A Theory of Regularized Markov Decision Processes

Matthieu Geist [1] **Bruno Scherrer** [2] **Olivier Pietquin** [1]

Abstract

Many recent successful (deep) reinforcement learning algorithms make use of regularization, generally based on entropy or Kullback-Leibler divergence. We propose a general theory of regularized Markov Decision Processes that generalizes these approaches in two directions: we consider a larger class of regularizers, and we consider the general modified policy iteration approach, encompassing both policy iteration and value iteration. The core building blocks of this theory are a notion of regularized Bellman operator and the Legendre-Fenchel transform, a classical tool of convex optimization. This approach allows for error propagation analyses of general algorithmic schemes of which (possibly variants of) classical algorithms such as Trust Region Policy Optimization, Soft Q-learning, Stochastic Actor Critic or Dynamic Policy Programming are special cases. This also draws connections to proximal convex optimization, especially to Mirror Descent.

1. Introduction

Many reinforcement learning algorithms make use of some kind of entropy regularization, with various motivations, such as improved exploration and robustness. Trust Region Policy Optimization (TRPO) (Schulman et al., 2015) is a policy iteration scheme where the greedy step is penalized with a Kullback-Leibler (KL) penalty between two consecutive policies. Dynamic Policy Programming (DPP) (Azar et al., 2012) is a reparametrization of a value iteration scheme regularized by a KL penalty between consecutive policies. Soft Q-learning, eg. (Fox et al., 2016; Schulman et al., 2017; Haarnoja et al., 2017), uses a Shannon entropy regularization in a value iteration scheme, while Soft Actor Critic (SAC) (Haarnoja et al., 2018) uses it in a policy iteration scheme. Value iteration has also been combined with a Tsallis entropy (Lee et al., 2018), with the motivation of having a sparse regularized greedy policy. Other approaches are based on a notion of temporal consistency equation, somehow extending the notion of Bellman residual to the regularized case (Nachum et al., 2017; Dai et al., 2018; Nachum et al., 2018), or on policy gradient (Williams, 1992; Mnih et al., 2016).

This non-exhaustive set of algorithms share the idea of using regularization, but they are derived from sometimes different principles, consider each time a specific regularization, and have ad-hoc analysis, if any. Here, we propose a general theory of regularized Markov Decision Processes (MDPs). To do so, a key observation is that (approximate) dynamic programming, or (A)DP, can be derived solely from the core definition of the Bellman evaluation operator. The framework we propose is built upon a regularized Bellman operator, and on an associated Legendre-Fenchel transform. We study the theoretical properties of these regularized MDPs and of the related regularized ADP schemes. This generalizes many existing theoretical results and provides new ones. Notably, it allows for an error propagation analysis for many of the aforementioned algorithms. This framework also draws connections to convex optimization, especially to Mirror Descent (MD).

A unified view of entropy-regularized MDPs has already been proposed by Neu et al. (2017). They focus on regularized DP through linear programming for the average reward case. Our contribution is complementary to this work (different MDP setting, we do not regularize the same quantity, we do not consider the same DP approach). Our use of the Legendre-Fenchel transform is inspired by Mensch & Blondel (2018), who consider smoothed finite horizon DP in directed acyclic graphs. Our contribution is also complementary to this work, that does not allow recovering aforementioned algorithms nor analyzing them. After a brief background, we introduce regularized MDPs and various related algorithmic schemes based on approximate modified policy iteration (Scherrer et al., 2015), as well as their analysis. All proofs are provided in the appendix.

2. Background

In this section, we provide the necessary background for building the proposed regularized MDPs. We write Δ_X the

[1]Google Research, Brain Team. [2]Université de Lorraine, CNRS, Inria, IECL, F-54000 Nancy, France. Correspondence to: Matthieu Geist <mfgeist@google.com>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

set of probability distributions over a finite set X and Y^X the set of applications from X to the set Y. All vectors are column vectors, except distributions, for left multiplication. We write $\langle \cdot, \cdot \rangle$ the dot product and $\| \cdot \|_p$ the ℓ_p-norm.

2.1. Unregularized MDPs

An MDP is a tuple $\{\mathcal{S}, \mathcal{A}, P, r, \gamma\}$ with $\mathcal{S}$ the finite[1] state space, $\mathcal{A}$ the finite action space, $P \in \Delta_{\mathcal{S}}^{\mathcal{S}\times\mathcal{A}}$ the Markovian transition kernel ($P(s'|s,a)$ denotes the probability of transiting to s' when action a is applied in state s), $r \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ the reward function and $\gamma \in (0,1)$ the discount factor.

A policy $\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}$ associates to each state a distribution over actions. The associated Bellman operator is defined as, for any function $v \in \mathbb{R}^{\mathcal{S}}$,

$$\forall s \in \mathcal{S},\ [T_\pi v](s) = \mathbb{E}_{a\sim\pi(.|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'|s,a}[v(s')]\right].$$

This operator is a γ-contraction in supremum norm and its unique fixed-point is the value function v_π. With $r_\pi(s) = \mathbb{E}_{a\sim\pi(.|s)}[r(s,a)]$ and $P_\pi(s'|s) = \mathbb{E}_{a\sim\pi(.|s)}[P(s'|s,a)]$), the operator can be written as $T_\pi v = r_\pi + \gamma P_\pi v$. For any function $v \in \mathbb{R}^{\mathcal{S}}$, we associate the function $q \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$,

$$q(s,a) = r(s,a) + \gamma \mathbb{E}_{s'|s,a}[v(s')].$$

Thus, the Bellman operator can also be written as $[T_\pi v](s) = \langle \pi(\cdot|s), q(s,\cdot)\rangle = \langle \pi_s, q_s \rangle$. With a slight abuse of notation, we will write $T_\pi v = \langle \pi, q \rangle = (\langle \pi_s, q_s \rangle)_{s\in\mathcal{S}}$.

From this evaluation operator, one can define the Bellman optimality operator as, for any $v \in \mathbb{R}^{\mathcal{S}}$,

$$T_* v = \max_\pi T_\pi v.$$

This operator is also a γ-contraction in supremum norm, and its fixed point is the optimal value function v_*. From the same operator, one can also define the notion of a policy being greedy respectively to a function $v \in \mathbb{R}^{\mathcal{S}}$:

$$\pi' \in \mathcal{G}(v) \Leftrightarrow T_* v = T_{\pi'} v \Leftrightarrow \pi' \in \operatorname*{argmax}_\pi T_\pi v.$$

Given this, we could derive value iteration, policy iteration, modified policy iteration, and so on. Basically, we can do all these things from the core definition of the Bellman evaluation operator. We'll do so from a notion of regularized Bellman evaluation operator.

2.2. Legendre-Fenchel transform

Let $\Omega : \Delta_{\mathcal{A}} \to \mathbb{R}$ be a strongly convex function. The Legendre-Fenchel transform (or convex conjugate) of Ω is $\Omega^* : \mathbb{R}^{\mathcal{A}} \to \mathbb{R}$, defined as

$$\forall q_s \in \mathbb{R}^{\mathcal{A}},\ \Omega^*(q_s) = \max_{\pi_s \in \Delta_{\mathcal{A}}} \langle \pi_s, q_s \rangle - \Omega(\pi_s).$$

[1] We assume a finite space for simplicity of exposition, our results extend to more general cases.

