KL-LEARNING: ONLINE SOLUTION OF KULLBACK-LEIBLER CONTROL PROBLEMS

JORIS BIERKENS, HILBERT J. KAPPEN

ABSTRACT. We introduce a stochastic approximation method for the solution of an ergodic Kullback-Leibler control problem. A Kullback-Leibler control problem is a Markov decision process on a finite state space in which the control cost is proportional to a Kullback-Leibler divergence of the controlled transition probabilities with respect to the uncontrolled transition probabilities. The algorithm discussed in this work allows for a sound theoretical analysis using the ODE method. In a numerical experiment the algorithm is shown to be comparable to the power method and the related Z-learning algorithm in terms of convergence speed. It may be used as the basis of a reinforcement learning style algorithm for Markov decision problems.

1. INTRODUCTION

In reinforcement learning [3, 13] we are interested in making optimal decisions in an uncertain environment.

Consider the setting where we are condemned to reside in a certain finite environment for an indefinite amount of time. Whenever we make a move in the environment from one state to another state, we incur a certain cost, depending on the transition. We cannot directly influence this incurred cost, but can hope to make transitions yielding a minimal average cost per transition.

This is an example of a *Markov decision process* [13] and in this paper we present a method that approximately solves this problem in a very general setting. The algorithm we present, *KL-learning* (Algorithm 1), observes randomly made moves (according to some Markov chain transition probabilities) and costs we incur, and finds from this, at no significant computational cost whatsoever, improved transition probabilities for the Markov chain.

This is in contrast to some other well known reinforcement learning algorithms, in which at every iteration an optimization over possible actions is necessary (e.g. Q-learning, [15]) or in which an optimization step is necessary to compute optimal actions (e.g. TD-learning, [12]).

The background for this method is the setting of the *Kullback-Leibler (KL) control problem*, introduced in [14]. A KL-control problem is a Markov decision process in which the control costs are proportional to a *Kullback-Leibler divergence* or *relative entropy*. In [14] also a reinforcement style learning algorithm (Z-learning) was presented, which operates under the assumption of there being an absorbing state in which no further costs are incurred. This assumption is not made in our algorithm; instead we assume ergodicity of the underlying Markov chain. Arguably this yields a more general setting, in which a hard reset of the algorithm is never necesssary. KL control problems may also be solved using techniques from graphical model inference [4].

As a preliminary, we introduce the KL-control setting in Section 2. In Section 3 the KL-learning algorithm is presented and motivated on a heuristic level. We then describe the ODE method [1, 7, 8, 10] in Section 4 along with an application to a stochastic gradient algorithm and Z-learning [14] as illustrative examples. We then apply the ODE method to KL-learning in Section 5. A numerical example is provided in Section 6 after which a short discussion follows in Section 7.

2. KULLBACK-LEIBLER CONTROL PROBLEMS

In this section we introduce the particular form of Markov decision process which have a particularly convenient solution. We will refer to these problems as Kullback-Leibler control problems. For a more detailed introduction, see [14].

Let $t = 0, 1, 2, \ldots$ denote time. Consider a Markov chain $(X_t)_{t=0}^{\infty}$ on a finite state space $S = \{1, \ldots, n\}$ with transition probabilities $q = [q(j|i)]$ which we call the *uncontrolled dynamics*. We will make no distinction between the notation q_{ij} and $q(j|i)$, where $q(j|i) = q_{ij}$ denotes the probability of jumping from state i to state j; the notation q_{ij} will me more convenient when working with matrices.

Suppose for every jump of the Markov chain from state i to state j in S a *transition dependent cost* $c(j|i)$ is incurred. Sometimes we will use the notation $c(i)$ to denote costs depending only state, i.e. $c(j|i) = c(i)$ for all $i, j = 1, \ldots, n$. A state i is called *absorbing* if $q(i|i) = 1$ and $c(i) = 0$.

We wish to change the transition probabilities in such a way as to minimize for example the total incurred cost (assuming there exist absorbing states where no further costs are incurred) or the average cost per stage. For deviating from the transition probabilities control costs are incurred equal to

$$\frac{1}{\beta}\mathrm{KL}(p(X_{t+1}|X_t)||q(X_{t+1}|X_t)) = \frac{1}{\beta}\sum_{j=1}^{n} p(X_{t+1} = j|X_t)\ln\left(\frac{p(X_{t+1} = j|X_t)}{q(X_{t+1} = j|X_t)}\right)$$

at every time step, in addition to the cost per transition $c(X_t|X_{t-1})$, where $\beta > 0$ is a weighing factor, indicating the relative importance of the control costs.

To put this problem in the usual form of a discrete time stochastic optimal control problem, we write $p_{ij} = \exp(u_j(i))q_{ij}$. This guarantees positive probabilities and absolute continuity of the controlled dynamics with respect to the uncontrolled dynamics. In the case of an infinite horizon problem and minimization of a total expected cost problem, the corresponding Bellman equation for the value function Φ is

$$\Phi(i) = \min_{(u_1,\ldots,u_n)\in\mathbb{R}^n}\left\{\sum_{j=1}^{n} c(j|i) + q(j|i)\exp(u_j)(u_j/\beta + \Phi(j))\right\},$$

where the minimization is over all $u_1, \ldots u_n$ such that $\sum_{j=1}^{n}\exp(u_j)q(j|i) = 1$. If there are no absorbing states, the total cost will always be infinite and the expression above has no meaning. We may then instead aim to minimize the expected average cost. For an average cost problem, the Bellman equation for the value function Φ is

$$\rho + \Phi(i) = \min_{(u_1,\ldots,u_n)\in\mathbb{R}^n}\left\{\sum_{j=1}^{n} c(j|i) + q(j|i)\exp(u_j)(u_j/\beta + \Phi(j))\right\}, \tag{1}$$

where again the minimization is over all $u_1, \ldots u_n$ such that $\sum_{j=1}^{n}\exp(u_j)q_{ij} = 1$, and where ρ is the optimal average cost. In the average cost case we restrict the possible solutions by requiring that

$$\sum_{i=1}^{n}\exp(-\beta\Phi(i)) = 1; \tag{2}$$

otherwise any addition by a scalar would result in another possible value function. The reason for the particular form of this restriction will become clear later.

Note that in case the total expected cost problem has a finite value function, the solution of the average cost problem (1) would have a solution with $\rho = 0$. This shows that in a sense the average cost problem is more general, since then (1) remains valid for the total expected cost problem. Therefore we will henceforth only consider the average cost problem case.

So far the derivations have been standard; see [2] for more information on dynamic programming and the Bellman equation.

