Stratégies d'échantillonnage pour l'apprentissage par renforcement batch

Raphael Fonteneau [1], **Susan A. Murphy** [2], **Louis Wehenkel** [1], **Damien Ernst** [1]

1. *Université de Liège, Belgique*
   {raphael.fonteneau,l.wehenkel,dernst}@ulg.ac.be
2. *Université du Michigan, USA*
   samurphy@umich.edu

RÉSUMÉ. *Cet article présente deux stratégies d'échantillonnage dans le contexte de l'apprentissage par renforcement en mode "batch". La première stratégie repose sur l'idée que les expériences susceptibles de mener à une modification de la politique de décision courante sont particulièrement informatives. Etant donné a priori un algorithme d'inférence de politiques de décision ainsi qu'un modèle prédictif du système, une expérience est réalisée si, étant donné le modèle prédictif, cette expérience mène à l'apprentissage d'une politique de décision différente. La deuxième stratégie exploite des résultats récemment publiés pour calculer des bornes sur le retour des politiques de décision de manière à sélectionner des expériences améliorant la précision des bornes afin de discriminer les politiques non-optimales. Ces deux stratégies sont illustrées sur des problèmes élémentaires et les résultats obtenus sont prometteurs.*

ABSTRACT. *We propose two strategies for experiment selection in the context of batch mode reinforcement learning. The first strategy is based on the idea that the most interesting experiments to carry out at some stage are those that are the most liable to falsify the current hypothesis about the optimal control policy. We cast this idea in a context where a policy learning algorithm and a model identification method are given a priori. The second strategy exploits recently published methods for computing bounds on the return of control policies from a set of trajectories in order to sample the state-action space so as to be able to discriminate between optimal and non-optimal policies. Both strategies are experimentally validated, showing promising results.*

MOTS-CLÉS : *apprentissage par renforcement, apprentissage actif, contrôle optimal.*

KEYWORDS: *reinforcement learning, active learning, optimal control.*

DOI:10.3166/RIA.27.171-194

1. Introduction

De nombreux problèmes de décision dans les domaines de l'ingénierie (Riedmiller, 2005), de la finance (Ingersoll, 1987), de la médecine (Murphy, 2003 ; 2005) ou de l'intelligence artificielle (Sutton, Barto, 1998) peuvent être formalisés comme des problèmes de contrôle optimal, dont l'objectif est de déterminer une politique de décision menant à l'optimisation d'un critère numérique. Souvent, ces problèmes sont abordés avec peu de connaissances sur la dynamique du système et la fonction de récompense qui les définissent.

Différentes approches ont déjà été proposées pour calculer des solutions approchées à ces problèmes dans le cas où les informations disponibles sont données sous forme d'un ensemble de transitions du système. Chacune de ces transitions est constituée d'un état, d'une décision prise dans cet état, de la valeur de la fonction de récompense et de la dynamique associées à ce couple état-décision. En particulier, une branche issue de l'apprentissage par renforcement (RL, *Reinforcement Learning*) - dont le but initial était la mise au point d'agents intelligents autonomes - aborde spécifiquement ce problème, que l'on désigne par BMRL par la suite (*Batch Mode RL*) (Fonteneau, 2011). Cette catégorie de problèmes se rencontre notamment en médecine (trajectoires de patients sous traitements médicaux), en marketing (historiques de clients) ou en finance (historiques de valeurs).

Etant donné un algorithme de type BMRL (c'est-à-dire capable d'apprendre une politique de décision dans un contexte BMRL), on s'intéresse dans cet article au problème de la génération d'ensembles de transitions à partir desquels l'algorithme BMRL puisse apprendre des politiques de décision performantes. Dans cet article, deux stratégies sont présentées :

– La première stratégie, initialement proposée par (Fonteneau *et al.*, 2011a) fait appel à un modéle prédictif (PM, *Predictive Model*) permettant d'estimer, à partir des transitions déjà disponibles, la dynamique du système et la fonction de récompense en tout point de l'espace état-décision. Le choix d'échantillonner une transition du système en un couple état-décision se fait si, considérant la valeur prédite par le modèle PM en ce couple, on observe une modification de la politique de décision calculée par l'algorithme BMRL. En pratique, cette stratégie consiste donc à chercher un couple état-décision pour lequel on prédit une modification de la politique de décision courante.

– La deuxième stratégie, esquissée par (Fonteneau *et al.*, 2010), est basée sur une méthode permettant de calculer, pour un ensemble de politiques de décision, des bornes sur les retours des politiques de décision. Le principe de l'approche est de déterminer des zones d'échantillonnage supposées améliorer la précision de ces bornes. Cette approche est fondée sur des hypothèses de continuité Lipschitzienne de la dynamique et de la fonction de récompense.

La suite de l'article s'organise de la manière suivante: après avoir formalisé le problème d'échantillonnage dans le contexte BMRL en section 2, on décrit et illustre expérimentalement les deux stratégies d'échantillonnage proposées dans les sections

3 et 4, respectivement. La section 5 propose une discussion de travaux connexes, et la section 6 conclut l'article.

2. Formalisation du problème

On considère un système déterministe à temps discret dont la dynamique stationnaire est donnée par l'équation

$$x_{t+1} = f(x_t, u_t) \quad t = 0, 1, \ldots, T-1,$$

où, pour tout $t \in \{0, \ldots, T-1\}$, l'état x_t est un élément d'un espace d'état normé borné $(\mathcal{X}, \|.\|_{\mathcal{X}})$ et u_t est un élément d'un espace de décision fini $\mathcal{U} = \left\{d^1, \ldots, d^m\right\}$ avec $m \in \mathbb{N}_0$. $T \in \mathbb{N}_0$ désigne l'horizon d'optimisation supposé fini. Une récompense instantanée

$$r_t = \rho(x_t, u_t) \in \mathbb{R}$$

est associée à une décision $u_t \in \mathcal{U}$ prise dans un état $x_t \in \mathcal{X}$. L'état initial du système $x_0 \in \mathcal{X}$ est supposé connu. Etant donnée une séquence de décisions $\mathbf{u} = (u_0, \ldots, u_{T-1}) \in \mathcal{U}^T$, on définit le retour $J^{\mathbf{u}}(x_0)$ de la séquence $\mathbf{u}$ à partir de x_0:

DÉFINITION 1 (Retour d'une politique de décision). —

$$\forall \mathbf{u} \in \mathcal{U}^T, \quad J^{\mathbf{u}}(x_0) = \sum_{t=0}^{T-1} \rho(x_t, u_t)$$

avec $x_{t+1} = f(x_t, u_t), \forall t \in \{0, \ldots, T-1\}$.

