Batch Policy Iteration Algorithms for Continuous Domains

Bilal Piot, Matthieu Geist, Olivier Pietquin

Batch Policy Iteration Algorithms for Continuous Domains

Bilal Piot bilal.piot@inria.fr
Univ. Lille, CNRS, Centrale Lille, INRIA,
UMR 9189 - CRIStAL, F-59000 Lille, France.

Matthieu Geist matthieu.geist@centralesupelec.fr
UMI 2958, GeorgiaTech-CNRS, CentraleSupélec
Université Paris-Saclay, Metz, France.

Olivier Pietquin olivier.pietquin@univ-lille1.fr
Univ. Lille, CNRS, Centrale Lille, INRIA,
UMR 9189 - CRIStAL, F-59000 Lille, France.
Now at Google DeepMind, London, United Kingdom

Editor: EWRL 2016

Abstract

This paper establishes the link between an adaptation of the policy iteration method for Markov decision processes with continuous state and action spaces and the policy gradient method when the differentiation of the mean value is directly done over the policy without parameterization. This approach allows deriving sound and practical batch Reinforcement Learning algorithms for continuous state and action spaces.

Keywords: Batch Reinforcement Learning, Policy gradient.

1. Introduction

Policy Search (PS) (Ng and Jordan, 2000; Fix and Geist, 2012) and more specifically Policy Gradient (PG) methods (Sutton et al., 1999; Peters et al., 2005; Degris et al., 2012; Silver et al., 2014) are well known to have a practical edge compared to value-based methods when it comes to Reinforcement Learning (RL) for Markov Decisions Processes (MDPs) with continuous state and action spaces. This principally comes from the fact that value-based methods, either inspired by the Policy Iteration (PI) (Lagoudakis and Parr, 2003) or Value Iteration (VI) algorithms (Ernst et al., 2005; Riedmiller, 2005), rely on the computation of the maximum of the action-value function over the set of actions: $\max_{a\in A} Q(\pi, s, a)$ (greedy step). If this set is infinite, then those methods become intractable. However, it is possible to adapt Policy Iteration-based RL algorithms, such as Least Squares Policy Iteration (LSPI) (Lagoudakis and Parr, 2003), by changing this global greedy step into a local improvement of the policy made by a gradient step. Indeed, let us take a closer look to the PI method for a finite action space. It consists in two steps, a first step of evaluation of the current policy π_k in order to obtain the action-value function $Q(\pi_k, s, a)$ followed by a greedy step $\pi_{k+1}(s) = \text{argmax}_{a\in A} Q(\pi_k, s, a)$ which improves the action-value function:

Discrete PI scheme

- Evaluation of $Q(\pi_k, ., .)$.
- Greedy step: $\pi_{k+1}(s) = \text{argmax}_{a \in A} Q(\pi_k, s, a)$.

In practice, when faced with batch data, the evaluation step can be realized for instance by the Least Square Temporal Difference (LSTD) algorithm (Bradtke and Barto, 1996) if features for the action-value function are provided. When no features are provided, one could easily adapt for instance the Fitted-Q algorithm (Ernst et al., 2005) that evaluates the optimal action value function in order to evaluate the current value function by replacing the optimality-operator by the evaluation operator.

Obviously, when the action space is continuous such a greedy step becomes cumbersome. However, a local improvement is still possible if we are able to compute the gradient of the actions-value function over the possible actions $\frac{\partial Q(\pi_k, s, \pi_k(s))}{\partial a}$.[1] Then, the global greedy step can be replaced by a local gradient-like improvement step: $\pi_{k+1}(s) = \pi_k(s) + \alpha_k \frac{\partial Q(\pi_k, s, \pi_k(s))}{\partial a}$, where $\alpha_k \in \mathbb{R}_+$. Thus, a canonical adaptation of the PI method in the continuous context is:

Continuous PI scheme

- Evaluation of $Q(\pi_k, ., .)$.
- Local improvement step: $\pi_{k+1}(s) = \pi_k(s) + \alpha_k \frac{\partial Q(\pi_k, s, \pi_k(s))}{\partial a}$.

