frac{\left|\left[\mathcal{F}_i(X)\right]_{jk} - \left[(\cdot)_i\right]_{jk}\right|}{\left|\left[\mathcal{F}_i(X)\right]_{jk}\right|}, \tag{27}$$

with $i \in \mathcal{S}$, and j and k corresponding to the matrices' rows and columns, respectively, obtaining the approximation errors $\Delta(\mathcal{F}(\bar{Y}^t))$ and $\Delta(\mathcal{F}(Y^t))$ for the online and offline algorithms, respectively. As it can be seen from the graph, the improving quality in the approximation of $\mathcal{E}(X)$ was accompanied by better control gains in both algorithms, as expected. We can also see from the graph that Online TD(λ), using only the first episode batch, was able to obtain a policy with smaller error than the offline version, even after processing all the batches, which agrees with the results presented in Figure 1.

V. Example: Faulty Underactuated Robotic Arm

For a more challenging task, we evaluated our proposal using the Robust and Fault Tolerant Control Environment for Robots (CERob) [26], that simulates a robotic manipulator arm with 3 joints, each containing an actuator and a brake. Each actuator may be in the status *active* or *passive*, and when an actuator is active, its corresponding motor responds infallibly to control. If it is in the passive status, it remains on or off, heedless to control, and is considered in a condition of *fault*. The *state* of the system contains the current angle of each joint, and the arm's task is to go from an initial state to a final one. This is accomplished by applying torque to the joints, using the actuators, considering their possibility of failure.

For this example, we utilized the simulation setup where only one of the three joints' actuator is subject to failure. Due to space restrictions, we refer the reader to [27] for a detailed description of the parameters concerning the simulator.

Online TD(λ) was applied to solve this problem with the parameter values: $L = 200$, $T = 100$, $K = 10$, $\lambda = 0.1$, and $\gamma_k = 0.1/k$. We note that these settings are almost the same as the ones used for the Samuelson's macroeconomic model example. We used them as they gave satisfactory results, with the exception of L, which had to be considerably larger for this more complex example. Again, the example's inherent randomness was dealt with by repeating the experiment, with the same settings, for 100 realizations of the Markov chain, generated randomly. Offline TD(λ) [21], utilizing the same parameters, was used as a baseline for performance comparison. For both algorithms, $\bar{Y}^{t=0}$ and $Y^{t=0}$ were initialized with zeros and $F^{\ell=0}$ with the optimal control gains for the problem.

Figure 3 shows the evolution of the approximation error through the episodes with respect to $\mathcal{E}(X)$ for both algorithms, where the approximation error formula is given by (26). In the graph, each curve represents the mean over 100 repetitions, with the grey area corresponding to the

standard deviation. The results were analogous to the ones obtained for the simpler example, with both controls being able to satisfactorily approximate $\mathcal{E}(X)$. This corroborates the previous results about the capability of model-free techniques being able to satisfactorily approximate $\mathcal{E}(X)$ in the absence of transition parameters. For this more challenging task, Online TD(λ) showed again a relatively faster convergence, once more at the initial time steps. This corroborates the hypothesis that the immediate use of sampled data to update the estimate of $\mathcal{E}(X)$ may present convergence advantages.

The approximation error of the control gains calculated after each episode batch also shows a decline, as expected, qualitatively similar to the previous one, as shown in Figure 4. These experimental results suggest that Online TD(λ) was able to adequately approximate $\mathcal{E}(X)$ in a model-free, online basis.

VI. Concluding remarks

In this work, we presented a *model-free*, *online* technique to solve the *jump linear quadratic optimal control* problem (JLQ) for Markov jump linear systems (MJLS). The proposed algorithm is an extension of a *model-free*, *offline* technique, presented in [21], that is based on *temporal differences* with *eligibility traces* (TD(λ)), an approximation technique from the field of reinforcement learning (RL) [24], [28]. Our method inherits the model-free characteristic from the technique it extends [21], whose appeal comes from the ability of entirely prescinding from the knowledge of the transition probabilities when approximating the optimal control. This is advantageous not only in cases where the transition model is difficult to establish in advance, but also in cases where they are costly to approximate, or they can change with time (e.g., non-stationary domains).

Following the RL literature [24], [28], our strategy derives an *online* variant of [21] that is able to make immediate use of the sampled data, which can potentially improve the approximation process. Experimental results were presented for two domains, Samuelson's macroeconomic model [25], and a Robotic arm simulator subject to failure [27]. The experimental results show a favorable performance for the proposed technique, when compared to its offline predecessor.

Building upon the pioneer work from [21], who brought TD(λ) to the MJLS context through Offline TD(λ), we hope that, by proposing Online TD(λ), we may help to pave the way for reinforcing the bridge between *Markov jump linear systems* and *Markov decision processes* (MDP) solution techniques.

References

[1] A. S. Morse, Ed., *Control Using Logic Based Switching*. London: Springer-Verlag, 1997.