DÉFINITION 2 (Politiques de décision optimales). — *On note* $J^*(x_0)$ *la valeur maximale de* $J^{\mathbf{u}}(x_0)$*:*

$$J^*(x_0) = \max_{\mathbf{u} \in \mathcal{U}^T} J^{\mathbf{u}}(x_0) \ .$$

Une séquence de décisions $\mathbf{u}^*$ *est optimale si*

$$J^{\mathbf{u}^*}(x_0) = J^*(x_0) \ .$$

On appelle "transition du système" un quadruplet

$$(x, u, \rho(x, u), f(x, u)) \in \mathcal{X} \times \mathcal{U} \times \mathbb{R} \times \mathcal{X}$$

qui rassemble les valeurs des fonctions f et ρ en un couple (x, u) de l'espace conjoint $\mathcal{X} \times \mathcal{U}$. Les algorithmes BMRL (Ormoneit, Sen, 2002 ; Ernst *et al.*, 2005 ; Riedmiller, 2005 ; Fonteneau, 2011) ont été introduits afin d'inférer des lois de contrôle quasi optimales à partir d'un ensemble de transitions du système

$$\mathcal{F}_n = \left\{\left(x^l, u^l, r^l, y^l\right)\right\}_{l=1}^{n}$$

où $r^l = \rho(x^l, u^l)$ et $y^l = f(x^l, u^l), \quad \forall l \in \{1, \ldots, n\}$. Dans la suite de cet article, on désigne par $BMRL$ un algorithme générique de type BMRL et on note $BMRL(\mathcal{F}_n, x_0)$ la politique de décision calculée par cet algorithme.

Cet article propose une stratégie d'échantillonnage dont l'objectif est d'acquérir un ensemble de transitions $\mathcal{F}_n$ de taille maximale $N_{\max} \in \mathbb{N}$ (c'est à dire $n \leq N_{\max}$) [1], à partir duquel une politique de décision de qualité $\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}} \in \mathcal{U}^T$ puisse être apprise par $BMRL$, c'est-à-dire telle que $J^{\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}}}(x_0)$ soit aussi proche que possible de $J^*(x_0)$. Dans les sections suivantes, on propose deux stratégies pour aborder ce problème.

3. Première approche: modification de la politique de décision courante

Une première description générique de l'approche est donnée en section 3.1, tandis qu'une instanciation basée sur une méthode de plus proche voisin est proposée en section 3.2. Une étude expérimentale menée sur le problème "car-on-the-hill" est donnée à la section 3.3.

3.1. Description générique de l'approche

Cette approche est motivée par deux constatations: d'une part, si l'ajout d'une nouvelle transition dans l'ensemble des transitions disponibles provoque une modification du résultat calculé par l'algorithme BMRL, alors cette transition est très probablement informative ; d'autre part, la mise au point d'un modèle PM à partir des données disponibles peut se faire simplement pour un grand nombre de problèmes. Partant de ces deux constatations, la stratégie développée dans cette section: (i) explore de manière itérative un ensemble de couples état-décision, (ii) calcule pour chacun de ces couples la valeur prédite de la transition en utilisant le modèle PM, et (iii) analyse l'influence de la transition prédite ajoutée aux transitions déjà disponibles sur la solution calculée par l'algorithme BMRL. Le résultat de cette analyse est utilisé afin de (iv) sélectionner un couple état-décision pour lequel une modification de la politique de décision calculée par l'algorithme BMRL est prédite.

Etant donné un algorithme $BMRL$, un modèle prédictif PM qui, à un triplet $(\mathcal{F}_n, x, u)$, associe un quadruplet $(x, u, \hat{r}_{\mathcal{F}_n}(x, u), \hat{y}_{\mathcal{F}_n}(x, u))$ et une suite d'entiers $(L_n)_n$, on procède itérativement pour chaque $n < N_{\max}$:

– A partir de l'ensemble $\mathcal{F}_n = \left\{\left(x^l, u^l, r^l, y^l\right)\right\}_{l=1}^n$ des transitions préalablement générées, on calcule une politique de décision

$$\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}} = BMRL(\mathcal{F}_n, x_0) \quad ;$$

– On tire au hasard un couple $(x, u) \in \mathcal{X} \times \mathcal{U}$ selon une distribution uniforme $p_{\mathcal{X}\times\mathcal{U}}(\cdot)$ sur l'espace $\mathcal{X} \times \mathcal{U}$;

1. Typiquement, dans les applications actuelles du BMRL, $N_{\max}$ est de l'ordre de la dizaine de milliers.

– A partir de $\mathcal{F}_n$ et du modèle PM, on calcule une "transition prédite" :

$$(x, u, \hat{r}_{\mathcal{F}_n}(x,u), \hat{y}_{\mathcal{F}_n}(x,u)) = PM(\mathcal{F}_n, x, u)$$

et on construit l'"ensemble prédit":

$$\hat{\mathcal{F}}_{n+1}(x,u) = \mathcal{F}_n \cup \{(x, u, \hat{r}_{\mathcal{F}_n}(x,u), \hat{y}_{\mathcal{F}_n}(x,u))\} \ ,$$

que l'on utilise pour calculer une "politique de décision prédite" :

$$\hat{\mathbf{u}}^*_{\hat{\mathcal{F}}_{\mathbf{n+1}}(\mathbf{x},\mathbf{u})} = BMRL(\hat{\mathcal{F}}_{n+1}(x,u), x_0) \quad ;$$

- Si $\hat{\mathbf{u}}^*_{\hat{\mathcal{F}}_{\mathbf{n+1}}(\mathbf{x},\mathbf{u})} \neq \tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}}$, on considère que f et ρ méritent d'être échantillonnées en (x,u), ce que l'on réalise; on obtient $\left(x^{n+1}, u^{n+1}, r^{n+1}, y^{n+1}\right)$ avec $x^{n+1} = x$, $u^{n+1} = u$, $r^{n+1} = \rho(x,u)$ et $y^{n+1} = f(x,u)$, et on ajoute cette nouvelle transition à l'ensemble courant :

$$\mathcal{F}_{n+1} = \mathcal{F}_n \cup \left\{\left(x^{n+1}, u^{n+1}, r^{n+1}, y^{n+1}\right)\right\} \quad ;$$

- Si $\hat{\mathbf{u}}^*_{\hat{\mathcal{F}}_{\mathbf{n+1}}(\mathbf{x},\mathbf{u})} = \tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}}$, on tire un autre couple (x', u') selon $p_{\mathcal{X}\times\mathcal{U}}(\cdot)$ et on itère le processus ;

- Si L_n couples ont été testés sans mener à une modification de la politique de décision courante, on tire un couple $\left(x^{n+1}, u^{n+1}\right)$ au hasard selon $p_{\mathcal{X}\times\mathcal{U}}(\cdot)$, et on ajoute la transition $\left(x^{n+1}, u^{n+1}, \rho(x^{n+1}, u^{n+1}), f(x^{n+1}, u^{n+1})\right)$ à $\mathcal{F}_n$.

Influence de l'algorithme $BMRL$ et du modèle PM. Pour obtenir de bons résultats, il est nécessaire que les capacités d'inférence de $BMRL$ soient les meilleures possibles. En général, les algorithmes BMRL utilisent des approximateurs de fonctions (Busoniu *et al.*, 2010) dont le but est de décrire soit le système lui-même (f et ρ), soit des fonctions de valeur état-décision, soit des politiques de décision. Etant donné qu'ici, pour chaque itération de l'algorithme, la seule connaissance disponible sur le problème est un ensemble de transitions du système, on suggère d'utiliser un algorithme BMRL faisant appel à des approximateurs non paramétriques, comme des méthodes du type "plus proche(s) voisin(s)" ou à base d'arbres.