It is remarkable that a straightforward computation using Lagrange multipliers, as in [14], yields that the optimal $u_j(i)$ and value function Φ solving (1) are given by the simple expressions

$$u_j^*(i) = \ln(z_j^*/\lambda^* z_i^*) - \beta c(j|i)), \qquad \Phi(i) = -\frac{1}{\beta}\ln(z_i^*), \tag{3}$$

with $z^* \in \mathbb{R}^n$ given implicitly by

$$\lambda^* z_i^* = \sum_{j=1}^n \exp(-\beta c(j|i)) q(j|i) z_j^*,$$

which may be written as $\lambda^* z^* = Hz^*$, with

$$H = [h_{ij}] \quad \text{with entries} \quad h_{ij} = \exp(-\beta c(j|i)) q(j|i) \tag{4}$$

and where $\lambda^* = \exp(-\beta\rho^*)$. This z^* should be normalized in such a way that the value function agrees with the value 0 in the absorbing states for a total expected cost problem, or with the normalization (2) in the average cost case, which is chosen in such a way that it corresponds to $||z^*||_1 = \sum_{i=1}^n z_i^* = 1$. The optimal transition probabilities simplify to

$$p(j|i)^* = q(j|i)\exp(-\beta c(j|i))\frac{z_j^*}{\lambda^* z_i^*}.$$

According to Perron-Frobenius theory of non-negative matrices (see [5]), if the uncontrolled Markov chain q is irreducible then there exists, by Observation 2.1 below, a simple eigenvalue λ^* of H equal to the spectral radius $\rho(H)$, with an eigenvector z^* which has only positive entries. Since λ^* is a simple eigenvalue, z^* is unique op to multiplication by a positive scalar. These λ^* and z^* (with z^* normalized as above) are called the *Perron-Frobenius* eigenvalue and eigenvector, respectively. The optimal average cost is given by $\rho^* = -\frac{1}{\beta}\ln\lambda^*$. In case of a total expected cost problem, where $\rho^* = 0$, it follows that $\lambda^* = 1$, which may also be shown directly by analysis of the matrix H.

Recall that a nonnegative matrix A is called *irreducible* if for every pair $i, j \in S$, there exists an $m \in \mathbb{N}$ such that $(A^m)_{ij} > 0$. In particular, a Markov chain p is called irreducible if the above property holds for its transition matrix.

2.1. **Observation.** Suppose the finite Markov chain q on $S = \{1,\dots,n\}$ with transition probabilities $q(j|i)$ is irreducible. Then H as given by (4) is irreducible. In particular, there exists a unique (modulo scalar multiples) positive eigenvector $z^* \in \mathbb{R}^n$ of H such that $Hz^* = \lambda^* z^*$, where $\lambda^* = \rho(H) = \sup_{\mu\in\sigma(H)}|\mu|$, the *spectral radius* of H.

Proof. Let $\gamma = \min_{i\in S} e^{-\beta c(j|i)}$. Let $i, j \in S$ and pick $m \in \mathbb{N}$ such that $(q^m)_{ij} > 0$. Then

$$(H^m)_{ij} = \sum_{k_1=1}^n \dots \sum_{k_{m-1}=1}^n H_{ik_1} H_{k_1k_2} \dots H_{k_{m-1},j} \geq \gamma^m \sum_{k_1=1}^n \dots \sum_{k_{m-1}=1}^n q_{ik_1} q_{k_1k_2} \dots q_{k_{m-1},j} > 0.$$

The existence and uniqueness of the eigenvalue and corresponding eigenvector is then an immediate corollary of the Perron-Frobenius theorem [5, Theorem 8.4.4]. □

Recall that a Markov chain $[p_{ij}]$ is said to satisfy detailed balance if there exists a probability distribution (p_i) such that $p_i p_{ij} = p_j p_{ji}$ for all i, j. In this case (p_i) is an invariant probability distribution for the Markov chain.

2.2. **Proposition.** Suppose the uncontrolled dynamics q satisfy detailed balance (with respect to the invariant probability distribution given by (q_i)).

(a) If the transition costs are actually state costs, i.e. $c(j|i) = c(i)$ for $i, j = 1,\dots,n$, then the optimal controlled dynamics satisfy detailed balance with invariant probability distribution given by

$$p_i \propto q_i \exp(\beta c(i))(z_i^*)^2, \quad i = 1,\dots,n.$$

(b) If the transition costs are symmetric, i.e. $c(j|i) = c(i|j)$ for $i, j = 1, \dots, n$, then the optimal controlled dynamics satisfy detailed balance with invariant probability distribution give by

$$p_i \propto q_i (z_i^*)^2, \quad i = 1, \dots, n.$$

Proof. We will prove (a), the proof of (b) is analogous.

Using that $p_{ij} = \exp(u_j^*(i)) q_{ij}$ with $u_j^*(i)$ given by (3), we verify that $p_i p_{ij} = p_j p_{ji}$ for all i, j. Indeed,

$$\begin{aligned} p_i p_{ij} &= q_i \exp(\beta c(i)) (z_i^*)^2 q_{ij} z_j^* / z_i^* \exp(-\beta c(i)) / Z = q_i q_{ij} z_i^* z_j^* / Z \\ &= q_j q_{ji} z_i^* z_j^* / Z = q_j \exp(\beta c(j)) (z_j^*)^2 q_{ji} z_i^* / z_j^* \exp(-\beta c(j) / Z = p_j p_{ji}, \end{aligned}$$

where $Z = \sum_{k=1}^n q_k \exp(\beta c(k)) (z_k^*)^2$ is a normalization constant. □

2.3. **Example: solution in case of trivial detailed balance.** If we take as uncontrolled dynamics $q_{ij} = q_j$, where q_j is a probability distribution on $\{1, \dots, n\}$, then $H_{ij} = \exp(-\beta c(i)) q_j$ is of rank one and has non-zero eigenvalue $\lambda^* = \sum_{j=1}^n q_j \exp(-\beta c(j))$ with eigenvector z^* given by $z_i^* = \exp(-\beta c(i))$. The optimal transition probabilities are given by

$$p_{ij} = q_j \exp(-\beta c(j)) / \sum_{k=1}^n q_k \exp(-\beta c(k)),$$

which again are independent of i. Therefore the Markov chain given by the controlled dynamics has invariant probability distribution $[p_j] = [p_{ij}]$. The optimal average cost is given by

$$\rho^* = -\frac{1}{\beta} \ln \lambda^* = -\frac{1}{\beta} \ln \sum_{k=1}^n q_k \exp(-\beta c(k)).$$

◇

3. KL-LEARNING

As explained in the previous section, a Kullback-Leibler control problem may be solved by finding the Perron-Frobenius eigenvalue λ^* and eigenvector z^* of the matrix H given by (4).

A straightforward way to find λ^* and z^* is using the *power method*, i.e. by performing the iteration

$$z_{k+1} = \frac{H z_k}{||H z_k||}. \tag{5}$$

This assumes that we have access to the full matrix H. Our goal is to relax this assumption, and to find z by iteratively stepping through states of the Markov chain using the uncontrolled dynamics q, using only the observations of the cost $c(j|i)$ when we make a transition from state i to state j..

In [14] a *stochastic approximation algorithm* (see [1, 3, 7, 8]), referred to as Z-learning, is introduced for the case $\lambda^* = 1$. We will extend this method here to the case where λ^* is a priori unknown.

In this section we will denote vectors by bold letters, e.g. v. Components of this vector will be denoted as $v(i)$ or v_i. The notation $\mathbb{1}$ is used for the column vector containing only ones. A vector $v \in \mathbb{R}^n$ is said to be nonnegative ($v \geq 0$) if $v(i) \geq 0$ for all $i = 1, \dots, n$ and positive ($v > 0$) if $v(i) > 0$ for all $i = 1, \dots, n$.

The algorithm we will consider is Algorithm 1. The parameter $M \in \mathbb{N}$ denotes the number of iterations of the algorithm, and $\gamma > 0$ indicates the stepsize. We assume that the Markov transition probabilities $q(\cdot|\cdot)$ are irreducible and aperiodic, and hence ergodic.

At every iteration, we make a random jump to a new state. Based on our observation of the incurred cost at the previous step, and current values of λ and two components of z, a number Δ is computed that says how much z and λ should be changed. The value of λ is always equal to $\sum_{i=1}^n z_i = ||z_i||_1$. Note that every step of the iteration consists of only simple algebraic operations and hence has time complexity $\mathcal{O}(1)$. In particular, no optimization is needed, as opposed to e.g. Q-learning [15].