We'll make use of the following properties (Hiriart-Urruty & Lemaréchal, 2012; Mensch & Blondel, 2018).

Proposition 1. *Let Ω be strongly convex, we have the following properties.*

i Unique maximizing argument: $\nabla\Omega^*$ is Lipschitz and satisfies $\nabla\Omega^*(q_s) = \operatorname{argmax}_{\pi_s\in\Delta_{\mathcal{A}}} \langle \pi_s, q_s \rangle - \Omega(\pi_s)$.

ii Boundedness: if there are constants L_Ω and U_Ω such that for all $\pi_s \in \Delta_{\mathcal{A}}$, we have $L_\Omega \leq \Omega(\pi_s) \leq U_\Omega$, then $\max_{a\in\mathcal{A}} q_s(a) - U_\Omega \leq \Omega^*(q_s) \leq \max_{a\in\mathcal{A}} q_s(a) - L_\Omega$.

iii Distributivity: for any $c \in \mathbb{R}$ (and $\mathbf{1}$ the vector of ones), we have $\Omega^*(q_s + c\mathbf{1}) = \Omega^*(q_s) + c$.

iv Monotonicity: $q_{s,1} \leq q_{s,2} \Rightarrow \Omega^*(q_{s,1}) \leq \Omega^*(q_{s,2})$.

A classical example is the negative entropy $\Omega(\pi_s) = \sum_a \pi_s(a) \ln \pi_s(a)$. Its convex conjugate is the smoothed maximum $\Omega^*(q_s) = \ln \sum_a \exp q_s(a)$ and the unique maximizing argument is the usual softmax $\nabla\Omega^*(q_s) = \frac{\exp q_s(a)}{\sum_b \exp q_s(b)}$. For a positive regularizer, one can consider $\Omega(\pi_s) = \sum_a \pi_s(a) \ln \pi_s(a) + \ln|\mathcal{A}|$, that is the KL divergence between π_s and a uniform distribution. Its convex conjugate is $\Omega^*(q_s) = \ln \sum_a \frac{1}{|\mathcal{A}|} \exp q_s(a)$, that is the Mellowmax operator (Asadi & Littman, 2017). The maximizing argument is still the softmax. Another less usual example is the negative Tsallis entropy (Lee et al., 2018), $\Omega(\pi_s) = \frac{1}{2}(\|\pi_s\|_2^2 - 1)$. The analytic convex conjugate is more involved, but it leads to the sparsemax as the maximizing argument (Martins & Astudillo, 2016).

3. Regularized MDPs

The core idea of our contribution is to regularize the Bellman evaluation operator. Recall that $[T_\pi v](s) = \langle \pi_s, q_s \rangle$. A natural idea is to replace it by $[T_{\pi,\Omega} v](s) = \langle \pi_s, q_s \rangle - \Omega(\pi_s)$. To get the related optimality operator, one has to perform state-wise maximization over $\pi_s \in \Delta_{\mathcal{A}}$, which gives the Legendre-Fenchel transform of $[T_{\pi,\Omega} v](s)$. This defines a smoothed maximum (Nesterov, 2005). The related maximizing argument defines the notion of greedy policy.

3.1. Regularized Bellman operators

We now define formally these regularized Bellman operators. With a slight abuse of notation, we write $\Omega(\pi) = (\Omega(\pi_s))_{s\in\mathcal{S}}$ (and similarly for Ω^* and $\nabla\Omega^*$).

Definition 1 (Regularized Bellman operators)**.** *Let $\Omega : \Delta_{\mathcal{A}} \to \mathbb{R}$ be a strongly convex function. For any $v \in \mathbb{R}^{\mathcal{S}}$ define $q \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ as $q(s,a) = r(s,a) + \gamma \mathbb{E}_{s'|s,a}[v(s')]$. The regularized Bellman evaluation operator is defined as*

$$T_{\pi,\Omega} : v \in \mathbb{R}^{\mathcal{S}} \to T_{\pi,\Omega} v = T_\pi v - \Omega(\pi) \in \mathbb{R}^{\mathcal{S}},$$

that is, state-wise, $[T_{\pi,\Omega}v](s) = \langle \pi_s, q_s \rangle - \Omega(\pi_s)$. *The regularized Bellman optimality operator is defined as*

$$T_{*,\Omega} : v \in \mathbb{R}^{\mathcal{S}} \to T_{*,\Omega}v = \max_{\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}} T_{\pi,\Omega}v = \Omega^*(q) \in \mathbb{R}^{\mathcal{S}},$$

that is, state-wise, $[T_{*,\Omega}v](s) = \Omega^*(q_s)$. *For any function* $v \in \mathbb{R}^{\mathcal{S}}$, *the associated unique greedy policy is defined as*

$$\pi' = \mathcal{G}_\Omega(v) = \nabla\Omega^*(q) \Leftrightarrow T_{\pi',\Omega}v = T_{*,\Omega}v,$$

that is, state-wise, $\pi'_s = \nabla\Omega^*(q_s)$.

To be really useful, these operators should satisfy the same properties as the classical ones. It is indeed the case (we recall that all proofs are provided in the appendix).

Proposition 2. *The operator* $T_{\pi,\Omega}$ *is affine and we have the following properties.*

i Monotonicity: let $v_1, v_2 \in \mathbb{R}^{\mathcal{S}}$ *such that* $v_1 \geq v_2$. *Then,*

$$T_{\pi,\Omega}v_1 \geq T_{\pi,\Omega}v_2 \text{ and } T_{*,\Omega}v_1 \geq T_{*,\Omega}v_2.$$

ii Distributivity: for any $c \in \mathbb{R}$, *we have that*

$$T_{\pi,\Omega}(v + c\mathbf{1}) = T_{\pi,\Omega}v + \gamma c\mathbf{1}$$
$$\text{and } T_{*,\Omega}(v + c\mathbf{1}) = T_{*,\Omega}v + \gamma c\mathbf{1}.$$

iii Contraction: both operators are γ*-contractions in supremum norm. For any* $v_1, v_2 \in \mathbb{R}^{\mathcal{S}}$,

$$\|T_{\pi,\Omega}v_1 - T_{\pi,\Omega}v_2\|_\infty \leq \gamma\|v_1 - v_2\|_\infty$$
$$\text{and } \|T_{*,\Omega}v_1 - T_{*,\Omega}v_2\|_\infty \leq \gamma\|v_1 - v_2\|_\infty.$$

3.2. Regularized value functions

The regularized operators being contractions, we can define regularized value functions as their unique fixed-points. Notice that from the following definitions, we could also easily derive regularized Bellman operators on q-functions.