Our main contribution (see Sec. 3) consists in showing, after introducing some notations relative to MDPs and differentiation in Sec. 2, that this PI method for continuous states and actions MDPs (called continuous PI) is in fact a sound algorithm. More precisely, it is a direct derivation of the PG method when the differentiation of the mean value $J_\nu(\pi) = \int_S V^\pi(s)\nu(ds)$ is directly done over the policy without parameterizing it. This establishes a strong link between PI and PG methods in the continuous setting. It should be noted that this link between PI and PG methods have already been studied in the finite action setting (Scherrer and Geist, 2014) where the relation between Conservative Policy Iteration (CPI) (Kakade and Langford, 2002) and PG methods has been highlighted. Finally, in Sec. 4 as a second contribution, we provide several practical instantiations of this continuous PI method which can be either feature-based or no.

2. Background and notations

Before introducing notations specific to MDPs and differentiation of functions, we give some general notations. Let B be a Borel space, ν a measure on B and $f \in \mathbb{R}^B$ a real function, the integral of f under ν, if it exists, is noted $\int_B f(b)\nu(db)$. The set of probability measures of the Borel space B is noted Δ_B and we have $\mathbb{E}_\nu[f(X)] = \int_B f(b)\nu(db)$, where $\mathbb{E}_\nu[f(X)]$ is the expectation of the function f under the probability measure ν.

Let $(H, \langle ., . \rangle_H)$ (where $\langle ., . \rangle_H$ is the dot product associated to H) and $(G, \langle ., . \rangle_G)$ be Hilbert spaces, $f \in G \to \mathbb{R}^H$ and $\nu \in \Delta_G$, then the notation $\mathbb{E}_\nu[f(X)] = \int_G f(g)\nu(dg)$

1. The notion of partial derivative and the notation associated is fully explained in appendix A.3.

means that $\int_G f(g)\nu(dg) \in \mathbb{R}^H$ is a function such that $\forall h \in H, \left(\int_G f(g)\nu(dg)\right)(h) = \int_G f(g)[h]\nu(dg)$. In particular if $f \in G^H$, then $\int_G \langle f(g), . \rangle_H \nu(dg) \in \mathbb{R}^H$ is a function such that $\forall h \in H, \left(\int_G \langle f(g), . \rangle_H \nu(dg)\right)(h) = \int_G \langle f(g), h \rangle_H \nu(dg)$.

2.1 MDP

We give some brief definitions relative to MDPs with continuous state and action spaces. More precisely, we focus on MDPs where the state space $S = \mathbb{R}^p$ and the action space $A = \mathbb{R}^q$ with $(p, q) \in \mathbb{N}^{*2}$. In that particular case, S and A can be seen as both Hilbert spaces (where the dot product is chosen to be the canonical dot product) and Borel spaces with the canonical distance. A continuous MDP (a complete and more formal presentation is done by Hernández-Lerma and Lasserre (1996)) is a tuple (S, A, R, P, γ) where the state space $S = \mathbb{R}^p$ represents the states of the environment, the action space $A = \mathbb{R}^q$ represents the possible actions the agent can take, the reward $R \in \mathbb{R}^{S \times A}$ is a bounded function that represents the local benefit of doing action a in state s, the dynamics $P(.|.)$ is a stochastic kernel on S given $S \times A$ and $\gamma \in]0, 1[$ is a discount factor. More precisely, if B is a measurable set of S and $(s, a) \in S \times A$, then $P(B|s, a)$ is the probability that the agent is in the measurable set B after doing action a in state s. Here, we focus on deterministic policies ($\pi \in A^S$) which are mappings from states to actions. More precisely, we are interested in the set of all equivalence classes of policies $\pi \in A^S$ such that the Lebesgue integral $\int_S \|\pi(s)\|_A^2 ds$, where $\|.\|_A$ is the canonical norm in A, is finite. This space of policies will be noted $L^2(A, S)$ or in short L^2 and it is an Hilbert space with the following natural dot product

$$\forall f, g \in L^2, \langle f, g \rangle_{L^2} = \int_S \langle f(s), g(s) \rangle_A ds,$$