Algorithm 1 KL-learning

$z \leftarrow \frac{1}{n}\mathbb{1}$, $\lambda \leftarrow 1$, $x \leftarrow$ any state in S
for $k = 1$ **to** M **do**
  $y \leftarrow$ independent draw from $q(\cdot|x)$
  $\Delta \leftarrow \exp(-\beta c(y|x))z(y)/\lambda - z(x)$
  $z(x) \leftarrow z(x) + \gamma\Delta$
  $\lambda \leftarrow \lambda + \gamma\Delta$
  $x \leftarrow y$
end for

A theoretical analysis of (a slightly modified version of) this algorithm will be performed in Section 5. The results of that section are summarized in Theorem 5.2. First we provide some intuition.

3.1. **Heuristic motivation.** Suppose at time m we are in state i. The expected value of Δ is

$$\sum_{j=1}^{n} q_{ij}\left(\exp(-\beta c_{ij})z(j)/\lambda - z(i)\right) = (Hz)_i/\lambda - z_i.$$

Since $\lambda = ||z||_1$, the update to z may be interpreted as

$$z_{\text{new}} = z + \gamma\left[(Hz)(i)/\lambda - z(i)\right] = (1-\gamma)z(i) + \gamma(Hz)(i)/||z||_1,$$

a convex combination of the old value of $z(i)$ and the value $z(i)$ would obtain after an iteration of the power method described above. The normalization is however based on the previous value of z but this does not affect the convergence of the power method.

The frequency of updates to the i-th component of z depends, on the long run, on the equilibrium distribution (q_i) of the underlying Markov chain. This will be a major concern in the convergence analysis of the algorithm. It will turn out that the convergence of the algorithm will depend on the stability properties of a certain matrix, A say. If we wish the algorithm to converge for a certain invariant distribution, this corresponds to the matrix DA being stable, where D is a diagonal matrix with the invariant distribution on the diagonal. This will be made clear in Section 5.

4. Analysis of stochastic approximation algorithms through the ODE method

In this section a general and powerful method for analyzing the behaviour and possible convergence of stochastic approximation algorithms is described. It will be applied to Algorithm 1 in Section 5. This method, called the *ODE method*[1], was first introduced by Ljung [10] and developed significantly by Kushner and coworkers [7, 8]. Accounts that are well suited for computer scientists and engineers may be found in [1, 3].

The theory is illustrated by applying it to some stochastic algorithms. The new contribution of this section to the existing theory is the necessity of *diagonal stability* for the convergence of certain stochastic algorithms, as discussed in Section 4.9.

4.1. **Outline of the ODE method.** The idea of the ODE method is to establish a relation between the trajectories of a stochastic algorithm with decreasing stepsize, and the trajectories of an ordinary differential equation. If all trajectories of the ODE converge to a certain equilibrium point, the same can then be said about trajectories of the stochastic algorithm. This is made more precise in the following theorem, which is a special case of [8, Theorem 6.6.1] tailored to our needs.

4.2. **Hypotheses.** Consider the general stochastic approximation algorithm given by Algorithm 2, assuming the following assumptions and notation:

(i) Let $\gamma_1, \gamma_2, \ldots$ be a sequence of step sizes, satisfying $\sum_{k=1}^{\infty}\gamma_k = \infty$ and $\sum_{k=1}^{\infty}\gamma_k^2 < \infty$;

[1] Here ODE is an abbreviation for *ordinary differential equation*.

Algorithm 2 General stochastic approximation algorithm for theoretical analysis

$x_0 \leftarrow$ any state in S
$\theta_0 \leftarrow$ any initial vector in $\mathbb{R}^n$
for $k = 1$ **to** ∞ **do**
  $x_k \leftarrow$ independent fraw from $q(\cdot|x_{k-1})$
  $\theta_k \leftarrow \theta_{k-1} + \gamma_k f(\theta_{k-1}, x_{k-1}, x_k)$
end for

(ii) Let $q(\cdot|\cdot)$ be irreducible aperiodic Markov transition probabilities on a finite state space S with invariant probabilities q_i, $i \in S$;
(iii) Suppose that $\{\theta_k : k \in \mathbb{N}\} \subset K$ with probability one, where K is some compact (i.e. closed and bounded) subset of $\mathbb{R}^n$;
(iv) Suppose $(\theta, x, y) \mapsto f(\theta, x, y) : K \times S \times S \to \mathbb{R}^n$ is continuous in θ for every $x, y \in S$.
(v) Define $\overline{g} : K \to \mathbb{R}^n$ by

$$\overline{g}(\theta) := \sum_{i \in S} \sum_{j \in S} q_i q_{ij} f(\theta, i, j). \tag{6}$$

(vi) Define $t_0 := 0$ and $t_k := \sum_{i=1}^{k} \gamma_i$ for $k \in \mathbb{N}$. Denote, for all $t \geq 0$ and $k = 0, 1, 2, \ldots$, $\theta^k(t) := \theta_p$ for the unique p such that $t_p \leq t + t_k < t_{p+1}$, and $\theta^k(t) := 0$ if $t + t_k < t_0$.

These assumptions are sufficient for our purposes. The sequence γ denotes the *stepsize* or *gain*. The conditions under (i) on γ are standard conditions to guarantee that the gain gradually decreases, but not too quickly, in which case the algorithm would stop making significant updates before being able to converge.

In [8] more general classes of algorithms and assumptions are considered.

4.3. **Theorem (convergence of stochastic algorithms with state dependent updates).** Suppose Assumptions 4.2 hold. Then, with full probability,

(i) Every sequence in the collection of functions $\{\theta^k : k \in \mathbb{N}\}$ (as defined under Assumption 4.2 (vi)) admits a convergent subsequence with a continuous limit;[2]
(ii) Let θ denote the limit of some converging subsequence in $\{\theta^k : k \in \mathbb{N}\}$ (which always exists by (i)). Then θ satisfies the ODE

$$\dot{\theta} = \overline{g}(\theta) \tag{7}$$

(iii) If a set $A \subset \mathbb{R}^n$ is globally asymptotically stable with respect to the ODE (7), then $\theta_k \to A$, i.e. $\min_{x \in A} |\theta_k - x| \to 0$.

Outline of proof. The proof consists of a verification of the conditions of [8, Theorem 6.6.1]. One key ingredient for this verification is Lemma 4.4 below, which says that convergence of the pair (x_{k-1}, x_k) to its equilibrium distribution $(q_i q_{ij})_{i,j \in S}$ happens exponentially fast. ⋄

Recall the *total variation distance* [9, Section 4.1] of two probability measures μ_1, μ_2 on a discrete space S,

$$||\mu_1 - \mu_2||_{\mathrm{TV}} := \sup_{A \subset S} |\mu_1(A) - \mu_2(A)| = \tfrac{1}{2} \sum_{i \in S} |\mu_1(i) - \mu_2(i)|.$$

4.4. **Lemma (Markov chain convergence to invariant distribution).** Let $q(j|i)$, $i, j \in S$, denote the transition probabilities of an irreducible, aperiodic Markov chain X on a finite state space S with invariant distribution q_i, $i \in S$. Let μ_k^x be the probability measure on $S \times S$ denoting the distribution of (X_{k-1}, X_k) given $X_0 = x$. Let $\overline{\mu}$ denote the probability measure on $S \times S$ given by $\overline{\mu}(i, j) = q_i q(j|i)$.

[2] Here by convergence we mean uniform convergence on bounded intervals.