Definition 2 (Regularized value function of policy π)**.** *Noted* $v_{\pi,\Omega}$, *it is defined as the unique fixed point of the operator* $T_{\pi,\Omega}$: $v_{\pi,\Omega} = T_{\pi,\Omega}v_{\pi,\Omega}$. *We also define the associated state-action value function* $q_{\pi,\Omega}$ *as*

$$q_{\pi,\Omega}(s,a) = r(s,a) + \gamma\mathbb{E}_{s'|s,a}[v_{\pi,\Omega}(s')]$$
$$\text{with } v_{\pi,\Omega}(s) = \mathbb{E}_{a\sim\pi(.|s)}[q_{\pi,\Omega}(s,a)] - \Omega(\pi(.|s)).$$

Thus, the regularized value function is simply the unregularized value of π for the reward $r_\pi - \Omega(\pi)$, that is $v_{\pi,\Omega} = (I - \gamma P_\pi)^{-1}(r_\pi - \Omega(\pi))$.

Definition 3 (Regularized optimal value function)**.** *Noted* $v_{*,\Omega}$, *it is the unique fixed point of the operator* $T_{*,\Omega}$: $v_{*,\Omega} = T_{*,\Omega}v_{*,\Omega}$. *We also define the associated state-action value function* $q_{*,\Omega}(s,a)$ *as*

$$q_{*,\Omega}(s,a) = r(s,a) + \gamma\mathbb{E}_{s'|s,a}[v_{*,\Omega}(s')]$$
$$\text{with } v_{*,\Omega}(s) = \Omega^*(q_{*,\Omega}(s,.)).$$

The function $v_{*,\Omega}$ is indeed the optimal value function, thanks to the following result.

Theorem 1 (Optimal regularized policy)**.** *The policy* $\pi_{*,\Omega} = \mathcal{G}_\Omega(v_{*,\Omega})$ *is the unique optimal regularized policy, in the sense that for all* $\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}$, $v_{\pi_{*,\Omega},\Omega} = v_{*,\Omega} \geq v_{\pi,\Omega}$.

When regularizing the MDP, we change the problem at hand. The following result relates value functions in (un)regularized MDPs.

Proposition 3. *Assume that* $L_\Omega \leq \Omega \leq U_\Omega$. *Let* π *be any policy. We have that* $v_\pi - \frac{U_\Omega}{1-\gamma}\mathbf{1} \leq v_{\pi,\Omega} \leq v_\pi - \frac{L_\Omega}{1-\gamma}\mathbf{1}$ *and* $v_* - \frac{U_\Omega}{1-\gamma}\mathbf{1} \leq v_{*,\Omega} \leq v_* - \frac{L_\Omega}{1-\gamma}\mathbf{1}$.

Regularization changes the optimal policy, the next result shows how it performs in the original MDP.

Theorem 2. *Assume that* $L_\Omega \leq \Omega \leq U_\Omega$. *We have that*

$$v_* - \frac{U_\Omega - L_\Omega}{1-\gamma} \leq v_{\pi_{*,\Omega}} \leq v_*.$$

3.3. Related Works

Some of these results already appeared in the literature, in different forms and with specific regularizers. For example, the contraction of $T_{*,\Omega}$ (Prop. 2) was shown in various forms, e.g. (Fox et al., 2016; Asadi & Littman, 2017; Dai et al., 2018), as well as the relation between (un)regularized optimal value functions (Th. 2), e.g. (Lee et al., 2018; Dai et al., 2018). The link to Legendre-Fenchel has also been considered before, e.g. (Dai et al., 2018; Mensch & Blondel, 2018; Richemond & Maginnis, 2017).

The core contribution of Sec. 3 is the regularized Bellman operator, inspired by Nesterov (2005) and Mensch & Blondel (2018). It allows building in a principled and general way regularized MDPs, and generalizing existing results easily. More importantly, it is the core building block of regularized (A)DP, studied in the next sections. The framework and analysis we propose next rely heavily on this formalism.

4. Regularized Modified Policy Iteration

Having defined the notion of regularized MDPs, we still need algorithms that solve them. As the regularized Bellman operators have the same properties as the classical ones, we can apply classical dynamic programming. Here, we consider directly the modified policy iteration approach (Puterman & Shin, 1978), that we regularize (reg-MPI for short):

$$\begin{cases} \pi_{k+1} = \mathcal{G}_\Omega(v_k) \\ v_{k+1} = (T_{\pi_{k+1},\Omega})^m v_k \end{cases}. \tag{1}$$

Given an initial v_0, reg-MPI iteratively performs a regularized greedy step to get π_{k+1} and a partial regularized evaluation step to get v_{k+1}.

With $m = 1$, we retrieve a regularized value iteration algorithm, that can be simplified as $v_{k+1} = T_{*,\Omega} v_k$ (as π_{k+1} is greedy resp. to v_k, we have $T_{\pi_{k+1},\Omega} v_k = T_{*,\Omega} v_k$). With $m = \infty$, we obtain a regularized policy iteration algorithm, that can be simplified as $\pi_{k+1} = \mathcal{G}_\Omega(v_{\pi_k,\Omega})$ (indeed, with a slight abuse of notation, $(T_{\pi_k,\Omega})^\infty v_{k-1} = v_{\pi_k,\Omega}$).

Before studying the convergence and rate of convergence of this general algorithmic scheme (with approximation), we discuss its links to state of the art algorithms (and more generally how it can be practically instantiated).

4.1. Related algorithms

Most existing schemes consider the negative entropy as the regularizer. Usually, it is also more convenient to work with q-functions. First, we consider the case $m = 1$. In the exact case, the regularized value iteration scheme can be written

$$q_{k+1}(s,a) = r(s,a) + \gamma \mathbb{E}_{s'|s,a}[\Omega^*(q_k(s',\cdot))].$$

In the entropic case, $\Omega^*(q_k(s,\cdot)) = \ln \sum_a \exp q_k(s,a)$. In an approximate setting, the q-function can be parameterized by parameters θ (for example, the weights of a neural network), write $\bar{\theta}$ the target parameters (computed during the previous iteration) and $\hat{\mathbb{E}}$ the empirical expectation over sampled transitions (s_i, a_i, r_i, s'_i), an iteration amounts to minimize the expected loss

$$J(\theta) = \hat{\mathbb{E}}\left[\left(\hat{q}_i - q_\theta(s_i,a_i)\right)^2\right] \quad (2)$$
$$\text{with } \hat{q}_i = r_i + \gamma \Omega^*(q_{\bar{\theta}}(s'_i,\cdot)).$$

Getting a practical algorithm may require more work, for example for estimating $\Omega^*(q_{\bar{\theta}}(s'_i,\cdot))$ in the case of continuous actions (Haarnoja et al., 2017), but this is the core principle of soft Q-learning (Fox et al., 2016; Schulman et al., 2017). This idea has also been applied using the Tsallis entropy as the regularizer (Lee et al., 2018).