Le meilleur modèle prédictif PM envisageable est un algorithme qui, pour chaque couple état-décision (x,u), renvoie une prédiction égale à $(x, u, \rho(x,u), f(x,u))$. Prédire avec précision $\rho(x,u)$ et $f(x,u)$ peut s'avérer difficile, en particulier lorsque n est petit. On peut dès lors envisager de travailler avec des ensembles de prédictions, ce qui d'un côté augmente la probabilité de détecter un couple susceptible de mener à une modification de la politique de décision courante, mais de l'autre, augmente également la probabilité que le couple choisi ne mène à aucune modification réelle ("fausses" prédictions). Si des connaissances a priori sur les fonctions f et ρ sont disponibles, on peut construire des ensembles de transitions "compatibles" avec ces

connaissances ainsi qu'avec les transitions précédemment acquises (voir par exemple (Fonteneau *et al.*, 2011b) où des connaissances de continuité Lipschitzienne sont exploitées). Ces connaissances peuvent ainsi être utilisées pour augmenter la précision de PM.

Influence de la suite $(L_n)_n$. Le $n-$ième terme de la suite $(L_n)_n$ définit le nombre maximal d'essais autorisés pour identifier une transition pour laquelle on prédit une modification de la politique de décision lorsque n transitions ont déjà été collectées. La valeur du terme L_n devrait être choisie de manière à assurer que, à la $n-$ième itération, s'il existe un couple état-décision pour lequel la transition correspondante mène à une modification de la politique courante, alors ce couple devrait être identifié avec une grande probabilité. Il peut cependant arriver que, pour certaines itérations n, il n'existe pas de transition (prédite) menant à une modification de la politique de décision (prédite). Dans un tel cas, L_n essais seront tout de même réalisés, ce qui peut être gênant en termes de temps de calcul si L_n est grand. Le choix des valeurs des termes de la suite $(L_n)_n$ résulte donc d'un compromis entre la volonté d'identifier avec grande probabilité les transitions menant à une modification de la politique de décision, et le besoin de limiter les temps de calcul lorsqu'il n'y a rien à identifier. Intuitivement, il semble raisonnable de penser qu'une telle suite devrait être croissante car, lorsque n augmente, on s'attend à ce que la politique courante tende vers l'optimalité et que le modèle prédictif PM devienne de plus en plus précis, et donc on s'attend aussi à ce qu'il soit de plus en plus complexe de prédire des modifications.

3.2. Implémentation avec une méthode du plus proche voisin

Dans cette section, on introduit les algorithmes $BMRL$ et PM utilisés pour illustrer notre stratégie d'échantillonnage dans les expériences détaillées en section 3.3. L'algorithme $BMRL$ fonctionne en approximant les fonctions f et ρ à partir des transitions disponibles puis en résolvant de manière exacte le problème de contrôle optimal défini par ces fonctions approchées. Cet algorithme est détaillé en section 3.2.1. En section 3.2.2, on détaille l'algorithme PM, qui fonde ses prédictions sur les approximations de f et ρ utilisées par $BMRL$.

3.2.1. Choix de l'algorithme $BMRL$

RL basé sur l'apprentissage d'un modèle. Le RL basé sur l'apprentissage d'un modèle consiste à résoudre de manière approchée un problème de contrôle optimal en approximant les fonctions inconnues f et ρ et en résolvant le problème de contrôle optimal "approché" défini par les approximations de f et ρ. Les valeurs y^l (resp. r^l) de la fonction f (resp. ρ) en $(x^l, u^l), l = 1 \ldots n$ sont utilisées pour apprendre une fonction $\tilde{f}_{\mathcal{F}_n}$ (resp. $\tilde{\rho}_{\mathcal{F}_n}$) définie sur l'espace $\mathcal{X} \times \mathcal{U}$. Le problème de contrôle optimal approché défini par les fonctions $\tilde{f}_{\mathcal{F}_n}$ et $\tilde{\rho}_{\mathcal{F}_n}$ est résolu et sa solution est utilisée comme solution approchée du problème de contrôle optimal defini par les "vraies" fonctions f et ρ.

Etant donnée une séquence de décisions $\mathbf{u} \in \mathcal{U}^T$ et un algorithme de type BMRL par apprentissage de modèle, on note $\tilde{J}^{\mathbf{u}}_{\mathcal{F}_n}(x_0)$ le retour approché de la séquence de décisions $\mathbf{u}$, c'est-à-dire le retour obtenu en considérant les approximations $\tilde{f}_{\mathcal{F}_n}$ et $\tilde{\rho}_{\mathcal{F}_n}$:

$$\forall \mathbf{u} \in \mathcal{U}^T, \tilde{J}^{\mathbf{u}}_{\mathcal{F}_n}(x_0) = \sum_{t=0}^{T-1} \tilde{\rho}_{\mathcal{F}_n}\left(\tilde{x}_t, u_t\right)$$

avec

$$\tilde{x}_{t+1} = \tilde{f}_{\mathcal{F}_n}\left(\tilde{x}_t, u_t\right), \ \forall t \in \{0, \ldots, T-1\}$$

et

$$\tilde{x}_0 = x_0 \ .$$

On note $\tilde{J}^{*}_{\mathcal{F}_n}(x_0)$ le retour approché maximal au départ de l'état initial $x_0 \in \mathcal{X}$ et selon les approximations $\tilde{f}_{\mathcal{F}_n}$ et $\tilde{\rho}_{\mathcal{F}_n}$:

$$\tilde{J}^{*}_{\mathcal{F}_n}(x_0) = \max_{\mathbf{u} \in \mathcal{U}^T} \tilde{J}^{\mathbf{u}}_{\mathcal{F}_n}(x_0) \ .$$

En utilisant ces notations, les algorithmes BMRL par apprentissage de modèle calculent une séquence de décisions $\tilde{\mathbf{u}}^{*}_{\mathcal{F}_\mathbf{n}} \in \mathcal{U}^T$ telle que $\tilde{J}^{\tilde{\mathbf{u}}^{*}_{\mathcal{F}_\mathbf{n}}}_{\mathcal{F}_n}(x_0)$ soit le plus proche possible de (idéalement, égal à) $\tilde{J}^{*}_{\mathcal{F}_n}(x_0)$. Ces algorithmes supposent implicitement qu'une politique de décision destinée au modèle appris mène aussi à un retour élevé pour le vrai modèle.

Partition de Voronoi. On spécifie dans cette section l'algorithme BMRL par apprentissage de modèle utilisé par la suite dans les simulations. Cet algorithme approxime f et ρ en utilisant des fonctions constantes par morceaux sur une partition de type Voronoi (Aurenhammer, 1991) de l'espace $\mathcal{X} \times \mathcal{U}$ (ce qui correspond à une approximation du plus proche voisin). L'algorithme est noté VRL (pour Voronoi RL) dans la suite. Etant donné un état initial $x_0 \in \mathcal{X}$, l'algorithme VRL retourne une séquence de décisions en boucle ouverte correspondant à un "déplacement optimal" parmi les cellules de Voronoi.