Then there exist constants $\alpha \in (0,1)$ and $C > 0$ such that

$$\max_{x\in S} ||\mu_k^x - \overline{\mu}||_{\mathrm{TV}} \le C\alpha^k, \quad \text{for all} \quad k \in \mathbb{N}. \tag{8}$$

Proof: Let ν_k^x denote the probability measure on S denoting the distribution of X_k given initial condition $X_0 = x$. By [9, Theorem 4.9], there exist constants $\widetilde{C} > 0$ and $\alpha \in (0,1)$ such that

$$\max_{x\in S} ||\nu_{k-1}^x - q||_{\mathrm{TV}} \le \widetilde{C}\alpha^{k-1} \quad \text{for } k \in \mathbb{N}.$$

Therefore

$$\begin{aligned}
\max_{x\in S} ||\mu_k^x - \overline{\mu}||_{\mathrm{TV}} &= \max_{x\in S} \tfrac{1}{2} \sum_{i\in S}\sum_{j\in S} |\mathbb{P}(X_{k-1} = i, X_k = j | X_0 = x) - q_i\, q(j|i)| \\
&= \max_{x\in S} \tfrac{1}{2} \sum_{i\in S} q(j|i) \sum_{j\in S} |\mathbb{P}(X_{k-1} = i | X_0 = x) - q_i| \\
&= \max_{x\in S} \tfrac{1}{2} \sum_{i\in S} |\mathbb{P}(X_{k-1} = i | X_0 = x) - q_i| = \max_{x\in S} ||\nu_{k-1}^x - q||_{\mathrm{TV}} \le \widetilde{C}\alpha^{k-1}.
\end{aligned}$$

By letting $C = \widetilde{C}/\alpha$ we find that (8) holds. □

4.5. **Remark.** Note that a boundedness assumption is made in Theorem 4.3. In practice, this is not an unreasonable assumption, since float sizes are bounded in many programming languages. The boundedness may be enforced by a projection step in the algorithm, leading to a slightly more complex formulation of Theorem 4.3. In particular, the resulting ODE becomes a projected ODE. See [8, Section 4.3].

4.6. **Example: A stochastic gradient algorithm.** Suppose we wish to minimize a function $h : \mathbb{R}^n \to \mathbb{R}$ with bounded first derivatives but that we do not have full access to its gradient $g_j = \frac{\partial h}{\partial \theta^j}$. Instead, the observations we make are determined by an underlying Markov chain (x_k) on the state space $\{1,\dots,n\}$ with aperiodic, irreducible transition probabilities q_{ij}. In case a jump is made to x_k, we observe $\frac{\partial h}{\partial \theta^{(x_k)}}(\theta)$ for some $\theta \in \mathbb{R}^n$. Is there a stochastic approximation algorithm that can minimize h under these restrictive conditions?

Consider Algorithm 3. We use e_i to denote the unit vector in direction i.

Algorithm 3 Stochastic gradient algorithm

$x_0 \leftarrow$ any state in S
$\theta_0 \leftarrow$ any initial vector in $\mathbb{R}^n$
for $k = 1$ **to** ∞ **do**
  $x_k \leftarrow$ independent fraw from $q(\cdot|x_{k-1})$
  $\theta_k \leftarrow \theta_{k-1} - \gamma_k \frac{\partial h(\theta_{k-1})}{\partial \theta^{(x_k)}} e_{(x_{k-1})}$
end for

Since ∇h is bounded, the trajectories of this algorithm are restricted to the bounded set

$$K = \{\theta \in \mathbb{R}^n : |\theta| \le \max(|\theta_0|, |\nabla h|)\}$$

with probability one. The corresponding ODE (in the sense of Theorem 4.3) is (7) with

$$\overline{g}^i(\theta) = -q_i \sum_{j=1}^n q_{ij} \frac{\partial h(\theta)}{\partial \theta^j}. \tag{9}$$

Let $R = [r_{ij}]$ be the matrix defined by $r_{ij} = q_i q_{ij}$, $i, j = 1,\dots,n$. We may then write

$$\dot{g}(\theta) = -R\nabla h(\theta). \tag{10}$$

Clearly the minimum θ^* of h, where $\nabla h(\theta^*) = 0$, gives an equilibrium point of this ODE. It is not immediately clear whether this is the only equilibrium point.

We will now make the following assumptions:

(11) The matrix R given by the entries $r_{ij} := q_i q_{ij}$ is symmetric, positive definite.

(12) The function h is twice differentiable and strictly convex.

Sufficient conditions for (11) to hold are the following:

(i) The Markov chain given by q_{ij} satisfies *detailed balance*, i.e. $q_i q_{ij} = q_j q_{ji}$ for $i, j = 1, \dots, n$;
(ii) The Markov chain given by q_{ij} is *strictly lazy*, i.e. $q_{ii} > \frac{1}{2}$ for $i = 1, \dots, n$.

Indeed, if these conditions are satisfied, then R is symmetric by the detailed balance condition. Since the Markov chain is lazy, R is strictly row diagonally dominant, so that all its eigenvalues are positive.

Under Assumption (11), define a Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$ by

$$V(\theta) = \tfrac{1}{2}\langle \nabla h(\theta), R\nabla h(\theta)\rangle,$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product on $\mathbb{R}^n$.

Write $H(\theta) = \left(\frac{\partial^2 h}{\partial\theta^i \theta^j}\right)_{i,j=1,\dots,n}$ to denote the Hessian matrix of h at θ, and note that, since h is strictly convex, the matrix $H(\theta)$ is positive definite for all $\theta \in \mathbb{R}^n$. Then if $\theta(t)$ satisfies (10),

$$\frac{d}{dt}V(\theta(t)) = \langle \nabla h(\theta(t)), RH(\theta(t))\dot\theta(t)\rangle = -\langle \nabla h(\theta(t)), RH(\theta(t))R\nabla h(\theta(t))\rangle \le 0,$$

with strict inequality if $\nabla h(\theta) \neq 0$. This shows that the ODE (10) is globally asymptotically stable with unique equilibrium θ^* satisfying $\nabla h(\theta^*) = 0$. By Theorem 4.3 (iii) therefore Algorithm 3 converges almost surely to θ^*.

In this case we are in some sense lucky to be able to find a Lyapunov function to establish global stability of the ODE. In the case of Algorithm 1 (KL-learning) we have not yet found a global Lyapunov function and so far can only achieve local stability around the equilibrium in certain cases. For illustrative purposes, we now also perform such a local analysis to the current example.

It is immediately clear that under assumption (11), the only equilibrium of the ODE (10) satisfies $\nabla h(\theta) = 0$. It remains to establish the stability of the ODE around that equilibrium point. The linearized version of the ODE around the equilibrium is

$$\dot\theta(t) = -RH(\theta^*)\theta. \tag{13}$$

We therefore need to determine the spectrum of the matrix $RH(\theta^*)$. Indeed $RH(\theta^*)R + RH(\theta^*)R$ is positive definite, so that by Lyapunov's theorem [6, Theorem 2.2.1] $RH(\theta^*)$ has only eigenvalues in the open right halfplane. We may conclude from this local analysis, by the Hartman-Grobman theorem [11, Section 2.8], that the equilibrium θ^* is locally asymptotically stable.