Alternatively, assume that q_k has been estimated. One could compute the regularized greedy policy analytically, $\pi_{k+1}(\cdot|s) = \nabla\Omega^*(q_k(s,\cdot))$. Instead of computing this for any state-action couple, one can generalize this from observed transitions to any state-action couple through a parameterized policy π_w, by minimizing the KL divergence between both distributions:

$$J(w) = \hat{\mathbb{E}}[\mathrm{KL}(\pi_w(\cdot|s_i)||\nabla\Omega^*(q_k(s_i,.)))]. \quad (3)$$

This is done in SAC (Haarnoja et al., 2018), with an entropic regularizer (and thus $\nabla\Omega^*(q_k(s,.)) = \frac{\exp q_k(s,\cdot)}{\sum_a \exp q_k(s,a)}$). This is also done in Maximum A Posteriori Policy Optimization (MPO) (Abdolmaleki et al., 2018b) with a KL regularizer (a case we discuss Sec. 5), or by Abdolmaleki et al. (2018a) with more general "conservative" greedy policies.

Back to SAC, q_k is estimated using a TD-like approach, by minimizing[2] for the current policy π:

$$J(\theta) = \hat{\mathbb{E}}[(\hat{q}_i - q_\theta(s_i,a_i))^2] \quad (4)$$
$$\text{with } \hat{q}_i = r_i + \gamma(\mathbb{E}_{a\sim\pi(\cdot|s'_i)}[q_{\bar{\theta}}(s'_i,a)] - \Omega(\pi(\cdot,s'_i)).$$

For SAC, we have $\Omega(\pi(\cdot,s)) = \mathbb{E}_{a\sim\pi(\cdot|s)}[\ln \pi(a|s)]$ specifically (negative entropy). This approximate evaluation step corresponds to $m = 1$, and SAC is therefore more a VI scheme than a PI scheme, as presented by Haarnoja et al. (2018) (the difference with soft Q-learning lying in how the greedy step is performed, implicitly or explicitly). It could be extended to the case $m > 1$ in two ways. One possibility is to minimize m times the expected loss (4), updating the target parameter vector $\bar{\theta}$ between each optimization, but keeping the policy π fixed. Another possibility is to replace the 1-step rollout of Eq. (4) by an m-step rollout (similar to classical m-step rollouts, up to the additional regularizations correcting the rewards). Both are equivalent in the exact case, but not in the general case.

Depending on the regularizer, Ω^* or $\nabla\Omega^*$ might not be known analytically. In this case, one can still solve the greedy step directly. Recall that the regularized greedy policy satisfies $\pi_{k+1} = \max_\pi T_{\pi,\Omega} v_k$. In an approximate setting, this amounts to maximize[3]

$$J(w) = \hat{\mathbb{E}}\left[\mathbb{E}_{a\sim\pi_w(\cdot|s_i)}[q_k(s_i,a)] - \Omega(\pi_w(\cdot|s_i)\right]. \quad (5)$$

This improvement step is used by Riedmiller et al. (2018) with an entropy, as well as by TRPO (up to the fact that the objective is constrained rather than regularized), with a KL regularizer (see Sec. 5).

To sum up, for any regularizer Ω, with $m = 1$ one can concatenate greedy and evaluation steps as in Eq. (2), with $m \geq 1$ one can estimate the greedy policy using either Eqs. (3) or (5), and estimate the q-function using Eq. 4, either performed m times repeatedly or combined with m-step rollouts, possibly combined with off-policy correction such as importance sampling or Retrace (Munos et al., 2016).

4.2. Analysis

We analyze the propagation of errors of the scheme depicted in Eq. (1), and as a consequence, its convergence and rate of convergence. To do so, we consider possible errors in both the (regularized) greedy and evaluation steps,

$$\begin{cases} \pi_{k+1} = \mathcal{G}_\Omega^{\epsilon'_{k+1}}(v_k) \\ v_{k+1} = (T_{\pi_{k+1},\Omega})^m v_k + \epsilon_{k+1} \end{cases}, \quad (6)$$

[2] Actually, a separate network is used to estimate the value function, but it is not critical here.

[3] One could add a state-dependant baseline to q_k, eg. v_k, this does not change the maximizer but can reduce the variance.

with $\pi_{k+1} = \mathcal{G}_\Omega^{\epsilon'_{k+1}}(v_k)$ meaning that for any policy π, we have $T_{\pi,\Omega} v_k \leq T_{\pi_{k+1},\Omega} v_k + \epsilon'_{k+1}$. The following analysis is basically the same as the one of Approximate Modified Policy Iteration (AMPI) (Scherrer et al., 2015), thanks to the results of Sec. 3 (especially Prop. 2).

The distance we bound is the loss $l_{k,\Omega} = v_{*,\Omega} - v_{\pi_k,\Omega}$. The bound will involve the terms $d_0 = v_{*,\Omega} - v_0$ and $b_0 = v_0 - T_{\pi_1,\Omega} v_0$. It requires also defining the following.

Definition 4 (Γ-matrix (Scherrer et al., 2015)). *For $n \in \mathbb{N}^*$, $\mathbb{P}_n$ is the set of transition kernels defined as **1)** for any set of n policies $\{\pi_1, \ldots, \pi_n\}$, $\prod_{i=1}^n (\gamma P_{\pi_i}) \in \mathbb{P}_n$ and **2)** for any $\alpha \in (0,1)$ and $(P_1, P_2) \in \mathbb{P}_n \times \mathbb{P}_n$, $\alpha P_1 + (1-\alpha) P_2 \in \mathbb{P}_n$. Any element of $\mathbb{P}_n$ is denoted Γ^n.*

We first state a point-wise bound on the loss. This is the same bound as for AMPI, generalized to regularized MDPs.

Theorem 3. *After k iterations of scheme (6), we have*

$$l_{k,\Omega} \leq 2 \sum_{i=1}^{k-1} \sum_{j=i}^{\infty} \Gamma^j |\epsilon_{k-i}| + \sum_{i=0}^{k-1} \sum_{j=i}^{\infty} \Gamma^j |\epsilon'_{k-i}| + h(k)$$

with $h(k) = 2 \sum_{j=k}^{\infty} \Gamma^j |d_0|$ or $h(k) = 2 \sum_{j=k}^{\infty} \Gamma^j |b_0|$.

Next, we provide a bound on the weighted ℓ_p-norm of the loss, defined for a distribution ρ as $\|l_k\|_{p,\rho}^p = \rho |l_k|^p$. Again, this is the AMPI bound generalized to regularized MDPs.

Corollary 1. *Let ρ and μ be distributions. Let p, q and q' such that $\frac{1}{q} + \frac{1}{q'} = 1$. Define the concentrability coefficients $C_q^i = \frac{1-\gamma}{\gamma^i} \sum_{j=i}^{\infty} \gamma^j \max_{\pi_1,\ldots,\pi_j} \left\| \frac{\rho P_{\pi_1} P_{\pi_2} \ldots P_{\pi_j}}{\mu} \right\|_{q,\mu}$. After k iterations of scheme (6), the loss satisfies*

$$\|l_{k,\Omega}\|_{p,\rho} \leq 2 \sum_{i=1}^{k-1} \frac{\gamma^i}{1-\gamma} (C_q^i)^{\frac{1}{p}} \|\epsilon_{k-i}\|_{pq',\mu} + \sum_{i=0}^{k-1} \frac{\gamma^i}{1-\gamma} (C_q^i)^{\frac{1}{p}} \|\epsilon'_{k-i}\|_{pq',\mu} + g(k)$$

with $g(k) = \frac{2\gamma^k}{1-\gamma} (C_q^i)^{\frac{1}{p}} \min(\|d_0\|_{pq',\mu}, \|b_0\|_{pq',\mu})$.