[2] C. G. Cassandras and J. Lygeros, *Stochastic Hybrid Systems*. Boca Raton, FL: Taylor & Francis, 2007.

[3] D. Liberzon, *Switching in Systems and Control*. Boston: Birkhäuser, 2003.

[4] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton, NJ: Princeton University Press, 2012.

[5] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, *Discrete-Time Markov Jump Linear Systems*, ser. Probability and Its Applications. New York: Springer-Verlag, 2005.

[6] O. L. V. Costa, M. D. Fragoso, and M. G. Todorov, *Continuous-Time Markov Jump Linear Systems*, ser. Probability and Its Applications. Heidelberg: Springer-Verlag, 2013.

[7] V. Dragan, T. Morozan, and A. Stoica, *Mathematical Methods in Robust Control of Linear Stochastic Systems*, ser. Mathematical concepts and methods in science and engineering. New York: Springer, 2006, vol. 50.

[8] M. Mariton, *Jump Linear Systems In Automatic Control*. New York: Marcel Dekker, 1990.

[9] E.-K. Boukas, *Stochastic Switching Systems: Analysis and Design*. Boston: Birkhäuser, 2005.

[10] V. Dragan, T. Morozan, and A. Stoica, *Mathematical Methods in Robust Control of Discrete-Time Linear Stochastic Systems*. Springer, 2010.

[11] L. El Ghaoui and M. A. Rami, "Robust state-feedback stabilization of jump linear systems via LMIs," *Internat. J. Robust Nonlinear Control*, vol. 6, no. 9/10, pp. 1015–1022, Nov. 1996.

[12] C. E. de Souza, "Robust stability and stabilization of uncertain discrete-time markovian jump linear systems," *IEEE Trans. Automat. Control*, vol. 51, no. 5, pp. 836–841, 2006.

[13] M. G. Todorov and M. D. Fragoso, "New methods for mode-independent robust control of Markov jump linear systems," *Systems Control Lett.*, vol. 90, pp. 38–44, 2016.

[14] C. F. Morais, M. F. Braga, R. C. L. F. Oliveira, and P. L. D. Peres, "$\mathcal{H}_2$ control of discrete-time Markov jump linear systems with uncertain transition probability matrix: improved linear matrix inequality relaxations and multi-simplex modeling," *IET Control Theory and Applications*, vol. 7, pp. 1665–1674, 2013.

[15] L. Zhang, E.-K. Boukas, and J. Lam, "Analysis and synthesis of markov jump linear systems with time-varying delays and partially known transition probabilities," *IEEE Trans. Automat. Control*, vol. 53, no. 10, pp. 2458–2464, 2008.

[16] M. Karan, P. Shi, and C. Y. Kaya, "Transition probability bounds for the stochastic stability robustness of continuous- and discrete-time Markovian jump linear systems," *Automatica*, vol. 42, no. 12, pp. 2159–2168, 2006.

[17] X. Luan, S. Zhao, and F. Liu, "$\mathcal{H}_\infty$ control for discrete-time Markov jump systems with uncertain transition probabilities," *IEEE Trans. Automat. Control*, vol. 58, no. 16, pp. 1566–1572, 2013.

[18] R. L. Beirigo, M. G. Todorov, and A. M. S. Barreto, "Count-based quadratic control of markov jump linear systems with unknown transition probabilities," in *Proc. of the 56th IEEE Conference on Decision & Control*, Melbourne, Australia, 2017.

[19] ——, "Transfer on count-based quadratic control of markov jump linear systems with unknown transition probabilities," in *Proc. of the Brazilian Conference on Dynamics, Control & Applications*, São José do Rio Preto, Brazil, 2017.

[20] L. Zhang, T. Yang, P. Shi, and Y. Zhu, *Analysis and Design of Markov Jump Systems with Complex Transition Probabilities*. Switzerland: Springer, 2016.

[21] O. L. V. Costa and J. C. C. Aya, "Monte Carlo TD(λ)-methods for the optimal control of discrete-time Markovian jump linear systems," *Automatica*, vol. 38, pp. 217–225, 2002.

[22] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 9, pp. 772–781, 1978.

[23] R. L. Beirigo, M. G. Todorov, and A. M. S. Barreto, "Online temporal differences for discrete-time Markov jump linear systems," *Automatica*, submitted, 2018.

[24] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.

[25] W. P. Blair and D. D. Sworder, "Feedback control of a class of linear discrete systems with jump parameters and quadratic cost criteria," *International Journal of Control*, vol. 21, pp. 833–844, 1975.

[26] A. A. G. Siqueira, M. H. Terra, and M. Bergerman, *Robust Control of Robots*. Heidelberg: Springer-Verlag, 2011.

[27] A. A. G. Siqueira and M. H. Terra, "Nonlinear and Markovian $\mathcal{H}_\infty$ controls of underactuated manipulators," *IEEE Trans. Control Syst. Technol.*, vol. 12, no. 6, pp. 811–826, Nov. 2004.

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