4.7. **Remark.** Under the assumption that we can only observe $\frac{\partial h}{\partial\theta^j}$ if we jump to state j, a simpler algorithm would consist of the update rule $\theta_k \leftarrow \theta_{k-1} - \gamma_k \frac{\partial h(\theta_{k-1})}{\partial\theta^{(x_k)}} e_k$, i.e. to update the (x_k)-th component of θ instead of the (x_{k-1})-th component. In this case a Lyapunov function would be given by $V(\theta) = \sum_{i=1}^n q_i \left(\frac{\partial h(\theta)}{\partial\theta^i}\right)^2$ and Assumption (11) would not be required. However, the analysis of Algorithm 3 has more in common with the upcoming analysis of Algorithm 1 (KL-learning), because in that algorithm the updates also depend upon the previous and current state of the Markov chain.

4.8. **Example: Z-learning.** In [14], the Z-learning algorithm is presented as a way to solve the eigenvector problem $Hz^* = z^*$, where $H = [h_{ij}]$ is a nonnegative irreducible matrix with spectral radius $\rho(H) = 1$ of the form $h_{ij} = \exp(-\beta c_{ij}) q_{ij}$ as in Section 2, with $[q_{ij}]$ the transition probabilities of some irreducible Markov chain on $S = \{1, \dots, n\}$. This problem is an important special case of the problem we address in this paper, namely solving $Hz^* = \lambda^* z^*$ with unknown spectral radius $\rho(H) = \lambda^*$.

The Z-learning algorithm is given by Algorithm 4.

Algorithm 4 Z-learning

$z_0 \leftarrow \mathbb{1}$, $x_0 \leftarrow$ any state in S
for $k = 1$ **to** ∞ **do**
  $x_k \leftarrow$ independent draw from $q(\cdot|x_{k-1})$
  $z_k \leftarrow z_{k-1} + \gamma_k \left(\exp(-\beta c(x_k|x_{k-1}))z_{k-1}(x_k) - z_{k-1}(x_{k-1})\right) e_{x_{k-1}}$
end for

The corresponding ODE (in the sense of Theorem 4.3 is given by

$$\dot{z}(t) = -D(I-H)z(t), \tag{14}$$

where D is a diagonal matrix given by $d_{ii} = q_i$, where (q_i) denotes the invariant probability distribution of the Markov chain given by $[q_{ij}]$.

It is immediate from the Perron-Frobenius theorem that the eigenvalues of the matrix $I-H$ are strictly contained in the closed right halfplane, with a one-dimensional eigenspace corresponding to the zero eigenvalue and all other eigenvalues having strictly positive real part. This still holds for a multiplication of $I-H$ by an arbitrary diagonal matrix with positive diagonal entries, but this is less immediate (see e.g. [6, Exercise 2.5.2] for the nonsingular case). Therefore the linear subspace spanned by z^* is globally attracting, and the Z-learning algorithm converges to this subspace by Theorem 4.3 (iii).

4.9. **D-stability as a necessary condition for convergence of stochastic approximation algorithms.** In the previous example, the positive stability of $I-H$ carried over to a multiplication by a positive diagonal matrix, $D(I-H)$, irrespective of the kind of diagonal matrix D. This kind of stability (invariant under left- (or right-)multiplication by an arbitrary positive diagonal matrix) is called D-stability in the literature (see e.g. [6, Section 2.5]), and the major difficulty with establishing local stability of the KL-learning algorithm consists of showing D-stability for the corresponding linearized ODE.

5. Theoretical analysis of KL-learning

The KL-learning algorithm (Algorithm 1) works well in practice, but for a rigorous theoretical analysis of its behaviour we need to make a few modifications, as given by Algorithm 5.

The modifications of Algorithm 5 with respect to Algorithm 1 are:

(i) The values of z, λ and Δ are indexed by the time parameter k to keep track of all values;
(ii) Instead of a single step size $\gamma > 0$ and a finite time horizon $M \in \mathbb{N}$ we consider an infinite time horizon and a decreasing sequence of stepsizes (γ_k);
(iii) At every iteration, if necessary, a projection is performed (in the computation of Δ_k) to ensure that $\lambda_k \geq \lambda_{\min} := \min_{i,j\in\{1,\dots,n\}} \exp(-\beta c(j|i))/2$.

The modification (i) is purely a notational matter. Modification (ii) is standard in the analysis of stochastic approximation algorithms. If we would keep the stepsize constant the theoretical analysis would be harder. The practical effect of keeping the stepsize fixed is that the values of (z_k, λ_k) will oscillate around the theoretical solution (z^*, λ^*) with a bandwith depending on γ. Modification (iii) has minimal practical effect; we have not seen cases in which the projection step was actually made. The theoretical solution λ^* satisfies $\lambda^* \geq 2\lambda_{\min}$ by theory on nonnegative matrices [5, Corollary 8.1.19]. The constant 2 is arbitrary, chosen to ensure that λ^* lies well above $\lambda_{\min}$. By Lemma 5.3, λ_k is bounded from above, so there is no need to prevent λ_k from growing large.

In this section we will write $\mathbb{R}^n_+ := \{x \in \mathbb{R}^n : x_i > 0 \text{ for } i = 1,\dots,n\}$.

In the discussion below we will only refer to Algorithm 5.

The initial value for λ_0 is moreless arbitrary, but it is important that $||z_0||_1 = \lambda_0$ and $\lambda_0 \in K$ with K given by Lemma 5.3. We impose the following conditions.

Algorithm 5 KL-learning, notation for analysis

$\lambda_0 \leftarrow 2\lambda_{\min} = \min_{i,j=1,\dots,n} \exp(-\beta c(j|i))$
$z_0(i) \leftarrow \frac{1}{n}\lambda_0$, for all $i = 1,\dots,n$,
$x_0 \leftarrow$ any state in S
for $k = 1$ **to** ∞ **do**
  $x_k \leftarrow$ independent draw from $q(\cdot|x_{k-1})$
  $\Delta_k \leftarrow \max\{\exp(-\beta c(x_k|x_{k-1}))z_{k-1}(x_k)/\lambda_{k-1} - z_{k-1}(x_{k-1}), (\lambda_{\min} - \lambda_{k-1})/\gamma_k\}$
  $z_k \leftarrow z_{k-1}$
  $z_k(x_{k-1}) \leftarrow z_{k-1}(x_{k-1}) + \gamma_k\Delta_k$
  $\lambda_k \leftarrow \lambda_{k-1} + \gamma_k\Delta_k$
end for

5.1. **Hypothesis.**

(i) Let $(\gamma_k)_{k\in\mathbb{N}}$ be a sequence of nonnegative real numbers such that $\sum_{m=1}^{\infty}\gamma_k = \infty$, $\sum_{m=1}^{\infty}\gamma_k^2 < \infty$.
(ii) Let $q(\cdot|\cdot)$ be irreducible aperiodic Markov transition probabilities on the finite state space $S = \{1,\dots,n\}$ with invariant probabilities q_i, $i \in S$;
(iii) Let $c \in \mathbb{R}^{n\times n}$;
(iv) Let the matrix $H = [h_{ij}] \in \mathbb{R}^{n\times n}$ be given by $h_{ij} = \exp(-\beta c_{ij})q_{ij}$ and the diagonal matrix $D = [d_i] \in \mathbb{R}^{n\times n}$ by $d_i = q_i$;
(iv) Define continuous time processes $(z^k(t))_{t\geq 0}$ and $(\lambda^k(t))_{t\geq 0}$ for $k = 0,1,\dots$ as in Hypothesis 4.2 (vi).