As this is the same bound (up to the fact that it deals with regularized MDPs) as the one of AMPI, we refer to Scherrer et al. (2015) for a broad discussion about it. It is similar to other error propagation analyses in reinforcement learning, and generalizes those that could be obtained for regularized value or policy iteration. The factor m does not appear in the bound. This is also discussed by Scherrer et al. (2015), but basically this depends on where the error is injected. We could derive a regularized version of Classification-based Modified Policy Iteration (CBMPI, see Scherrer et al. (2015) again) and make it appear.

So, we get the same bound for reg-MPI that for unregularized AMPI, no better nor worse. This is a good thing, as it justifies considering regularized MDPs, but it does no explain the good empirical results of related algorithms.

With regularization, policies will be more stochastic than in classical approximate DP (that tends to produce deterministic policies). Such stochastic policies can induce lower concentrability coefficients. We also hypothesize that regularizing the greedy step helps controlling the related approximation error, that is the $\|\epsilon'_{k-i}\|_{pq',\mu}$ terms. Digging this question would require instantiating more the algorithmic scheme and performing a finite sample analysis of the resulting optimization problems. We left this for future work, and rather pursue the general study of solving regularized MDPs, with varying regularizers now.

5. Mirror Descent Modified Policy Iteration

Solving a regularized MDP provides a solution that differs from the one of the unregularized MDP (see Thm. 2). The problem we address here is estimating the original optimal policy while solving regularized greedy steps. Instead of considering a fixed regularizer $\Omega(\pi)$, the key idea is to penalize a divergence between the policy π and the policy obtained at the previous iteration of an MPI scheme. We consider more specifically the Bregman divergence generated by the strongly convex regularizer Ω.

Let π' be some given policy (typically π_k, when computing π_{k+1}), the Bregman divergence generated by Ω is

$$\begin{aligned} \Omega_{\pi'_s}(\pi_s) &= D_\Omega(\pi_s || \pi'_s) \\ &= \Omega(\pi_s) - \Omega(\pi'_s) - \langle \nabla \Omega(\pi'_s), \pi_s - \pi'_s \rangle. \end{aligned}$$

For example, the KL divergence is generated by the negative entropy: $\mathrm{KL}(\pi_s || \pi'_s) = \sum_a \pi_s(a) \ln \frac{\pi_s(a)}{\pi'_s(a)}$. With a slight abuse of notation, as before, we will write

$$\Omega_{\pi'}(\pi) = D_\Omega(\pi || \pi') = \Omega(\pi) - \Omega(\pi') - \langle \nabla \Omega(\pi'), \pi - \pi' \rangle.$$

This divergence is always positive, it satisfies $\Omega_{\pi'}(\pi') = 0$, and it is strongly convex in π (so Prop. 1 applies).

We consider a reg-MPI algorithmic scheme with a Bregman divergence replacing the regularizer. For the greedy step, we simply consider $\pi_{k+1} = \mathcal{G}_{\Omega_{\pi_k}}(v_k)$, that is

$$\pi_{k+1} = \operatorname*{argmax}_\pi \langle q_k, \pi \rangle - D_\Omega(\pi || \pi_k).$$

This is similar to the update of the Mirror Descent (MD) algorithm in its proximal form (Beck & Teboulle, 2003), with $-q_k$ playing the role of the gradient in MD. Therefore, we will call this approach Mirror Descent Modified Policy Iteration (MD-MPI). For the partial evaluation step, we can regularize according to the previous policy π_k, that is $v_{k+1} = (T_{\pi_{k+1},\Omega_{\pi_k}})^m v_k$, or according to the current policy π_{k+1}, that is $v_{k+1} = (T_{\pi_{k+1},\Omega_{\pi_{k+1}}})^m v_k$. As

$\Omega_{\pi_{k+1}}(\pi_{k+1}) = 0$, this simplifies as $v_{k+1} = (T_{\pi_{k+1}})^m v_k$, that is a partial unregularized evaluation.

To sum up, we will consider two general algorithmic schemes based on a Bregman divergence, MD-MPI types 1 and 2 respectively defined as

$$\begin{cases} \pi_{k+1} = \mathcal{G}_{\Omega_{\pi_k}}(v_k) \\ v_{k+1} = (T_{\pi_{k+1},\Omega_{\pi_k}})^m v_k \end{cases}, \begin{cases} \pi_{k+1} = \mathcal{G}_{\Omega_{\pi_k}}(v_k) \\ v_{k+1} = (T_{\pi_{k+1}})^m v_k \end{cases}$$

and both initialized with some v_0 and π_0.

5.1. Related algorithms

To derive practical algorithms, the recipes provided in Sec. 4.1 still apply, just replacing Ω by Ω_{π_k}. If $m = 1$, greedy and evaluation steps can be concatenated (only for MD-MPI type 1). In the general case ($m \geq 1$) the greedy policy (for MD-MPI types 1 and 2) can be either directly estimated (Eq. (5)) or trained to generalize the analytical solution (Eq. (3)). The partial evaluation can be done using a TD-like approach, either done repeatedly while keeping the policy fixed or considering m-step rollouts. Specifically, in the case of a KL divergence, one could use the fact that $\Omega^*_{\pi_k}(q_k(s,\cdot)) = \ln \sum_a \pi_k(a|s) \exp q_k(s,a)$ and that $\nabla \Omega^*_{\pi_k}(q_k(s,\cdot)) = \frac{\pi_k(\cdot|s) \exp q_k(s,\cdot)}{\sum_a \pi_k(a|s) \exp q_k(s,a)}$.

This general algorithmic scheme allows recovering state of the art algorithms. For example, MD-MPI type 2 with $m = \infty$ and a KL divergence as the regularizer is TRPO (Schulman et al., 2015) (with a direct optimization of the regularized greedy step, as in Eq. (5), up to the use of a constraint instead of a regularization). DPP can be seen as a reparametrization[4] of MD-MPI type 1 with $m = 1$ (Azar et al., 2012, Appx. A). MPO (Abdolmaleki et al., 2018b) is derived from an expectation-maximization principle, but it can be seen as an instantiation of MD-MPI type 2, with a KL divergence, a greedy step similar to Eq. (3) (up to additional regularization) and an evaluation step similar to Eq. (4) (without regularization, as in type 2, with m-step return and with the Retrace off-policy correction). This also generally applies to the approach proposed by Abdolmaleki et al. (2018a) (up to an additional subtelty in the greedy step consisting in decoupling updates for the mean and variance in the case of a Gaussian policy).