where $\langle ., . \rangle_A$ is the canonical dot product relative to A. To evaluate a policy, we use the concepts of value function $V \in \mathbb{R}^{L^2 \times S}$ and action-value function $Q \in \mathbb{R}^{L^2 \times S \times A}$:

$$Q(\pi, s, a) = \mathbb{E}_{s,a}^{\pi}[\sum_{t=0}^{+\infty} \gamma^t R(s_t, a_t)], \quad V(\pi, s) = Q(\pi, s, \pi(s)),$$

where $\mathbb{E}_{s,a}^{\pi}$ is the expectation over the distribution of the admissible trajectories $(s_0, a_0, s_1, ., \dots)$ obtained by executing the policy π starting from $s_0 = s$ and $a_0 = a$. These functions are well-defined when the policies are deterministic and the reward bounded (Hernández-Lerma and Lasserre, 1996). Moreover, $V(\pi, s)$ follows the Bellman equation:

$$V(\pi, s) = R(s, \pi(s)) + \gamma \int_S V(\pi, s') P(ds', s, \pi(s)). \tag{1}$$

One can show that verifying the Bellman equation is equivalent to the existence of a distribution $d_{\nu,\pi} \in \Delta_S$, called the γ-weighted occupancy distribution induced by policy π starting with the distribution $\nu \in \Delta_S$, such that:

$$\forall \nu \in \Delta_S, \int_S V(\pi, s)\nu(ds) = \frac{1}{1-\gamma} \mathbb{E}_{d_{\nu,\pi}}\left[R(X, \pi(X))\right] = \frac{1}{1-\gamma} \int_S R(s, \pi(s)) d_{\nu,\pi}(ds).$$

This result is important and will be used in one of our proofs. Explicit formulae (depending on ν, P, γ, π) of $d_{\nu,\pi}$ exist (Peters et al., 2005; Scherrer and Geist, 2014) but require more definitions and here the existence is sufficient to prove our results.

2.2 Differentiability

In this section, we give some brief definitions relative to differentiable functions. More properties needed for some of our proofs are given in the appendix A. Let $(X, \|.\|_X)$ and $(Y, \|.\|_Y)$ be two normed vector spaces and $f \in Y^X$ a function. We say that f is Fréchet-differentiable at the point $x \in X$ if there exists a linear and continuous function $Df(x) \in Y^X$ and a function $\epsilon \in Y^X$ such that:

$$\forall h \in X, f(x+h) = f(x) + Df(x)[h] + \|h\|_X \epsilon(h),$$

where $\lim_{\|h\|_X \to 0} \epsilon(h) = 0$. A function f is differentiable if it is differentiable for all $x \in X$. If f is linear and continuous then $\forall x \in X, Df(x) = f$.

Let $(H, \langle ., . \rangle_H)$ be an Hilbert space and $f \in \mathbb{R}^H$ a real and differentiable function. Let $x \in H$, $Df(x) \in \mathbb{R}^H$ is a linear form and by the Riez theorem, there exists a vector noted $\frac{\partial f(x)}{\partial h} \in H$ such that $Df(x) = \left\langle \frac{\partial f(x)}{\partial h}, . \right\rangle_H$. The function $\left\langle \frac{\partial f(x)}{\partial h}, . \right\rangle_H \in \mathbb{R}^H$ is such that $\forall y \in H, \left\langle \frac{\partial f(x)}{\partial h}, . \right\rangle_H (y) = \left\langle \frac{\partial f(x)}{\partial h}, y \right\rangle_H$. The vector $\frac{\partial f(x)}{\partial h}$ is called the gradient of f at x (where h is a dummy variable canonically associated to H).

3. Formal link between policy-gradient and continuous LSPI

In this section, we show that continuous PI is in fact an uphill method that maximizes the mean value. Therefore, it is closely related to the PG method. For sake of simplicity, we assume that all operations of integration and differentiation are licit.