5.2. **Theorem (Convergence of KL-learning).** Consider Algorithm 5 under the conditions of Hypothesis 5.1. Then

(a) With full probability, for any sequence of processes (z^k,λ^k) there exists a subsequence uniformly on bounded intervals to some continuous functions (z,λ), $z:[0,\infty)\to\mathbb{R}^n_+$ and $\lambda:[0,\infty)\to(0,\infty)$.
(b) The trajectories of Algorithm 5, as well as the limiting functions (z,λ) given by (b), are constrained to a closed, bounded set K given by (17).
(c) Such a limit (z,λ) satisfies the ODE

$$\begin{cases} \dot z(t) = f(z(t),\lambda(t)) + w, \\ \dot\lambda(t) = h(z(t),\lambda(t)) + \mu, \qquad t \geq 0, \end{cases} \tag{15}$$

with $f:\mathbb{R}^n_+\times(0,\infty)\to\mathbb{R}^n$ and $h:\mathbb{R}^n_+\times(0,\infty)\to\mathbb{R}$ given by

$$f(z,\lambda) := D\left(\frac{1}{\lambda}H - I\right)z, \quad h(z,\lambda) := \mathbb{1}^T f(z,\lambda), \tag{16}$$

where

$$D = \mathrm{diag}(q(1),\dots,q(n)),$$

with q the unique invariant probability distribution for the Markov chain with transition probabilities q_{ij}. Here $w \in \mathbb{R}^n_{\geq 0}$ and $\mu \geq 0$ denote the minimum force necessary to keep $(z(t),\lambda(t))$ in K (the continuous time equivalent of the projection step to ensure that $\lambda_k \geq \lambda_{\min}$; see [8, Section 4.3]).
(d) The ODE (15) admits a unique equilibrium (z^*,λ^*) in the interior of K, where $Hz^* = \lambda^* z^*$ and $||z^*||_1 = \lambda^*$.
(e) If any of the conditions of Proposition 5.10 hold, then the equilibrium (z^*,λ^*) as mentioned under (d) is locally asymptotically stable.

Proof. By Lemma 5.3 the trajectories of Algorithm 5 are constrained to the compact set K given by (17). We may apply a variant Theorem 4.3 suitable for projected algorithms (see [8, Theorem 6.6.1] to deduce (a), (b) and (c), where for (c) we may use Lemma 5.4. Results (d) and (e) follow from Propositions 5.5, Remark 5.6, 5.8 and 5.10, where we note that no projection force is necessary in the interior of K. □

5.3. **Lemma (algorithm invariants).** Under Hypothesis 5.1, the trajectories (z_k, λ_k) of Algorithm 5 are contained in the compact set

$$K = \left\{(z,\lambda) \in [0,M]^n \times [\lambda_{\min}, nM] : ||z||_1 = \lambda\right\}, \tag{17}$$

with $M := \max_{i,j=1,\dots,n} \exp(-\beta c_{ij})$;

Proof. Note that, by the maximum operation in the algorithm, $\Delta_k \geq \exp(-\beta c(x_k|x_{k-1}))z_{k-1}(x_k) - z_{k-1}(x_{k-1}))$. The update for z_k therefore satisfies

$$z_k(x_{k-1}) \geq (1-\gamma_k)z_{k-1}(x_{k-1}) + \gamma_k \exp(-\beta c(x_k|x_{k-1}))z_{k-1}(x_k)/\lambda_{k-1} > 0 \tag{18}$$

It follows immediately by induction that $z_k(i) > 0$ for $i = 1,\dots,n$ and $k = 1,\dots,n$. The update-rule $\lambda_k \leftarrow \lambda_{k-1} + \gamma_k \Delta_k$ for λ_k ensures that $\lambda_k = ||z_k||_1 \geq \lambda_{\min}$ for all $k = 1,\dots,n$. We will show by induction that $z_k(i) \leq M$ for all $k \in \mathbb{N}$ and $i = 1,\dots,n$. Recall that $z_0(i) \leq \frac{M}{n} \leq M$ for $i = 1,\dots,n$. Suppose $z_{k-1}(i) \leq M$ for all i, and some $k \in \mathbb{N}$. If no projection occurs

$$z_k(x_{k-1}) \leq (1-\gamma_k)z_{k-1}(x_{k-1}) + \gamma_k \max_{i,j=1,\dots,n} \exp(-\beta c_{ij}) \leq M,$$

where we used that $z_{k-1}(i) \leq \lambda_{k-1}$ for all i. If projection does occur,

$$z_k(x_{k-1}) \leq \lambda_k = \lambda_{k-1} + \gamma_k(\lambda_{\min} - \lambda_{k-1})/\gamma_k = \lambda_{\min} < M.$$

□

5.4. **Lemma.** The function $\overline{g}$ corresponding to Algorithm 5 in the sense of Theorem 4.3 is given by

$$\overline{g}(z,\lambda) = \begin{bmatrix} f(z,\lambda) \\ h(z,\lambda) \end{bmatrix},$$

where f and h are given by (16).

Proof. A straightforward computation. □

5.5. **Proposition.** Suppose D is a diagonal matrix with positive entries and H a nonnegative matrix. Suppose $f : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ and $h : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ are given by (16).

Consider the ODE (15) with initial values $(z(0), \lambda)$ such that $z(0) > 0$ and $\lambda(0) = ||z(0)||_1$.

The orbits $(z(t), \lambda(t))$ satisfy $z(t) \geq 0$ and $\lambda(t) = ||z(t)||_1 > 0$ for all $t \geq 0$. Furthermore a point $(z,\lambda) \in \mathbb{R}^n \times \mathbb{R}$ is an equilibrium point if and only if $Hz = \lambda z$.

Proof. Suppose that for some $t \geq 0$, we have $z(t) \geq 0$, $\lambda(t) > 0$. If for some component $z(t)(k)$ of $z(t)$ we have $z(t)(k) = 0$, then

$$[f(z(t),\lambda(t))](k) = \left[D\left(\frac{1}{\lambda(t)}H - I\right)z(t)\right](k) = \frac{1}{\lambda(t)}[DHz(t)](k) \geq 0.$$

Also, as $\lambda \downarrow 0$, then

$$h(z,\lambda) = \mathbb{1}^T D\left(\frac{1}{\lambda}H - I\right)z \to \infty,$$

so that λ always remains positive. For $z(t) \geq 0$,

$$\frac{d}{dt}||z(t)||_1 = \sum_{k=1}^n \dot{z}(t)(k) = \mathbb{1}^T \dot{z}(t) = g(z(t),\lambda(t)) = \dot{\lambda}(t).$$

Because $z(0)(k) > 0$ for all $k = 1,\dots,n$ and $\lambda(0) = ||z(0)||_1$, it follows that $z(t) \geq 0$ for all t and $||z(t)|| = \lambda(t)$.

It is straightforward that $Hz = \lambda z$ if and only if (z,λ) is an equilibrium point. □

5.6. **Remark.** In light of the proposition above, we may consider the dynamical system

$$\begin{cases} \dot{z}(t) = D\left(\frac{1}{||z(t)||_1}H - I\right)z(t), \\ z(0) = z_0, \end{cases} \tag{19}$$

with $z_0(k) > 0$ for all $k = 1,\dots,n$, instead of (15), thus reducing the dimensionality from $\mathbb{R}^{n+1}$ to $\mathbb{R}^n$.

5.7. **Definition.** A matrix $A \in \mathbb{R}^{n\times n}$ is called *stable* (or *strictly stable*) if all eigenvalues $\lambda \in \sigma(A)$ satisfy $\operatorname{Re}\lambda \leq 0$ (or $\operatorname{Re}\lambda < 0$, respectively).

5.8. **Proposition.** Suppose H is a nonnegative irreducible matrix and D is a diagonal matrix with positive entries on the diagonal.