Tout d'abord, on fait l'hypothèse que les couples de l'ensemble $\left\{(x^l, u^l)\right\}_{l=1}^{n}$ donnés par $\mathcal{F}_n$ sont distincts deux à deux :

$$\forall l, l' \in \{1, \ldots, n\}, \quad (x^l, u^l) = (x^{l'}, u^{l'}) \implies l = l' \ .$$

On fait également l'hypothèse que chaque décision de l'espace $\mathcal{U}$ a été prise au moins une fois lors de la génération des transitions incluses dans $\mathcal{F}_n$:

$$\forall u \in \mathcal{U}, \quad \exists l \in \{1, \ldots, n\}, u^l = u \ .$$

Le modèle appris se base sur n cellules de Voronoi $\left\{V^l\right\}_{l=1}^n$ définissant une partition de taille n de l'espace $\mathcal{X} \times \mathcal{U}$. La cellule V^l associée au couple (x^l, u^l) est définie comme l'ensemble des couples $(x, u) \in \mathcal{X} \times \mathcal{U}$ tels que :

$$(i) \quad u = u^l \ , \tag{1}$$

$$(ii) \quad l \in \arg\min_{l':u^{l'}=u} \left\{ \|x - x^{l'}\|_{\mathcal{X}} \right\} \ , \tag{2}$$

$$(iii) \quad l = \min_{l'} \left\{ l' \in \arg\min_{l':u^{l'}=u} \left\{ \|x - x^{l'}\|_{\mathcal{X}} \right\} \right\} . \tag{3}$$

$\left\{V^l\right\}_{l=1}^n$ forme bien une partition de $\mathcal{X} \times \mathcal{U}$ puisque l'on peut aisément vérifier que chaque couple $(x, u) \in \mathcal{X} \times \mathcal{U}$ appartient à une et à une seule cellule de Voronoi. La fonction f (resp. ρ) est approximée par une fonction constante par morceaux $\tilde{f}_{\mathcal{F}_n}$ (resp. $\tilde{\rho}_{\mathcal{F}_n}$) de la manière suivante :

$$\begin{aligned} \forall l \in \{1, \ldots, n\}, \forall (x, u) \in V^l, \quad \tilde{f}_{\mathcal{F}_n}(x, u) &= y^l, \\ \tilde{\rho}_{\mathcal{F}_n}(x, u) &= r^l \ . \end{aligned}$$

A partir de $\tilde{f}_{\mathcal{F}_n}$ et $\tilde{\rho}_{\mathcal{F}_n}$, on définit une suite finie de fonctions de valeur approchées $\left(\tilde{Q}^*_{T-t}\right)_{t=0}^{T-1}$ comme suit : $\forall t \in \{0, \ldots, T-1\}$, $\forall (x, u) \in \mathcal{X} \times \mathcal{U}$,

$$\tilde{Q}^*_{T-t}(x, u) = \tilde{\rho}_{\mathcal{F}_n}(x, u) + \arg\max_{u' \in \mathcal{U}} \tilde{Q}^*_{T-t-1}\left(\tilde{f}_{\mathcal{F}_n}(x, u), u'\right) \ ,$$

avec

$$Q^*_1(x, u) = \tilde{\rho}_{\mathcal{F}_n}(x, u), \quad \forall (x, u) \in \mathcal{X} \times \mathcal{U}.$$

A partir de la suite de fonctions $\left(\tilde{Q}^*_{T-t}\right)_{t=0}^{T-1}$, on calcule une politique de décision en boucle ouverte

$$\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}} = (\tilde{u}^*_{\mathcal{F}_n,0}, \ldots, \tilde{u}^*_{\mathcal{F}_n,T-1}) \in \mathcal{U}^T$$

solution du problème de contrôle optimal approché, c'est-à-dire telle que

$$\tilde{J}^{\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}}}_{\mathcal{F}_n}(x_0) = \tilde{J}^*_{\mathcal{F}_n}(x_0) \quad ,$$

de la manière suivante :

$$\tilde{u}^*_{\mathcal{F}_n,0} \quad \in \quad \arg\max_{u' \in \mathcal{U}} \tilde{Q}^*_T(\tilde{x}^*_0, u') \ ,$$

et, $\forall t \in \{0, \ldots, T-2\}$,

$$\tilde{u}^*_{\mathcal{F}_n,t+1} \quad \in \quad \arg\max_{u' \in \mathcal{U}} \tilde{Q}^*_{T-(t+1)}\left(\tilde{f}_{\mathcal{F}_n}\left(\tilde{x}^*_t, \tilde{u}^*_{\mathcal{F}_n,t}\right), u'\right)$$

où

$$\tilde{x}_0^* = x_0$$

et

$$\tilde{x}_{t+1}^* = \tilde{f}_{\mathcal{F}_n}(\tilde{x}_t^*, \tilde{u}_{\mathcal{F}_n,t}^*), \forall t \in \{0, \ldots, T-1\}.$$

Toutes les fonctions de la suite $\left(\tilde{Q}_{T-t}^*\right)_{t=0}^{T-1}$ sont constantes dans chaque cellule, ce qui permet d'extraire facilement la politique de décision $\tilde{\mathbf{u}}_{\mathcal{F}_\mathbf{n}}^*$ en utilisant un algorithme de type Viterbi (Viterbi, 1967) dont la complexité est linéaire en fonction de n, T et la cardinalité m de l'espace $\mathcal{U}$. Une version tabulaire de l'algorithme VRL est donnée en figure 1. D'autre part, l'algorithme VRL possède des propriétés de consistance lorsque les fonctions f et ρ sont Lipschitziennes et que la dispersion de l'ensemble de transitions $\mathcal{F}_n$ converge vers 0.

3.2.2. Choix de l'algorithme PM

L'algorithme PM utilisé dans les simulations fait appel aux fonctions $\tilde{f}_{\mathcal{F}_n}$ et $\tilde{\rho}_{\mathcal{F}_n}$ calculées par l'algorithme VRL. Etant donné un ensemble de transitions $\mathcal{F}_n$ et un couple $(x, u) \in \mathcal{X} \times \mathcal{U}$, l'algorithme PM renvoie

$$(x, u, \hat{r}_{\mathcal{F}_n}(x, u), \hat{y}_{\mathcal{F}_n}(x, u)) = PM(\mathcal{F}_n, x, u)$$

tel que

$$\hat{r}_{\mathcal{F}_n}(x, u) = \tilde{\rho}_{\mathcal{F}_n}(x, u)$$

et

$$\hat{y}_{\mathcal{F}_n}(x, u) = \tilde{f}_{\mathcal{F}_n}(x, u)\ .$$

3.3. Résultats expérimentaux

Cette section illustre la stratégie d'échantillonnage décrite ci-dessus sur le problème jouet "car-on-the-hill" (Ernst, 2005), un problème classique souvent utilisé pour tester les algorithmes d'apprentissage par renforcement.

3.3.1. Présentation du problème jouet "car-on-the-hill"

Un point de masse unitaire − représentant un véhicule − doit être conduit au sommet d'une colline située à droite sur la figure 1 par application d'une force horizontale. Pour certains états initiaux du système, la puissance maximale du véhicule ne permet pas d'atteindre le sommet de la colline. Le véhicule doit donc grimper sur le flanc d'une autre colline située à gauche, puis la redescendre afin de prendre de la vitesse et atteindre le sommet de la colline de droite.

La dynamique du véhicule est donnée par l'équation différentielle :

$$\ddot{z} = \frac{1}{1 + \left(\frac{dH(z)}{dz}\right)^2}\left(\frac{u}{m_c} - g\frac{dH(z)}{dz} - \dot{z}^2\frac{dH(z)}{dz}\frac{d^2H(z)}{dz^2}\right)$$

Algorithme 1 L'algorithme VRL (de l'anglais Voronoi Reinforcement Learning). $Q_{T-t,l}$ est la valeur prise par la fonction $\tilde{Q}^*_{T-t}$ dans la cellule de Vornoi V^l.