The PG method, introduced by Sutton et al. (1999), consists in maximizing the mean value $J_\nu \in \mathbb{R}^{L^2}$, where $\nu \in \Delta_S$ and $\forall \pi \in L^2, J_\nu(\pi) = \int_S V(\pi, s)\nu(ds)$, by a gradient ascent. Thus, it consists in the following algorithm:

$$\pi_{k+1} = \pi_k + \alpha_k \frac{\partial J_\nu(\pi_k)}{\partial \pi},$$

where $\alpha_k \in \mathbb{R}_+^*$ is the gradient step. For now, we do not consider the problem of the evaluation of $\frac{\partial J_\nu(\pi_k)}{\partial \pi}$ which is key in all PG methods. A more general uphill algorithm which allows to maximize J_ν (Nocedal and Wright, 2006) is:

$$\pi_{k+1} = \pi_k + \alpha_k p_k,$$

where p_k is such that $\left\langle \frac{\partial J_\nu(\pi_k)}{\partial \pi}, p_k \right\rangle > 0$. In practice, PG methods do not consider a general set of policies such as L^2 but a set of parameterized policies (Sutton et al., 1999; Peters and Schaal, 2006; Silver et al., 2014). Therefore, the differentiation is not done over a policy but over a vector of parameters. Here, we are going to keep this more general kind of differentiation as it will highlight the link with continuous PI. With the more formal notations described in Secs. 2.2 and A.3, the update of continuous PI can be written:

$$\pi_{k+1} = \pi_k + \alpha_k \frac{\partial Q(\pi_k, ., \pi_k(.))}{\partial a},$$

where $\forall \pi \in L^2, \frac{\partial Q(\pi, ., \pi(.))}{\partial a} \in L^2$ is a function such that:

$$\forall s \in S, \frac{\partial Q(\pi, ., \pi(.))}{\partial a}(s) = \frac{\partial Q(\pi, s, \pi(s))}{\partial a}.$$

Therefore, if we manage to show that $\left\langle \frac{\partial J_\nu(\pi)}{\partial \pi}, \frac{\partial Q(\pi,.,\pi(.))}{\partial a} \right\rangle_{L^2} > 0$, it will imply that continuous PI is an uphill algorithm for the mean value J_ν. This will establish the link between maximizing the mean value by gradient ascent (the PG method) and the continuous PI approach.

To do so, we suppose that the mean value $J_\nu(\pi) = \int_S V(\pi, s)\nu(ds)$ has the following property:

$$\forall(\nu,\pi) \in \Delta_S \times L^2, DJ_\nu(\pi) = \left\langle \frac{\partial \int_S V(\pi,s)\nu(ds)}{\partial \pi}, . \right\rangle_{L^2} = \int_S \left\langle \frac{\partial V(\pi,s)}{\partial \pi}, . \right\rangle_{L^2} \nu(ds).$$

This property only implies that the permutation between the integral and differential operations is licit and it can be seen as a generalization of the Liebniz integral rule (Flanders, 1973). Then, thanks to Th. 1, we find an expression of $\frac{\partial J_\nu(\pi)}{\partial \pi}$ and thanks to Th. 2 we calculate $\left\langle \frac{\partial J_\nu(\pi)}{\partial \pi}, \frac{\partial Q(\pi,.,\pi(.))}{\partial a} \right\rangle_{L^2}$. It is important to note that Th. 1 is just another variant of the classical calculus of the gradient of the mean value with respect to the policy originally done by Sutton et al. (1999) for parameterized stationary policies in the discrete scenario and by Silver et al. (2014) for parameterized deterministic policies in the continuous scenario. However, here this calculus is done over a non-parameterized deterministic policy and a proof is given for sake of completeness.

Theorem 1 $DJ_\nu(\pi) = \left\langle \frac{\partial J_\nu(\pi)}{\partial \pi}, . \right\rangle_{L^2} = \frac{1}{1-\gamma} \int_S \left\langle \frac{\partial Q(\pi,s,\pi(s))}{\partial a}, . \right\rangle_A \circ \delta_s d_{\nu,\pi}(ds)$,

where $\forall s \in S, \delta_s \in A^{L^2}$ *is a function such that* $\forall \pi \in L^2, \delta_s(\pi) = \pi(s)$.