Then (19) has a unique equilibrium point z^*. This equilibrium point satisfies

(i) $||z^*||_1 = \lambda^* := \rho(H)$,
(ii) $z^* > 0$, and
(iii) $Hz^* = \lambda^* z^*$.

The equilibrium is locally (asymptotically) stable if and only if the matrix $D(H - \lambda^* I - z^* \mathbb{1}^T)$ is (strictly) stable.

Proof. By Remark 5.6 above, we may apply Proposition 5.5 to conclude that z^* satisfies (iii) for some $\lambda^* > 0$, and $z^* \geq 0$, $z^* \neq 0$. Since H is nonnegative and irreducible, there is, up to scaling by a positive constant, only a single eigenvector with nonnegative components. This is the Perron vector whose eigenvalue satisfies $\lambda^* = \rho(H)$, and which has only positive components, so that (i) and (ii) follow.

The linearization of $z \mapsto D\left(\frac{1}{||z||_1}H - I\right)z$ around z is given by

$$v \mapsto D\left(\frac{1}{||z||_1}H - I\right)v - \frac{1}{||z||_1^2}DHz\mathbb{1}^T v,$$

which reduces to

$$v \mapsto D\left(\frac{1}{\lambda^*}(H - z^*\mathbb{1}^T) - I\right)v$$

for $z = z^*$. Multiplication by λ^* does not affect the stability properties of this matrix, so the stability of the equilibrium z^* is determined by the spectrum of the matrix $D(H - z^*\mathbb{1}^T - \lambda^* I)$. □

The stability of the matrix $D(H - \lambda^* I - z^*\mathbb{1}^T)$ seems to be a non-trivial issue. In Proposition 5.10 below we collect some facts that we have already obtained. For this we need a lemma.

5.9. **Lemma.** Suppose H and D satisfy the conditions of Proposition 5.8. Then the matrix $D(H - \lambda^* I - z^*\mathbb{1}^T)$, with $\lambda^* = \rho(H)$ and z^* the corresponding positive eigenvector, is nonsingular.

Proof. We will omit *-superscripts in this proof, so $z = z^*$ and $\lambda = \lambda^*$. Write $A = H - \lambda I - z\mathbb{1}^T$.

Let w and z denote the left and right Perron-Frobenius eigenvectors of H. Note that $(z, w) > 0$ and $(\mathbb{1}, z) > 0$. Also note that any $\zeta \in \mathbb{R}^n$ may be written as $\zeta = \alpha z + \eta$, with $\eta \perp \mathbb{1}$, by picking $\alpha = (\zeta, \mathbb{1})/(z, \mathbb{1})$ and $\eta = \zeta - \alpha z$. Therefore we may choose a basis of $\mathbb{R}^n$ consisting of the vector $v_1 = z$ and some vectors $v_2, \ldots, v_n$ spanning $\mathbb{1}^\perp$. Let S denote the matrix with columns $v_1, \ldots, v_n$. Then the first column of $S^{-1}(H - \lambda I)S$ consists of zeroes since $(H - \lambda I)z = 0$, and only the first column of $S^{-1}z\mathbb{1}^T S$ is nonzero. So adding $S^{-1}z\mathbb{1}^T S$ to $S^{-1}(H - \lambda I)S$ only increases the range of the resulting matrix. Therefore, $\operatorname{range}(H - \lambda I) \subset \operatorname{range}(A)$.

Since $w^T(H - \lambda I) = 0^T$, we have that w is perpendicular to the range of $H - \lambda I$. But w is not perpendicular to the range of A since $(w, z) > 0$. In other words, the inclusion $\operatorname{range}(H - \lambda I) \subset \operatorname{range}(A)$ is strict, $\operatorname{rank}(A) > \operatorname{rank}(H - \lambda I) = n - 1$, so that $\operatorname{rank}(A) = n$ and $\det(A) \neq 0$. Hence also $\det(DA) \neq 0$. □

5.10. **Proposition.** Suppose H and D satisfy the conditions of Proposition 5.8. The matrix $D(H - \lambda^* I - z^*\mathbb{1}^T)$, with $\lambda^* = \rho(H)$ and z^* the corresponding positive eigenvector, is strictly stable in any of the following cases:

(i) $D = \beta I$ for some $\beta > 0$,
(ii) $\mathbb{1}^T H = \lambda^* \mathbb{1}^T$ (so $\mathbb{1}^T$ is a left Perron vector),
(ii) $D, H \in \mathbb{R}^{2\times 2}$.

Proof. As before, we will omit *-superscripts in this proof, so $z = z^*$ and $\lambda = \lambda^*$.

(i) Suppose v is an eigenvector of $A = H - \lambda I - z\mathbb{1}^T$ with eigenvalue $\mu \neq 0$. Define $w = v + \left(\frac{\mathbb{1}^T v}{\mu}\right) z$. Then

$$(H - \lambda I)\, w = (H - \lambda I)\, v = \left(\mu v + z\mathbb{1}^T v\right) = \mu w,$$

which shows that $\mu \in \sigma(H - \lambda I)$, and since $\mu \neq 0$, it follows that $\mathrm{Re}\,\mu < 0$. So all $\mu \in \sigma(A)$ have $\mathrm{Re}\,\mu < 0$, except possibly the case where $\mu = 0$ but this case is excluded by Lemma 5.9 above.

(ii) Let $B = (H - \lambda I)\mathrm{diag}(z)$. Then B has a positive diagonal and negative off-diagonal entries. Furthermore $B\mathbb{1} = 0$ and $\mathbb{1}^T B = 0$. This shows that B is row diagonally dominant and column diagonally dominant, so that the same holds for $B + B^T$. It follows that $B + B^T$ is positive semidefinite. Note that $B + B^T$ is a singular M-matrix (see [6], Section 2.5.5). Since H is irreducible, also $B + B^T$ is irreducible. Therefore the nullspace of $B + B^T$ is one-dimensional and it is spanned by $\mathbb{1}$. Also $z\mathbb{1}^T\mathrm{diag}(z) = zz^T$ is symmetric positive semidefinite, and $(\mathbb{1}, zz^T\mathbb{1}) > 0$. It follows that $B + B^T + 2zz^T$ is positive definite. Multiplying on both sides by D (a congruence transform) gives that $D(H - \lambda I - z\mathbb{1}^T)\mathrm{diag}(z)D + \mathrm{diag}(z)D(H - \lambda I - z\mathbb{1}^T)^T D$ is symmetric positive definite. By Lyapunov's theorem [6, Theorem 2.2.1], it follows that $D(H - \lambda I - z\mathbb{1}^T)$ is strictly stable.

(iii) By (i) we have that $A = H - \lambda I - z\mathbb{1}^T$ is strictly stable. In 2 dimensions, this is equivalent to $\det(A) > 0$ and $\mathrm{tr}(A) < 0$. This immediately implies that $\det(DA) = \det(D)\det(A) > 0$. We compute

$$\mathrm{tr}(DA) = \mathrm{tr}\begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}\begin{bmatrix} h_{11} - \lambda - z_1 & h_{12} - z_1 \\ h_{21} - z_2 & h_{22} - \lambda - z_2 \end{bmatrix} = d_1(h_{11} - \lambda - z_1) + d_2(h_{22} - \lambda - z_2) < 0,$$

since the diagonal of $H - \lambda I$ has nonpositive entries (which may be seen by [5], Theorem 8.3.2).

□

We think the above proposition can be generalized significantly. In fact, we propose the following conjecture. If the conjecture holds, Theorem 5.2 (e) can be formulated unconditionally.