5.2. Analysis

Here, we propose to analyze the error propagation of MD-MPI (and thus, its convergence and rate of convergence). We think this is an important topic, as it has only been partly studied for the special cases discussed in Sec. 5.1. For example, DPP enjoys an error propagation analysis in supremum norm (yet it is a reparametrization of a special case of MD-MPI, so not directly covered here), while TRPO or MPO are only guaranteed to have monotonic improvements, under some assumptions. Notice that we do not claim that our analysis covers all these cases, but it will provide the key technical aspects to analyze similar schemes (much like CBMPI compared to AMPI, as discussed in Sec. 4.2 or by Scherrer et al. (2015); where the error is injected changes the bounds).

In Sec. 4.2, the analysis was a straightforward adaptation of the one of AMPI, thanks to the results of Sec. 3 (the regularized quantities behave like their unregularized counterparts). It is no longer the case here, as the regularizer changes over iterations, depending on what has been computed so far. We will notably need a slightly different notion of approximate regularized greediness.

Definition 5 (Approximate Bregman divergence-regularized greediness)**.** *Write $J_k(\pi)$ the (negative) optimization problem corresponding to the Bregman divergence-regularized greediness (that is, negative regularized Bellman operator of π applied to v_k):*

$$J_k(\pi) = \langle -q_k, \pi \rangle + D_\Omega(\pi||\pi_k) = -T_{\pi,\Omega_{\pi_k}} v_k.$$

We write $\pi_{k+1} \in \mathcal{G}^{\epsilon'_{k+1}}_{\Omega_{\pi_k}}(v_k)$ if for any policy π the policy π_{k+1} satisfies

$$\langle \nabla J_k(\pi_{k+1}), \pi - \pi_{k+1} \rangle + \epsilon'_{k+1} \geq 0.$$

In other words, $\pi_{k+1} \in \mathcal{G}^{\epsilon'_{k+1}}_{\Omega_{\pi_k}}(v_k)$ means that π_{k+1} is ϵ'_{k+1}-close to satisfying the optimality condition, which might be slightly stronger than being ϵ'_{k+1}-close to the optimal (as for AMPI or reg-MPI). Given this, we consider MD-MPI with errors in both greedy and evaluation steps, type 1

$$\begin{cases} \pi_{k+1} = \mathcal{G}^{\epsilon'_{k+1}}_{\Omega_{\pi_k}}(v_k) \\ v_{k+1} = (T_{\pi_{k+1},\Omega_{\pi_k}})^m v_k + \epsilon_{k+1} \end{cases}$$

and type 2

$$\begin{cases} \pi_{k+1} = \mathcal{G}^{\epsilon'_{k+1}}_{\Omega_{\pi_k}}(v_k) \\ v_{k+1} = (T_{\pi_{k+1}})^m v_k + \epsilon_{k+1} \end{cases}.$$

The quantity we are interested in is $v_* - v_{\pi_k}$, that is suboptimality in the unregularized MDP, while the algorithms compute new policies with a regularized greedy operator. So, we need to relate regularized and unregularized quantities when using a Bregman divergence based on the previous policy. The next lemma is the key technical result that allows analyzing MD-MPI.

[4] Indeed, if one see MD-MPI as a Mirror Descent approach, one can see DPP as a dual averaging approach, somehow updating a kind of cumulative q-functions directly in the dual. However, how to generalize this beyond the specific DPP algorithm is unclear, and we let it for future work.

Lemma 1. *Assume that $\pi_{k+1} \in \mathcal{G}^{\epsilon'_{k+1}}_{\Omega_{\pi_k}}(v_k)$, as defined in Def. 5. Then, the policy π_{k+1} is ϵ'_{k+1}-close to the regularized greedy policy, in the sense that for any policy π*

$$T_{\pi,\Omega_{\pi_k}} v_k - T_{\pi_{k+1},\Omega_{\pi_k}} v_k \le \epsilon'_{k+1}.$$

Moreover, we can relate the (un)regularized Bellman operators applied to v_k. For any policy π (so notably for the unregularized optimal policy π_*), we have

$$T_\pi v_k - T_{\pi_{k+1},\Omega_{\pi_k}} v_k \le \epsilon'_{k+1} + D_\Omega(\pi||\pi_k) - D_\Omega(\pi||\pi_{k+1}),$$

$$T_\pi v_k - T_{\pi_{k+1}} v_k \le \epsilon'_{k+1} + D_\Omega(\pi||\pi_k) - D_\Omega(\pi||\pi_{k+1}).$$

We're interested in bounding the loss $l_k = v_* - v_{\pi_k}$, or some related quantity, for each type of MD-MPI. To do so, we introduce quantities similar to the ones of the AMPI analysis (Scherrer et al., 2015), defined respectively for types 1 and 2: **1)** The distance between the optimal value function and the value before approximation at the k^{th} iteration, $d^1_k = v_* - (T_{\pi_k,\Omega_{\pi_{k-1}}})^m v_{k-1} = v_* - (v_k - \epsilon_k)$ and $d^2_k = v_* - (T_{\pi_k})^m v_{k-1} = v_* - (v_k - \epsilon_k)$; **2)** The shift between the value before approximation and the policy value a iteration k, $s^1_k = (T_{\pi_k,\Omega_{\pi_{k-1}}})^m v_{k-1} - v_{\pi_k} = (v_k - \epsilon_k) - v_{\pi_k}$ and $s^2_k = (T_{\pi_k})^m v_{k-1} - v_{\pi_k} = (v_k - \epsilon_k) - v_{\pi_k}$; **3)** the Bellman residual at iteration k, $b^1_k = v_k - T_{\pi_{k+1},\Omega_{\pi_k}} v_k$ and $b^2_k = v_k - T_{\pi_{k+1}} v_k$.

For both types ($h \in \{1, 2\}$), we have that $l^h_k = d^h_k + s^h_k$, so bounding the loss requires bounding these quantities, which is done in the following lemma (quantities related to both types enjoy the same bounds).