Proof The proof is provided in the appendix B. ■

Theorem 2 $\left\langle \frac{\partial J_\nu(\pi)}{\partial \pi}, \frac{\partial Q(\pi,.,\pi(.))}{\partial a} \right\rangle_{L^2} = DJ_\nu(\pi) \left[\frac{\partial Q(\pi,.,\pi(.))}{\partial a} \right] \geq 0$,

where the equality happens when $\frac{\partial Q(\pi,.,\pi(.))}{\partial a} = 0$ *almost everywhere (relatively to the measure* $d_{\nu,\pi}$*).*

Proof The proof is provided in the appendix C. ■

We conclude that the continuous PI method is indeed an uphill method to maximize the mean value which makes it a sound algorithm. However, to make it a practical batch RL algorithm, we still need to show how we can efficiently compute $\frac{\partial Q(\pi_k,.,\pi_k(.))}{\partial a}$ from a set of batch data $D_{RL} = (s_i, a_i, s'_i, r_i)_{i=1}^{N_{RL}}$ where $s'_i \sim P(.|s_i, a_i)$ which is done in the next section.

4. Toward practical algorithms

To provide a practical algorithm for the proposed continuous PI scheme, one has to estimate state-action value functions and to differentiate them respectively to the actions.

A natural approach consists in considering a linearly parameterized state-action value function $Q_\theta(s,a) = \theta^\top \phi(s,a)$, for some feature vector $\phi : \mathbb{R}^{p+q} \to \mathbb{R}^d$. For a given policy π_k, the parameter vector θ_k can be estimated using the LSTD algorithm (Bradtke and Barto, 1996). Notice that considering continuous actions in LSTD causes no problem. Then, the policy is updated with $\frac{\partial Q_\theta(s,a)}{\partial a} = \left(\frac{\partial \phi(s,a)}{\partial a}\right)^\top \theta$. If the components of the feature vector are

radial basis functions, for example, this gradient can be easily analytically computed. The computed policy after K iterations is then $\pi_K(s) = \left(\frac{\partial \phi(s,a)}{\partial a}\right)^\top \left(\sum_{k=1}^K \alpha_k \theta_k\right)$, each parameter vector θ_k being computed thanks to LSTD.

This approach requires a linear parameterization. A nonlinearly parameterized state-action value function could still be estimated by using an iterated projected fixed-point approach. For example, if the Q-function is represented as a neural network, the neural fitted-Q approach of Riedmiller (2005) could be easily adapted (this would amount to replace the Bellman optimality operator by the Bellman evaluation operator). Moreover, with such a neural network representation, the gradient respectively to the action can be efficiently computed using backpropagation.

In some case, it might be more natural to encode directly a policy, for example $\pi_\Theta(s) = \Theta^\top \phi(s)$, with $\Theta \in \mathbb{R}^{d \times q}$ a parameter matrix. The proposed approach is critic-based, but such a parameterization can be envisioned. Consider the advantage function $A(\pi, s, a) = Q(\pi, s, a) - V(\pi, s)$, it can be parameterized as $A_\Theta(s, a) = \langle a - \pi(s), \pi_\Theta(s) \rangle_A$. From this, we can parameterize a state-action value function by adding a linearly parameterized value function: $Q_{\Theta,w}(s, a) = A_\Theta(s, a) + w^\top \varphi(s)$. This Q-function can be estimated with LSTD and its gradient is simply $\frac{\partial Q_{\Theta,w}(s,a)}{\partial a} = \pi_\Theta(s)$. The policy computed after K iterations is then $\pi_K(s) = \left(\sum_{k=1}^K \alpha_k \Theta_k\right)^\top \phi(s)$, a linear mixture of the policies π_{Θ_k}. This is reminiscent of compatible function approximation for (natural) policy gradient with a critic and of the related parameterization of the advantage function, see for example Peters et al. (2005).