5.11. **Conjecture.** Suppose H and D satisfy the conditions of Proposition 5.8. Then the matrix $D(H - \lambda^* I - z^*\mathbb{1}^T)$ is strictly stable.

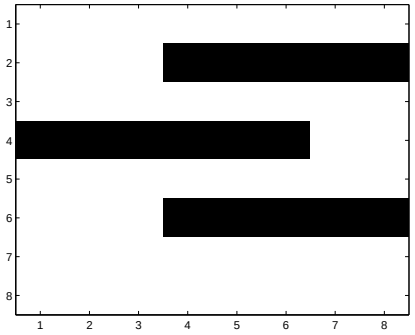
6. Numerical experiment

Consider the example of a gridworld (Figure 1 (a)), where some walls are present in a finite grid. Suppose the uncontrolled dynamics q allow to move through the walls, but walking through a wall is very costly, say a cost of 100 per step through a wall is incurred. Where there is no wall, a cost of 1 per step is incurred. There is a single state, in the bottom right, where no costs are incurred. The uncontrolled dynamics are such that with equal probability we may move left, right, up, down or stay where we are (but it is impossible to move out of the gridworld). The value function for this problem can be seen in Figure 1 (b). In order to be able to compare our algorithm to the original Z-learning algorithm, the cost vector is normalized in such a way that $\lambda^* = 1$, so that Z-learning converges on the given input.

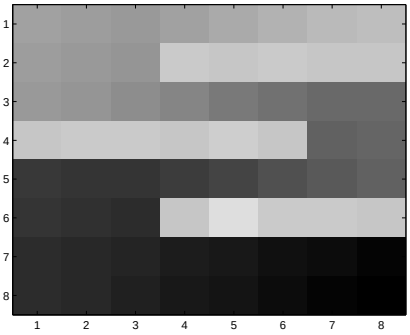
The result of running the stochastic approximation algorithm, with a constant gain of $\gamma = 0.05$ is portrayed in Figure 1 (c), where it is compared to Z-learning (see Section 4.8 and [14]). This result may also be compared to the use of the power method in Figure 1 (d). Here the following version of the power method is used, in order to be able to give a fair comparison with our stochastic method.

$$z_k = z_{k-1} + \gamma_k(Hz_{k-1} - z_{k-1}).$$

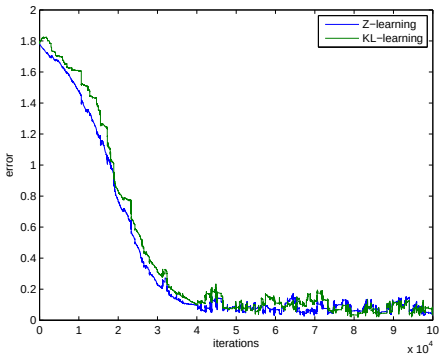
Note that for each iteration, the number of operations is (for sparse H) proportional to the number of non-zero elements in H. In the stochastic method the number of operations per iteration is of order 1. Comparing the graphs in Figure 1 (c) and (d), we see that KL-learning does not disappoint in terms of speed of convergence, with respect to Z-learning as well as the power method.



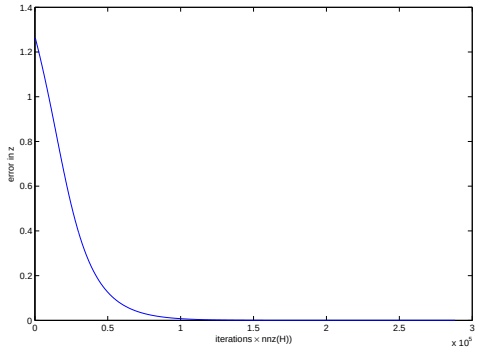
(a) Grid world



(b) Value function



(c) Comparison between Z-learning and KL-learning



(d) Power method

FIGURE 1. Numerical experiment

7. DISCUSSION

The strength of KL control is its very general applicability. The only requirements are the existence of some uncontrolled dynamics governed by a Markov chain, and some state or transition dependent cost. The Markov chain may actually be derived from a graph of allowed transitions, giving every allowed transition equal probability. A disadvantage is that we cannot directly influence the control cost; it is determined by the KL divergence.

KL control is very useful if we know which moves (e.g. in a game) are allowed and we wish to find out which moves are best. The control cost of KL divergence form has a regularizing effect: no move will be made with probability one (unless it is the only allowed move). You could say that there is always a possibility to perform an exploratory move, instead of an exploiting move, under the controlled dynamics.

This immediately suggests the use of KL learning as a reinforcement learning algorithm. The initial transition probabilities represent exploratory dynamics. At every iteration, we could compute a new version of the optimal transition probabilities and use these as a new mixture of exploitation and exploration. The practical implications of this idea will be the topic of further research.

The KL learning algorithm seems to work well in practice and a basis has been provided for its theoretical analysis. Some questions remain to be answered. In particular, if Conjecture 5.11 is true, then regardless of the structure of the problem we know that the solution of the control problem is a locally asymptotically stable equilibrium of the algorithm. It would be even more convenient if a Lyapunov function for the ODE (15) could be found, which would imply global convergence of KL learning.

So far numerical results indicate that KL learning is a reliable algorithm. In the near future we will apply it to practical examples and evaluate its performance relative to other reinforcement learning algorithms.

REFERENCES

[1] Albert Benveniste, Michel Métivier, and Pierre Priouret. *Adaptive algorithms and stochastic approximations*, volume 22 of *Applications of Mathematics (New York)*. Springer-Verlag, Berlin, 1990. Translated from the French by Stephen S. Wilson.

[2] Dimitri P. Bertsekas. *Dynamic programming and optimal control. Vol. I*. Athena Scientific, Belmont, MA, third edition, 2005.

[3] D.P. Bertsekas and J.N. Tsitsiklis. Neuro-dynamic programming, athena scientific. *Belmont, MA*, 1996.

[4] Kappen H.J., GÃČÂşmez V., and Opper M. Optimal control as a graphical model inference problem. *Journal for Machine Learning Research (JMLR)*, pages 1–11, February 2012.

[5] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, 1990. Corrected reprint of the 1985 original.

[6] Roger A. Horn and Charles R. Johnson. *Topics in matrix analysis*. Cambridge University Press, Cambridge, 1991.

[7] Harold J. Kushner and Dean S. Clark. *Stochastic approximation methods for constrained and unconstrained systems*, volume 26 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 1978.

[8] Harold J. Kushner and G. George Yin. *Stochastic approximation and recursive algorithms and applications*, volume 35 of *Applications of Mathematics (New York)*. Springer-Verlag, New York, second edition, 2003. Stochastic Modelling and Applied Probability.

[9] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. *Markov chains and mixing times*. American Mathematical Society, Providence, RI, 2009. With a chapter by James G. Propp and David B. Wilson.

[10] Lennart Ljung. Analysis of recursive stochastic algorithms. *IEEE Trans. Automatic Control*, AC-22(4):551–575, 1977.

[11] Lawrence Perko. *Differential equations and dynamical systems*, volume 7 of *Texts in Applied Mathematics*. Springer-Verlag, New York, third edition, 2001.

[12] R.S. Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3(1):9–44, 1988.

[13] Cs. Szepesvári. *Algorithms for Reinforcement Learning*. Morgan and Claypool, July 2010.

[14] E. Todorov. Linearly-solvable markov decision problems. *Advances in neural information processing systems*, 19:1369, 2007.

[15] C.J.C.H. Watkins. *Learning from delayed rewards*. PhD thesis, King's College, Cambridge, 1989.