Lemma 2. *Let $k \ge 1$, define $x_k = (I - \gamma P_{\pi_k})\epsilon_k + \epsilon'_{k+1}$ and $y_k = -\gamma P_{\pi_*}\epsilon_k + \epsilon'_{k+1}$, as well as $\delta_k(\pi_*) = D_\Omega(\pi_*||\pi_k) - D_\Omega(\pi_*||\pi_{k+1})$. We have for $h \in \{1, 2\}$:*

$$b^h_k \le (\gamma P_{\pi_k})^m b^h_{k-1} + x_k,$$
$$s^h_k \le (\gamma P_{\pi_k})^m (I - \gamma P_{\pi_k})^{-1} b^h_{k-1} \text{ and}$$
$$d^h_{k+1} \le \gamma P_{\pi_*} d^h_k + y_k + \sum_{j=1}^{m-1} (\gamma P_{\pi_{k+1}})^j b^h_k + \delta_k(\pi_*).$$

These bounds are almost the same as the ones of AMPI (Scherrer et al., 2015, Lemma 2), up to the additional $\delta_k(\pi_*)$ term in the bound of the distance d^h_k. One can notice that summing these terms gives a telescopic sum: $\sum_{k=0}^{K-1} \delta_k(\pi_*) = D_\Omega(\pi_*||\pi_0) - D_\Omega(\pi_*||\pi_K) \le D_\Omega(\pi_*||\pi_0) \le \sup_\pi D_\Omega(\pi||\pi_0)$. For example, if D_Ω is the KL divergence and π_0 the uniform policy, then $\|\sup_\pi D_\Omega(\pi||\pi_0)\|_\infty = \ln|\mathcal{A}|$. This suggests that we must bound the regret L_K defined as

$$L_k = \sum_{k=1}^{K} l_k = \sum_{k=1}^{K} (v_* - v_{\pi_k}).$$

Theorem 4. *Define $R_{\Omega_{\pi_0}} = \|\sup_\pi D_\Omega(\pi||\pi_0)\|_\infty$, after K iterations of MD-MPI, for $h = 1, 2$, the regret satisfies*

$$L_K \le 2\sum_{k=2}^{K}\sum_{i=1}^{k-1}\sum_{j=i}^{\infty} \Gamma^j |\epsilon_{k-i}| + \sum_{k=1}^{K}\sum_{i=0}^{k-1}\sum_{j=i}^{\infty} \Gamma^j |\epsilon'_{k-i}| + \sum_{k=1}^{K} h(k) + \frac{1-\gamma^K}{(1-\gamma)^2} R_{\Omega_{\pi_0}} \mathbf{1}.$$

with $h(k) = 2\sum_{j=k}^{\infty} \Gamma^j |d_0|$ or $h(k) = 2\sum_{j=k}^{\infty} \Gamma^j |b_0|$.

From this,we can derive an ℓ_p-bound for the regret.

Corollary 2. *Let ρ and μ be distributions over states. Let p, q and q' be such that $\frac{1}{q} + \frac{1}{q'} = 1$. Define the concentrability coefficients C^i_q as in Cor. 1. After K iterations, the regret satisfies*

$$\|L_K\|_{p,\rho} \le 2\sum_{k=2}^{K}\sum_{i=1}^{k-1} \frac{\gamma^i}{1-\gamma} (C^i_q)^{\frac{1}{p}} \|\epsilon_{k-i}\|_{pq',\mu} + \sum_{k=1}^{K}\sum_{i=0}^{k-1} \frac{\gamma^i}{1-\gamma} (C^i_q)^{\frac{1}{p}} \|\epsilon'_{k-i}\|_{pq',\mu} + g(k) + \frac{1-\gamma^K}{(1-\gamma)^2} R_{\Omega_{\pi_0}}.$$

with $g(k) = 2\sum_{k=1}^{K} \frac{\gamma^k}{1-\gamma} (C^k_q)^{\frac{1}{p}} \min(\|d_0\|_{pq',\mu}, \|b_0\|_{pq',\mu})$.

This result bounds the regret, while it is usually the loss that is bounded. Both can be related as follows.

Proposition 4. *For any $p \ge 1$ and distribution ρ, we have* $\min_{1\le k\le K} \|v_* - v_{\pi_k}\|_{1,\rho} \le \frac{1}{K}\|L_K\|_{p,\rho}$.

This means that if we can control the average regret, then we can control the loss of the best policy computed so far. This suggests that practically we should not use the last policy, but this best policy.

From Cor. 2 can be derived the convergence and rate of convergence of MD-MPI in the exact case.

Corollary 3. *Both MD-MPI type 1 and 2 enjoy the following rate of convergence, when no approximation is done ($\epsilon_k = \epsilon'_k = 0$),*

$$\frac{1}{K}\|L_K\|_\infty \le \frac{1-\gamma^K}{(1-\gamma)^2} \frac{2\gamma\|v_* - v_0\|_\infty + R_{\Omega_{\pi_0}}}{K}.$$

In classical DP and in regularized DP (see Cor. 1), there is a linear convergence rate (the bound is $\frac{2\gamma^K}{1-\gamma}\|v_* - v_0\|_\infty$), while in this case we only have a logarithmic convergence rate. We also pay an horizon factor (square dependency in $\frac{1}{1-\gamma}$ instead of linear). This is normal, as we bound the regret instead of the loss. Bounding the regret in classical DP would lead to the bound of Cor. 3 (without the $R_{\Omega_{\pi_0}}$ term).

The convergence rate of the loss of MD-MPI is an open question, but a sublinear rate is quite possible. Compared to classical DP, we slow down greediness by adding the Bregman divergence penalty. Yet, this kind of regularization is used in an approximate setting, where it favors stability empirically (even if studying this further would require much more work regarding the $\|\epsilon'_k\|$ term, as discussed in Sec. 4.2).

As far as we know, the only other approach that studies a DP scheme regularized by a divergence and that offers a convergence rate is DPP, up to the reparameterization we discussed earlier. MD-MPI has the same upper-bound as DPP in the exact case (Azar et al., 2012, Thm. 2). However, DPP bounds the loss, while we bound a regret. This means that if the rate of convergence of our loss can be sublinear, it is superlogarithmic (as the rate of the regret is logarithmic), while the rate of the loss of DPP is logarithmic.

To get more insight on Cor. 2, we can group the terms differently, by grouping the errors.

Corollary 4. *With the same notations as Cor. 2, we have*

$$\frac{1}{K}\|L_k\|_{p,\rho} \leq \sum_{i=1}^{K-1} \frac{\gamma^i}{1-\gamma}(C_q^i)^{\frac{1}{p}} \frac{2E_{K-i} + E'_{K-i}}{K} + \frac{1}{K}\left(g(k) + \frac{1-\gamma^K}{(1-\gamma)^2} R_{\Omega_{\pi_0}}\right),$$

with $E_i = \sum_{j=1}^{i} \|\epsilon_j\|_{pq',\mu}$ *and* $E'_i = \sum_{j=1}^{i} \|\epsilon'_j\|_{pq',\mu}$.

Compared to the bound of AMPI (Scherrer et al., 2015, Thm. 7), instead of propagating the errors, we propagate the sum of errors over previous iterations normalized by the total number of iterations. So, contrary to approximate DP, it is no longer the last iterations that have the highest influence on the regret. Yet, we highlight again the fact that we bound a regret, and bounding the regret of AMPI would provide a similar result.

Our result is similar to the error propagation of DPP (Azar et al., 2012, Thm. 5), except that we sum norms of errors, instead of norming a sum of errors, the later being much better (as it allows the noise to cancel over iterations). Yet, as said before, DPP is not a special case of our framework, but a reparameterization of such one. Consequently, while we estimate value functions, DPP estimate roughly at iteration k a sum of k advantage functions (converging to $-\infty$ for any suboptimal action in the exact case). As explained before, where the error is injected does matter. Knowing if the DPP's analysis can be generalized to our framework (MPI scheme, ℓ_p bounds) remains an open question.