Entrées: un état initial $x_0 \in \mathcal{X}$, un ensemble de transitions $\mathcal{F}_n = \left\{\left(x^l, u^l, r^l, y^l\right)\right\}_{l=1}^n$;
Sorties: une séquence de décisions $\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}}$ et $\tilde{J}^*_{\mathcal{F}_n}(x_0)$;
Initialisation:
Créer une matrice $n \times m$ V telle que $V(i,j)$ contient l'indice de la cellule de Voronoi (CV) où $\left(\tilde{f}_{\mathcal{F}_n}(x^i, u^i), d^j\right)$ se trouve ;
for $i = 1$ **to** n **do**
  $Q_{1,i} \leftarrow r^i$;
end for

Algorithme:
for $t = T-2$ **to** 0 **do**
  for $i = 1$ **to** n **do**
    $l \leftarrow \underset{l' \in \{1,\ldots,m\}}{\arg\max} \left\{Q_{T-t-1,V(i,l')}\right\}$;
    $Q_{T-t,i} \leftarrow r^i + Q_{T-t-1,V(i,l)}$;
  end for
end for
$l \leftarrow \underset{l' \in \{1,\ldots,m\}}{\arg\max}\ Q_{T,i'}$ où i' désigne l'indice de la CV où $(x_0, d^{l'})$ se trouve ;
$l_0^* \leftarrow$ indice de la CV où (x_0, d^l) se trouve ;
$\tilde{J}^*_{\mathcal{F}_n}(x_0) \leftarrow Q_{T,l_0^*}$;
$i \leftarrow l_0^*$;
$\tilde{u}^*_{\mathcal{F}_n,0} \leftarrow u^{l_0^*}$;
for $t = 0$ **to** $T-2$ **do**
  $l^*_{t+1} \leftarrow \underset{l' \in \{1,\ldots,m\}}{\arg\max} \left\{Q_{T-t-1,V(i,l')}\right\}$;
  $\tilde{u}^*_{\mathcal{F}_n,t+1} \leftarrow d^{l^*_{t+1}}$;
  $i \leftarrow V(i, l^*_{t+1})$;
end for
return $\tilde{\mathbf{u}}^*_{\mathcal{F}_\mathbf{n}} = \left(\tilde{u}^*_{\mathcal{F}_n,0}, \ldots, \tilde{u}^*_{\mathcal{F}_n,T-1}\right)$, $\tilde{J}^*_{\mathcal{F}_n}(x_0)$.

où $z \in [-1,1]$ est la position horizontale du vehicule (donnée en m), $\dot{z} \in [-3,3]$ est la vitesse du véhicule (donnée en m/s), $u \in \{-4,4\}$ est la force horizontale (donnée en N), $g = 9.81 m/s^2$ est la constante de gravité et H est le profil du terrain :

$$H(z) = \begin{cases} z^2 + z & \text{si} \quad z < 0\,, \\ \frac{z}{\sqrt{1+5z^2}} & \text{si} \quad z \geq 0\,. \end{cases}$$

La masse du véhicule vaut $m_c = 1kg$. La durée d'un pas de temps est $T_s = 0.1s$ et la dynamique à temps discret du véhicule f est obtenue par intégration de la dynamique à temps continu entre chaque pas de temps. L'espace de décision $\mathcal{U}$ contient deux

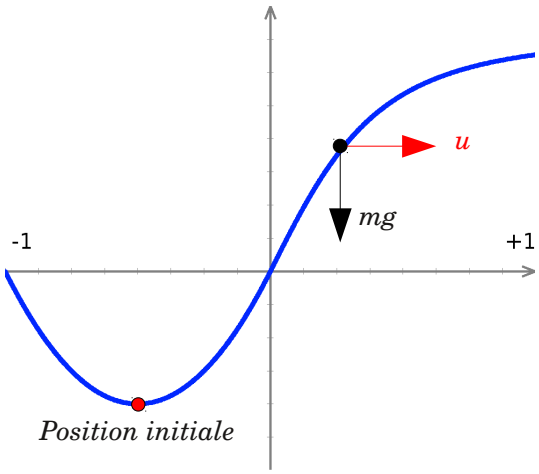

Figure 1. Illustration du problème-jouet "car-on-the-hill"

éléments: -4 et 4. Lorsque la position z ou la vitesse $\dot{z}$ dépasse les bornes, le véhicule atteint un état absorbant dans lequel il reste indépendamment de la décision prise. Si $z_{t+1} < -1$ ou si $|\dot{z}_{t+1}| > 3$, alors le véhicule atteint un état absorbant "perdant" s_{-1} et reçoit une récompense de -1 à chaque pas de temps jusqu'à $t = T-1$. Si $z_{t+1} \geq 1$ et $|\dot{z}_{t+1}| \leq 3$, alors le véhicule atteint un état absorbant "gagnant" s_1, et reçoit une récompense de $+1$ à chaque pas de temps jusqu'à $t = T-1$. Les états absorbants s_{-1} et s_{+1} sont supposés connus. L'espace d'état est donc égal à

$$\mathcal{X} = [-1, 1] \times [-3, 3] \cup \{s_1, s_{-1}\} \ .$$

L'objectif est de déterminer une séquence de décisions maximisant la somme des récompenses obtenues sur un horizon $T = 20$ lorsque le véhicule démarre au creux de la vallée en $x_0 = [-0.5, 0]$. Une telle séquence permettra aussi de mener le véhicule au sommet de la colline en une durée minimale.

L'algorithme VRL détaillé en section 3.2.1 ne donne pas d'information sur la manière de gérer les états absorbants. Cela peut être fait en ajoutant à l'ensemble de transitions $m \times n_{abs}$ "transitions artificielles", où n_{abs} désigne le nombre d'états absorbants du problème. Dans le cadre du problème "car-on-the-hill", cela se traduit par l'ajout de 4 transitions artificielles :

$$\big\{(s_1, 4, 1, s_1), (s_1, -4, 1, s_1), (s_{-1}, 4, -1, s_{-1}), (s_{-1}, -4, -1, s_{-1})\big\}.$$

La définition des cellules de Voronoi reste identique à celle donnée par les équations (1), (2) et (3) lorsque x^l n'est pas un état absorbant. Dans tous les autres cas, la norme $\|.\|_{\mathcal{X}}$ peut être (abusivement) étendue aux états absorbants de la manière suivante :

$$\|x - x^l\|_{\mathcal{X}} = \begin{cases} 0 & \text{si } x = x^l \ , \\ +\infty & \text{si } x \neq x^l \ . \end{cases}$$

3.3.2. Protocole expérimental

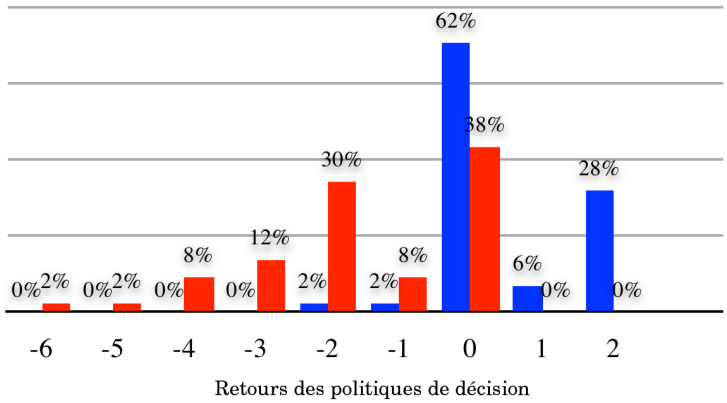

Figure 2. Distribution des retours des politiques de décision calculées à partir de $\mathcal{F}^k_{N_{\max}}\ ,\quad k=1\dots q$ *(en bleu, à gauche) et* $\mathcal{G}^k_{N_{\max}}\ , k=1\dots q$ *(en rouge, à droite)*