Alternatively, this last approach can be motivated by a first-order Taylor expansion of the state-action value function when the only varying parameter is the action a:

$$Q(\pi, s, a) = V(\pi, s) + \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, a - \pi(s) \right\rangle_A + \|a - \pi(s)\|_A \epsilon(a - \pi(s)),$$

where $\lim_{\|a-\pi(s)\|_A \to 0} \epsilon(a - \pi(s)) = 0$. Therefore, the policy π_Θ can alternatively be seen as a parameterization of the partial derivative of the state-action value function $\frac{\partial Q(\pi,s,\pi(s))}{\partial a}$. We have considered a linear parameterization for π_Θ, but as before, a nonlinear parameterization could be envisioned, by using an iterated projected fixed point approach.

5. Conclusion

We have shown that the natural adaptation of the Policy Iteration (PI) to the continuous scenario scheme, which consists in replacing the classical global greedy step by a local gradient improvement step, can be motivated by the close link it has to the policy gradient method. Indeed, continuous PI can be seen as an uphill method for the mean value as shown in Sec. 3. In addition, we sketch practical related algorithms, that rely mainly on LSTD or an iterated projected fixed point approach if no features are provides, which should make continuous-PI algorithms as easy to use as their discrete counterparts. Finally, as perspective, we would like to test those different algorithms on benchmark problems.

References

S. Bradtke and A. Barto. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 1996.

T. Degris, P. M Pilarski, and R.S. Sutton. Model-free reinforcement learning with continuous action in practice. In *Proc. of ACC*, pages 2177–2182, 2012.

D. Ernst, P. Geurts, and L. Wehenkel. Tree-based batch mode reinforcement learning. In *Journal of Machine Learning Research*, 2005.

Jeremy Fix and Matthieu Geist. Monte-carlo swarm policy search. In *Swarm and Evolutionary Computation*, pages 75–83. Springer, 2012.

H. Flanders. Differentiation under the integral sign. *The American Mathematical Monthly*, 80(6):615–627, 1973.

O. Hernández-Lerma and J.B. Lasserre. *Discrete-time Markov control processes*. Springer, 1996.

S. Kakade and J. Langford. Approximately optimal approximate reinforcement learning. In *Proc. of ICML*, volume 2, pages 267–274, 2002.

M.G. Lagoudakis and R. Parr. Least-squares policy iteration. *Journal of Machine Learning Research*, 2003.

A.Y. Ng and M. Jordan. Pegasus: A policy search method for large mdps and pomdps. In *Proc. of UAI*, pages 406–415. Morgan Kaufmann Publishers Inc., 2000.

Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

J. Peters and S. Schaal. Policy gradient methods for robotics. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2219–2225. IEEE, 2006.

J. Peters, S. Vijayakumar, and S. Schaal. Natural actor-critic. In *Proc. of ECML*, pages 280–291. Springer, 2005.

Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pages 317–328. Springer, 2005.

B. Scherrer and M. Geist. Local policy search in a convex space and conservative policy iteration as boosted policy search. In *Proc. of ECML*, pages 35–50. Springer, 2014.

D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller. Deterministic policy gradient algorithms. In *Proc. of ICML*, 2014.

R.S. Sutton, D.A. McAllester, S.P. Singh, Y. Mansour, et al. Policy gradient methods for reinforcement learning with function approximation. In *Proc. of NIPS*, volume 99, pages 1057–1063, 1999.

Appendix A. Differential Calculus

In this section, we recall some important properties of differential calculus that are used in our proofs.

A.1 Differential of a Composition

Let $(X, \|.\|_X), (Y, \|.\|_Y), (Z, \|.\|_Z)$ be three normed vector spaces, $f \in Y^X$ and $g \in Z^Y$. If f is differentiable at x and g is differentiable at $y = f(x)$, then $g \circ f \in Z^X$ is differentiable at x and :

$$Dg \circ f(x) = Dg(f(x)) \circ Df(x). \tag{2}$$

A.2 Differential of a multidimensional output

Let $((Y_i, \|.\|_{Y_i}))_i^n$ be a family of $n \in \mathbb{N}^*$ normed vector spaces, $(X, \|.\|_X)$ a normed vector space and $(f_i)_{i=1}^n$ a family of functions such that $f_i \in Y_i^X$ is differentiable at x, then the function $f = (f_1, \ldots, f_n) \in Y^X$, where $Y = \prod_{i=1}^n Y_i$, is differentiable at x and:

$$Df(x) = (Df_1(x), \ldots, Df_n(x)). \tag{3}$$

A.3 Differential of a multidimensional input form

Let $((H_i, \langle ., . \rangle_{H_i}))_{i=1}^n$ be a finite family of $n \in \mathbb{N}^*$ Hilbert spaces. Let $H = \prod_{i=1}^n H_i$ the cartesian product of the family $((H_i, \langle ., . \rangle_{H_i}))_{i=1}^n$. If for $x = (x_1, \ldots, x_n) \in H$ and $y = (y_1, \ldots, y_n) \in H$ we define $\langle ., . \rangle_H$ such that

$$\langle x, y \rangle = \sum_{i=1}^{n} \langle x_i, y_i \rangle_{H_i},$$

then $(H, \langle ., . \rangle_H)$ is an Hilbert space. Now, let consider $f \in \mathbb{R}^H$ differentiable. For $i \in (1, \ldots, n)$, we use the notation $x = (x_1, \ldots, x_n) = (x_{-i}, x_i)$ where $x_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$. For a fixed x_{-i}, we can define the function $f^{x_{-i}} \in \mathbb{R}^{H_i}$ such that:

$$\forall x_i \in H_i, f^{x_{-i}}(x_i) = f((x_{-i}, x_i)) = f(x).$$

As f is differentiable, $f^{x_{-i}}$ is differentiable:

$$Df^{x_{-i}}(x_i) = \left\langle \frac{\partial f^{x_{-i}}(x_i)}{\partial h_i}, . \right\rangle_{H_i}.$$

To simplify the notations, we write $\frac{\partial f(x)}{\partial h_i}$ in lieu of $\frac{\partial f^{x_{-i}}(x_i)}{\partial h_i}$. $\frac{\partial f(x)}{\partial h_i}$ is the partial derivative (or the partial gradient) of f evaluated in x with respect to the variable h_i canonically associated to the Hilbert space H_i. In addition, $Df(x)$ is such that:

$$Df(x) = \sum_{i=1}^{n} \left\langle \frac{\partial f(x)}{\partial h_i}, . \right\rangle_{H_i} \circ \delta_{H_i},$$

where $\delta_{H_i} \in \mathbb{R}^H$ is a function such that $\forall x = (x_1, \ldots, x_n), \delta_{H_i}(x) = x_i$.

Appendix B. Proof of Th. 1

Proof We recall that:

$$DV^s(\pi) = \left\langle \frac{\partial V(\pi, s)}{\partial \pi}, . \right\rangle_{L^2},$$

where $\forall s \in S, V^s \in S^{L^2}$ such that $\forall \pi \in L^2, V^s(\pi) = V(\pi, s)$. As, for all $s \in S$, we have $V(\pi, s) = Q(\pi, s, \pi(s))$ then:

$$V^s = Q \circ (I_{L^2}, I_s, \delta_s),$$

where $I_s \in S^{L^2}$ is such that $I_s(\pi) = s$ and $I_{L^2} \in L^{2^{L^2}}$ is such that $I_{L^2}(\pi) = \pi$.

Using Eq. (2) and Eq. (3), we have:

$$DV^s(\pi) = DQ(\pi, s, \pi(s)) \circ (DI_{L^2}(\pi), DI_s(\pi), D\delta_s(\pi)).$$

As I_s is a constant and I_{L^2} and δ_s are linear and continuous, then:

$$DV^s(\pi) = DQ(\pi, s, \pi(s)) \circ (I_{L^2}, 0, \delta_s).$$

We also recall that:

$$DQ(\pi, s, \pi(s)) = \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial \pi}, . \right\rangle_{L^2} \circ \delta_{L^2} + \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial s}, . \right\rangle_{S} \circ \delta_S + \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, . \right\rangle_{A} \circ \delta_A.$$

Thus, we have:

$$DV^s(\pi) = \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial \pi}, . \right\rangle_{L^2} + \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, . \right\rangle_{A} \circ \delta_s.$$