To get further insight, we can express the bound using different concentrability coefficients.

Corollary 5. *Define the concentrability coefficient* $C_q^{l,k} = \frac{(1-\gamma)^2}{\gamma^l - \gamma^k} \sum_{i=l}^{k-1} \sum_{j=i}^{\infty} c_q(j)$*, the regret then satisfies*

$$\|L_K\|_{p,\rho} \leq 2\sum_{i=1}^{K-1} \frac{\gamma - \gamma^{i+1}}{(1-\gamma)^2}(C_q^{1,i+1})^{\frac{1}{p}} \|\epsilon_{K-i}\|_{pq',\mu} + \sum_{i=0}^{K-1} \frac{1-\gamma^{i+1}}{(1-\gamma)^2}(C_q^{0,i+1})^{\frac{1}{p}} \|\epsilon'_{K-i}\|_{pq',\mu} + f(k)$$

with $f(k) = \frac{\gamma - \gamma^{K+1}}{(1-\gamma)^2}(C_q^{1,K+1})^{\frac{1}{p}} \min(\|d_0\|_{pq',\mu}, \|b_0\|_{pq',\mu}) + \frac{1-\gamma^K}{(1-\gamma)^2} R_{\Omega_{\pi_0}}$.

We observe again that contrary to ADP, the last iteration does not have the highest influence, and we do not enjoy a decrease of influence at the exponential rate γ towards the initial iterations. However, we bound a different quantity (regret instead of loss), that explains this behavior. Here again, bounding the regret in AMPI would lead to the same bound (up to the term $R_{\Omega_{\pi_0}}$). Moreover, sending p and K to infinity, defining $\epsilon = \sup_j \|\epsilon_j\|_\infty$ and $\epsilon' = \sup_j \|\epsilon'_j\|_\infty$, we get $\limsup_{K\to\infty} \frac{1}{K}\|L_K\|_\infty \leq \frac{2\gamma\epsilon + \epsilon'}{(1-\gamma)^2}$, which is the classical asymptotical bound for approximate value and policy iterations (Bertsekas & Tsitsiklis, 1996) (usually stated without greedy error). It is generalized here to an approximate MPI scheme regularized with a Bregman divergence.

6. Conclusion

We have introduced a general theory of regularized MDPs, where the usual Bellman evaluation operator is modified by either a fixed convex function or a Bregman divergence between consecutive policies. For both cases, we proposed a general algorithmic scheme based on MPI. We shown how many (variations of) existing algorithms could be derived from this general algorithmic scheme, and also analyzed and discussed the related propagation of errors.

We think that this framework can open many perspectives, among which links between (approximate) DP and proximal convex optimization (going beyond mirror descent), temporal consistency equations (roughly regularized Bellman residuals), regularized policy search (maximizing the expected regularized value function), inverse reinforcement learning (thanks to uniqueness of greediness in this regularized framework) or zero-sum Markov games (regularizing the two-player Bellman operators). We develop more these points in the appendix.

This work also lefts open questions, such as combining the propagation of errors with a finite sample analysis, or what specific regularizer one should choose for what context. Some approaches also combine a fixed regularizer and a divergence (Akrour et al., 2018), a case not covered here and worth being investigated.

References

Abdolmaleki, A., Springenberg, J. T., Degrave, J., Bohez, S., Tassa, Y., Belov, D., Heess, N., and Riedmiller, M. Relative entropy regularized policy iteration. *arXiv preprint arXiv:1812.02256*, 2018a.

Abdolmaleki, A., Springenberg, J. T., Tassa, Y., Munos, R., Heess, N., and Riedmiller, M. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations (ICLR)*, 2018b.

Akrour, R., Abdolmaleki, A., Abdulsamad, H., Peters, J., and Neumann, G. Model-free trajectory-based policy optimization with monotonic improvement. *The Journal of Machine Learning Research (JMLR)*, 19(1):565–589, 2018.

Asadi, K. and Littman, M. L. An alternative softmax operator for reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2017.

Azar, M. G., Gómez, V., and Kappen, H. J. Dynamic policy programming. *Journal of Machine Learning Research (JMLR)*, 13(Nov):3207–3245, 2012.

Beck, A. and Teboulle, M. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

Bertsekas, D. P. and Tsitsiklis, J. N. *Neuro-Dynamic Programming*. Athena Scientific, 1st edition, 1996. ISBN 1886529108.

Dai, B., Shaw, A., Li, L., Xiao, L., He, N., Liu, Z., Chen, J., and Song, L. Sbeed: Convergent reinforcement learning with nonlinear function approximation. In *International Conference on Machine Learning (ICML)*, 2018.

Fox, R., Pakman, A., and Tishby, N. Taming the noise in reinforcement learning via soft updates. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2016.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning (ICML)*, 2017.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning (ICML)*, 2018.

Hiriart-Urruty, J.-B. and Lemaréchal, C. *Fundamentals of convex analysis*. Springer Science & Business Media, 2012.

Lee, K., Choi, S., and Oh, S. Sparse markov decision processes with causal sparse tsallis entropy regularization for reinforcement learning. *IEEE Robotics and Automation Letters*, 3(3):1466–1473, 2018.

Martins, A. and Astudillo, R. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *International Conference on Machine Learning (ICML)*, 2016.

Mensch, A. and Blondel, M. Differentiable dynamic programming for structured prediction and attention. In *International Conference on Machine Learning (ICML)*, 2018.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2016.

Munos, R., Stepleton, T., Harutyunyan, A., and Bellemare, M. Safe and efficient off-policy reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1054–1062, 2016.

Nachum, O., Norouzi, M., Xu, K., and Schuurmans, D. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

Nachum, O., Chow, Y., and Ghavamzadeh, M. Path consistency learning in tsallis entropy regularized mdps. *arXiv preprint arXiv:1802.03501*, 2018.

Nesterov, Y. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

Neu, G., Jonsson, A., and Gómez, V. A unified view of entropy-regularized Markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

Puterman, M. L. and Shin, M. C. Modified policy iteration algorithms for discounted markov decision problems. *Management Science*, 24(11):1127–1137, 1978.

Richemond, P. H. and Maginnis, B. A short variational proof of equivalence between policy gradients and soft q learning. *arXiv preprint arXiv:1712.08650*, 2017.

Riedmiller, M., Hafner, R., Lampe, T., Neunert, M., Degrave, J., Wiele, T., Mnih, V., Heess, N., and Springenberg, J. T. Learning by playing solving sparse reward tasks from scratch. In *International Conference on Machine Learning (ICML)*, pp. 4341–4350, 2018.

Scherrer, B., Ghavamzadeh, M., Gabillon, V., Lesner, B., and Geist, M. Approximate modified policy iteration and

its application to the game of tetris. *Journal of Machine Learning Research (JMLR)*, 16:1629–1676, 2015.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International Conference on Machine Learning (ICML)*, 2015.

Schulman, J., Chen, X., and Abbeel, P. Equivalence between policy gradients and soft q-learning. *arXiv preprint arXiv:1704.06440*, 2017.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.