Les performances de notre stratégie d'échantillonnage sont comparées avec les performances d'une stratégie d'échantillonnage uniforme. Pour ce faire, on teste $q = 50$ fois notre stratégie d'échantillonnage, où chaque test $k = 1 \dots q$ est initialisé avec un ensemble de transitions $\mathcal{F}^k_m$ contenant $m = 2$ transitions (une transition pour chaque décision) :

$$\forall k \in \{1,\dots,q\}, \mathcal{F}^k_m = \quad \{(x_0,-4,\rho(x_0,-4),f(x_0,-4))\ , \\ (x_0,+4,\rho(x_0,+4),f(x_0,+4))\}\ .$$

Notre stratégie d'échantillonnage est mise en oeuvre sur chacun des ensembles de transitions $\mathcal{F}^k_m\ ,\quad k = 1\dots q$ jusqu'à ce que chacun contienne $N_{\max} = 1000$ transitions. On obtient ainsi q suites finies d'ensembles de transitions, chaque suite contenant $(N_{\max} - m + 1)$ termes :

$$\mathcal{F}^1_m, \mathcal{F}^1_{m+1}, \dots, \mathcal{F}^1_{N_{\max}}, \dots, \mathcal{F}^q_m, \mathcal{F}^q_{m+1}, \dots, \mathcal{F}^q_{N_{\max}}.$$

On génère également q suites finies d'ensembles de transitions contenant chacune $(N_{\max} - m + 1)$ termes

$$\mathcal{G}^1_m, \mathcal{G}^1_{m+1}, \dots, \mathcal{G}^1_{N_{\max}}, \dots, \mathcal{G}^q_m, \mathcal{G}^q_{m+1}, \dots, \mathcal{G}^q_{N_{\max}}$$

où, pour chaque $k = 1 \dots q$, et pour chaque $n = m \dots N_{\max} - 1$, chaque ensemble $\mathcal{G}^k_{n+1}$ est obtenu en ajoutant à $\mathcal{G}^k_n$ une transition $(x, u, \rho(x, u), f(x, u))$ telle que (x, u) est tiré selon $p_{\mathcal{X}\times\mathcal{U}}(\cdot)$. Les termes de la suite $(L_n)_n$ utilisée pour ces simulations sont définis de la manière suivante :

$$\forall n \in \{m,\dots,N_{\max}\}, L_n = mn\ .$$

La distribution de probabilités $p_{\mathcal{X}\times\mathcal{U}}(\cdot)$ est telle que la probabilité de tirer un couple état-décision (x, u) avec $x = s_1$ ou $x = s_{-1}$ est nulle, et uniforme ailleurs.

3.3.3. Analyse des résultats

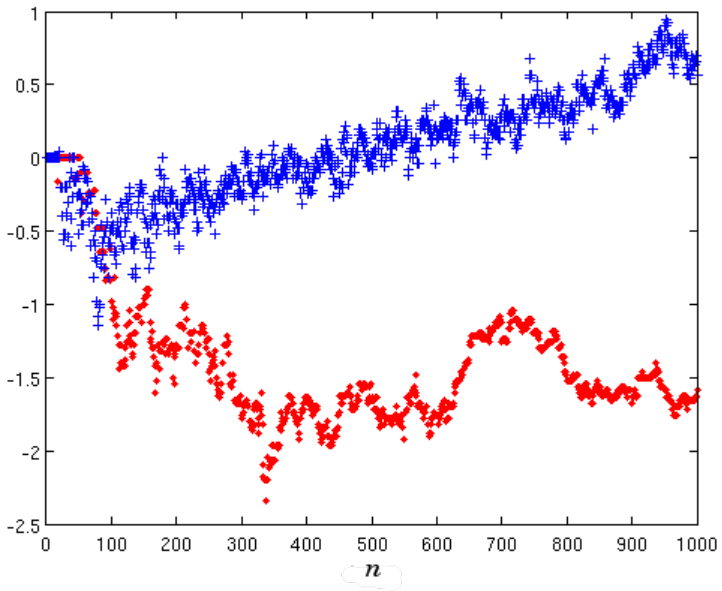

Figure 3. Evolution des performances moyennes de notre stratégie d'échantillonnage $\mathcal{M}(n)$ (croix bleues) comparée à l'évolution des performances moyennes d'un échantillonnage uniforme $M_{unif}(n)$ (points rouges)

Performances des politiques de décision calculées à partir des ensembles de $N_{\max}$ transitions. On calcule les retours des $2q$ politiques de décision calculées par l'algorithme VRL à partir des ensembles finaux contenant $N_{\max}$ transitions $\mathcal{F}^k_{N_{\max}}$ et $\mathcal{G}^k_{N_{\max}}$, $k = 1 \ldots q$. Les résultats, exprimés en termes de distribution des retours des politiques de décision apprises, sont donnés en figure 2.

On observe que l'algorithme VRL parvient à calculer, pour 28 % des ensembles de transitions obtenus à partir de notre stratégie d'échantillonnage, une politique de décision pour laquelle le retour vaut 2, alors qu'aucune politique de décision menant à un retour strictement positif n'est calculée à partir des bases de données générées par tirage uniforme. A titre informatif, il est nécessaire de générer des ensembles de 10 000 transitions si l'on souhaite obtenir, avec un tirage uniforme, des performances équivalentes.

Performances moyennes et distribution des retours des politiques de décision apprises. Pour une cardinalité donnée n ($m \leq n \leq N_{\max}$), on calcule la perfor-

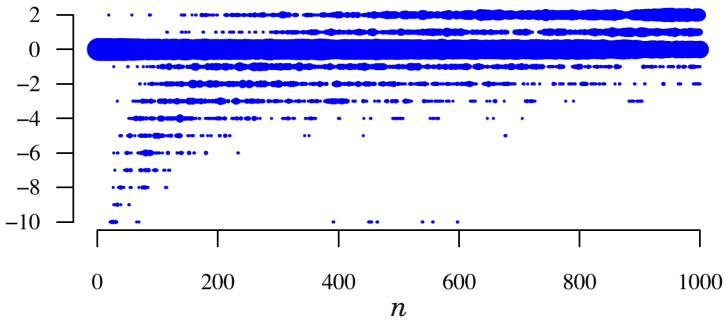

Figure 4. Distribution des retours des politiques de décision $\tilde{\mathbf{u}}^*_{\mathcal{F}^{\mathbf{k}}_{\mathbf{n}}}$ *,k=1 . . . q,* $n = m \ldots N_{\max}$*. Pour chaque valeur de* n*, la surface d'un disque correspondant à un retour* $r = -10 \ldots 2$ *est proportionnelle au nombre de politiques de l'ensemble* $\{\tilde{\mathbf{u}}^*_{\mathcal{F}^{\mathbf{k}}_{\mathbf{n}}}\}_{k=1}^{q}$ *pour lesquelles le retour vaut* r