We recall that:

$$DQ^{(s,\pi(s))}(\pi) = \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial \pi}, . \right\rangle_{L^2},$$

where $\forall (s, a) \in S \times A, Q^{s,a} \in \mathbb{R}^{L^2}$ is a function such that $\forall \pi \in L^2, Q^{(s,a)}(\pi) = Q(\pi, s, a)$. Moreover:

$$Q^{(s,\pi(s))} = R \circ (I_s, I_a) + \gamma J_{P(.|s,\pi(s))},$$

where $\forall a \in A, I_a \in A^{L^2}$ is such that $\forall \pi \in L^2, I_a(\pi) = a$. Using Eq. (2), Eq. (3) and the fact that I_a and I_s are constant, we have:

$$\begin{aligned} DQ^{(s,\pi(s))}(\pi) &= DR(s, \pi(s)) \circ (0, 0) + \gamma \int_S \left\langle \frac{\partial V(\pi, s)}{\partial \pi}, . \right\rangle_{L^2} P(ds'|s, \pi(s)), \\ &= \gamma \int_S \left\langle \frac{\partial V(\pi, s')}{\partial \pi}, . \right\rangle_{L^2} P(ds'|s, \pi(s)). \end{aligned}$$

Thus:

$$DV^s(\pi) = \left\langle \frac{\partial V(\pi, s)}{\partial \pi}, . \right\rangle_{L^2} = \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, . \right\rangle_{A} \circ \delta_s + \gamma \int_S \left\langle \frac{\partial V(\pi, s')}{\partial \pi}, . \right\rangle_{L^2} P(ds'|s, \pi(s)),$$

$$= \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, . \right\rangle_{A} \circ \delta_s + \mathbb{E}_{P(.|s,\pi(s))} \left[DV^X(\pi) \right].$$

So, $DV^s(\pi)$ verifies a Bellman equation (1) which implies that:

$$\forall \nu \in \Delta_S, \mathbb{E}_\nu[DV^X(\pi)] = \int_S DV^s(\pi)\nu(ds) = \frac{1}{1-\gamma}\int_S \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, . \right\rangle_A \circ \delta_s d_{\nu,\pi}(ds).$$

And as :

$$\begin{aligned} DJ_\nu(\pi) &= \left\langle \frac{\partial \int_S V(\pi, s)\nu(ds)}{\partial \pi}, . \right\rangle_{L^2} = \int_S \left\langle \frac{\partial V(\pi, s)}{\partial \pi}, . \right\rangle_{L^2} \nu(ds), \\ &= \int_S DV^s(\pi)\nu(ds) = \mathbb{E}_\nu[DV^X(\pi)], \end{aligned}$$

We have the final result:

$$DJ_\nu(\pi) = \frac{1}{1-\gamma}\int_S \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, . \right\rangle_A \circ \delta_s d_{\nu,\pi}(ds) = \frac{1}{1-\gamma}\mathbb{E}_{d_{\nu,\pi}}\left[\left\langle \frac{\partial Q(\pi, X, \pi(X))}{\partial a}, . \right\rangle_A \circ \delta_X \right].$$

■

Appendix C. Proof of Th. 2

Proof It follows directly from Th. 1. Indeed :

$$\begin{aligned} DJ_\nu(\pi)\left[\frac{\partial Q(\pi, ., \pi(.))}{\partial a}\right] &= \frac{1}{1-\gamma}\int_S \left\langle \frac{\partial Q(\pi, s, \pi(s))}{\partial a}, \frac{\partial Q(\pi, s, \pi(s))}{\partial a} \right\rangle_A d_{\nu,\pi}(ds), \\ &= \frac{1}{1-\gamma}\int_S \left\| \frac{\partial Q(\pi, s, \pi(s))}{\partial a} \right\|_A^2 d_{\nu,\pi}(ds), \\ &= \frac{1}{1-\gamma}\mathbb{E}_{d_{\nu,\pi}}\left[\left\| \frac{\partial Q(\pi, X, \pi(X))}{\partial a} \right\|_A^2 \right] \geq 0. \end{aligned}$$

Clearly, the equality case happens when $\frac{\partial Q(\pi, ., \pi(.))}{\partial a}$ is almost everywhere null (relatively to $d_{\nu,\pi}$). ■