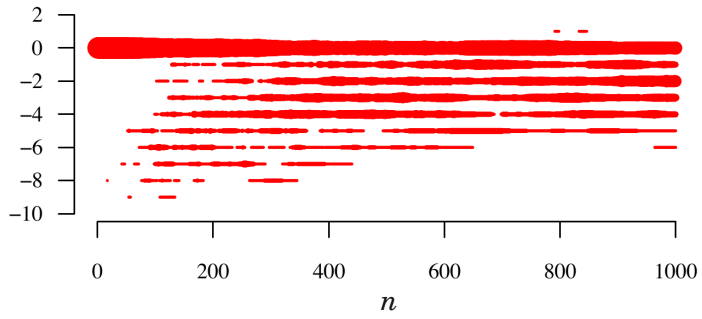

Figure 5. Distribution des retours des politiques de décision $\tilde{\mathbf{u}}^*_{\mathcal{G}^{\mathbf{k}}_{\mathbf{n}}}$ *,k=1 . . . q,* $n = m \ldots N_{\max}$*. Pour chaque valeur de* n*, la surface d'un disque correspondant à un retour* $r = -10 \ldots 2$ *est proportionnelle au nombre de politiques de l'ensemble* $\{\tilde{\mathbf{u}}^*_{\mathcal{G}^{\mathbf{k}}_{\mathbf{n}}}\}_{k=1}^{q}$ *pour lesquelles le retour vaut* r

mance moyenne $\mathcal{M}(n)$ des q séquences de décisions $\tilde{\mathbf{u}}^*_{\mathcal{F}^{\mathbf{k}}_{\mathbf{n}}}$,k=1 . . . q calculées par l'algorithme VRL à partir des ensembles de transitions $\mathcal{F}^k_n$, $k = 1 \ldots q$:

$$\mathcal{M}(n) = \frac{1}{q} \sum_{k=1}^{q} J^{\tilde{\mathbf{u}}^*_{\mathcal{F}^{\mathbf{k}}_{\mathbf{n}}}}(x_0) \; .$$

On calcule également la performance moyenne $\mathcal{M}_{unif}(n)$ des q séquences de décisions $\tilde{\mathbf{u}}^*_{\mathcal{G}^k_n}, k = 1 \ldots q$ calculées par l'algorithme VRL à partir des ensembles $\mathcal{G}^k_n, k = 1 \ldots q$ obtenus par échantillonnage uniforme :

$$\mathcal{M}_{unif}(n) = \frac{1}{q} \sum_{k=1}^{q} J^{\tilde{\mathbf{u}}^*_{\mathcal{G}^k_n}}(x_0) \, .$$

Les valeurs de $\mathcal{M}(n)$ et $\mathcal{M}_{unif}(n)$ pour $n = m \ldots N_{\max}$ sont comparées en figure 3. On donne également en figure 4 (resp. 5) la distribution des retours des politiques $\tilde{\mathbf{u}}^*_{\mathcal{F}^k_n}, k = 1 \ldots q, n = m \ldots N_{\max}$ (resp. $\tilde{\mathbf{u}}^*_{\mathcal{G}^k_n}, k = 1 \ldots q, n = m \ldots N_{\max}$).

On observe que, à partir de notre stratégie d'échantillonnage, des politiques menant à un retour de 2 sont apprises à partir d'ensembles contenant moins de 200 transitions. On remarque également qu'aucune politique menant à un retour de 2 n'a pu être apprise à partir des ensembles de transitions tirées uniformément $\mathcal{G}^k_n, k = 1 \ldots q, n = m \ldots N_{\max}$.

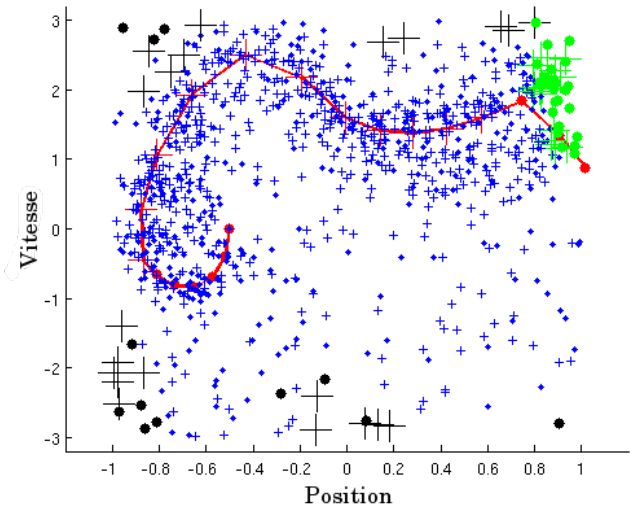

Figure 6. Représentation de l'ensemble de transitions $\mathcal{F}^1_{N_{\max}}$ (obtenu avec notre stratégie d'échantillonnage)

Représentations de $\mathcal{F}^1_{N_{\max}}$ et $\mathcal{G}^1_{N_{\max}}$. On représente graphiquement les transitions contenues dans l'ensemble $\mathcal{F}^1_{N_{\max}}$ (resp. $\mathcal{G}^1_{N_{\max}}$) en figure 6 (resp. 7). Chaque transition (x^l, u^l, r^l, y^l) est représentée par un symbole situé en $x^l = [z, \dot{z}]$. Un signe '+' indique que $u^l = +4$, tandis qu'un signe '•' indique que $u^l = -4$. Le symbole est bleu si $r^l = 0$. Des symboles plus grands et coloriés en noir (vert) sont utilisés si $r^l = -1$ ($r^l = 1$). La courbe rouge représente la trajectoire du véhicule conduit selon

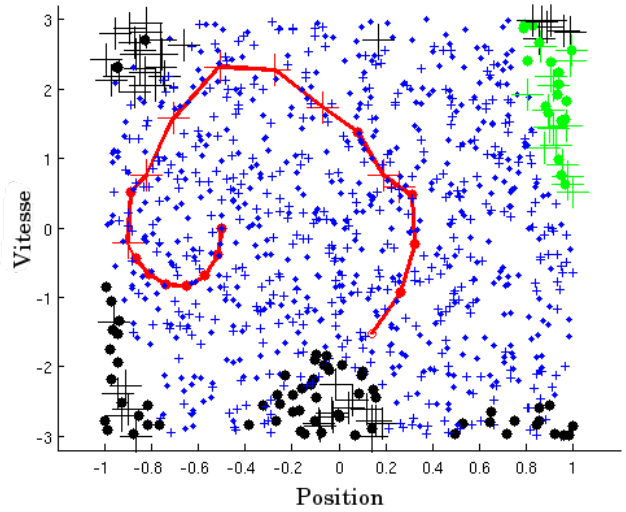

Figure 7. Représentation de l'ensemble de transitions $\mathcal{G}^1_{N_{\max}}$ (obtenu à partir de tirages uniformes)

la politique de décision $\tilde{\mathbf{u}}^*_{\mathcal{F}^1_{\mathbf{N}_{\max}}}$ (resp. $\tilde{\mathbf{u}}^*_{\mathcal{G}^1_{\mathbf{N}_{\max}}}$). On peut observer à la figure 6 que notre stratégie tend à échantillonner des transitions situées au voisinage de trajectoires ayant de bonnes performances.

4. Deuxième approche: augmentation de la précision des bornes

Cette deuxième approche est applicable dans le cas où les fonctions f et ρ sont Lipschitziennes, c'est à dire dans le cas où il existe deux constantes L_f et L_ρ telles que :

$$\begin{aligned} \forall\, (x, x', u) \in \mathcal{X}^2 \times \mathcal{U}, \quad \|f(x,u) - f(x',u)\|_{\mathcal{X}} &\leq L_f \|x - x'\|_{\mathcal{X}}, \\ |\rho(x,u) - \rho(x',u)| &\leq L_\rho \|x - x'\|_{\mathcal{X}}\ . \end{aligned}$$

On suppose également que deux constantes L_f et L_ρ satisfaisant les équations ci-dessus sont connues. L'approche se base sur une technique de calcul de bornes (inférieure et supérieur) sur le retour des politiques de décision récemment développée (Fonteneau *et al.*, 2011b) dont on donne les principaux résultats en section 4.1. La stratégie d'échantillonnage est ensuite décrite en section 4.2. Une illustration sur un problème jouet est donnée en section 4.3.

4.1. Calculer des bornes sur le retour des politiques de décision

Etant donnée une séquence de décisions $\mathbf{u} = (u_0, \ldots, u_{T-1}) \in \mathcal{U}^T$, une borne inférieure sur le retour $J^{\mathbf{u}}(x_0)$ peut être calculée de la façon suivante :

PROPOSITION 3 (Borne inférieure). —
Soit $\left[\left(x^{l_t}, u^{l_t}, r^{l_t}, y^{l_t}\right)\right]_{t=0}^{T-1} \in \mathcal{F}_n^T$ *une séquence de transitions telle que*

$$u^{l_t} = u_t \quad \forall t \in \{0, \ldots, T-1\} .$$

Alors:

$$J^{\mathbf{u}}(x_0) \geq \sum_{t=0}^{T-1} r^{l_t} - \sum_{t=0}^{T-1} L_{Q_{T-t}} \left\| y^{l_{t-1}} - x^{l_t} \right\|_{\mathcal{X}} ,$$

avec

$$L_{Q_{T-t}} = L_\rho \sum_{i=0}^{T-t-1} \left(L_f\right)^i .$$

Ce résultat est démontré dans (Fonteneau *et al.*, 2011b). Une borne inférieure maximale peut être calculée en maximisant la borne donnée ci-dessus sur l'ensemble des séquences de transitions satisfaisant la condition $u^{l_t} = u_t \quad \forall t \in \{0, \ldots, T-1\}$. Dans la suite, on note $\mathcal{F}_{n,\mathbf{u}}^T$ l'ensemble des séquences de transitions satisfaisant à cette condition :

$$\mathcal{F}_{n,\mathbf{u}}^T = \left\{ \left[\left(x^{l_t}, u^{l_t}, r^{l_t}, y^{l_t}\right)\right]_{t=0}^{T-1} \in \mathcal{F}_n^T | u^{l_t} = u_t \quad \forall t \in 0, \ldots, T-1 \right\}$$

On a:

DÉFINITION 4 (Borne inférieure maximale). —

$$L^{\mathbf{u}}(\mathcal{F}_n, x_0) = \max_{\left[\left(x^{l_t}, u^{l_t}, r^{l_t}, y^{l_t}\right)\right]_{t=0}^{T-1} \in \mathcal{F}_{n,\mathbf{u}}^T} \sum_{t=0}^{T-1} r^{l_t} - \sum_{t=0}^{T-1} L_{Q_{T-t}} \left\| y^{l_{t-1}} - x^{l_t} \right\|_{\mathcal{X}} .$$

De façon analogue, une borne supérieure minimale $U^{\mathbf{u}}(\mathcal{F}_n, x_0)$ peut être calculée:

DÉFINITION 5 (Borne supérieure minimale). —

$$U^{\mathbf{u}}(\mathcal{F}_n, x_0) = \min_{\left[\left(x^{l_t}, u^{l_t}, r^{l_t}, y^{l_t}\right)\right]_{t=0}^{T-1} \in \mathcal{F}_{n,\mathbf{u}}^T} \sum_{t=0}^{T-1} r^{l_t} + \sum_{t=0}^{T-1} L_{Q_{T-t}} \left\| y^{l_{t-1}} - x^{l_t} \right\|_{\mathcal{X}} .$$

Ces deux bornes ont la propriété de converger vers le retour de la séquence de décisions $\mathbf{u}$ quand la dispersion de l'échantillon converge vers zero (démonstration donnée dans (Fonteneau *et al.*, 2011b)):

DÉFINITION 6 (Dispersion de l'échantillon $\mathcal{F}_n$). —

$$\alpha^*(\mathcal{F}_n) = \sup_{x\in\mathcal{X}} \min_{l\in\{1,\dots,n\}} \left\|x^l - x\right\|_{\mathcal{X}} .$$

PROPOSITION 7 (Précision des bornes). —

$$\begin{aligned} \exists C_b > 0 : \quad J^{\mathbf{u}}(x_0) - L^{\mathbf{u}}(\mathcal{F}_n, x_0) &\leq C_b\alpha^*(\mathcal{F}_n) , \\ U^{\mathbf{u}}(\mathcal{F}_n, x_0) - J^{\mathbf{u}}(x_0) &\leq C_b\alpha^*(\mathcal{F}_n) . \end{aligned}$$

4.2. Stratégie d'échantillonnage

Avant de décrire plus en détail cette deuxième stratégie d'échantillonnage, on introduit quelques définitions. On remarque tout d'abord qu'une politique de décision ne peut être optimale (étant donné un ensemble de transitions $\mathcal{F}$) qu'à condition que sa borne supérieure ne soit pas inférieure strictement à la borne inférieure d'une quelconque autre politique. Une politique de décision vérifiant ce critère est qualifiée de "politique optimale candidate étant donné $\mathcal{F}_n$". On note $\Pi(\mathcal{F}_n, x_0)$ l'ensemble des politiques satisfaisant cette propriété:

DÉFINITION 8 (Politiques optimales candidates étant donné $\mathcal{F}_n$). —

$$\Pi(\mathcal{F}n, x_0) = \left\{ \mathbf{u} \in \mathcal{U}^T \quad | \quad \forall \mathbf{u}' \in \mathcal{U}^T, U^{\mathbf{u}}(\mathcal{F}_n, x_0) \geq L^{\mathbf{u}'}(\mathcal{F}_n, x_0) \right\} .$$

On définit également l'ensemble des transitions compatibles avec $\mathcal{F}$:

DÉFINITION 9 (Transitions compatibles avec $\mathcal{F}_n$). —
Une transition $(x, u, r, y) \in \mathcal{X} \times \mathcal{U} \times \mathbb{R} \times \mathcal{X}$ est compatible avec l'ensemble de transitions $\mathcal{F}_n$ si

$$\forall (x^l, u^l, r^l, y^l) \in \mathcal{F}_n, \quad \left(u^l = u\right) \implies \begin{cases} \left|r - r^l\right| & \leq & L_\rho \|x - x^l\|_{\mathcal{X}}, \\ \left\|y - y^l\right\|_{\mathcal{X}} & \leq & L_f \|x - x^l\|_{\mathcal{X}} . \end{cases}$$

On désigne par $\mathcal{C}(\mathcal{F}_n) \subset \mathcal{X} \times \mathcal{U} \times \mathbb{R} \times \mathcal{U}$ l'ensemble qui contient toutes les transitions compatibles avec $\mathcal{F}_n$.

Cette deuxième stratégie génère de nouvelles transitions de manière itérative. Etant donné un ensemble de n transitions préalablement générées $\mathcal{F}_n$, la $(n+1)-$ème transition est générée à partir du couple état-décision $(x^{n+1}, u^{n+1}) \in \mathcal{X} \times \mathcal{U}$ qui minimise, dans les pires conditions, l'imprécision des bornes des politiques optimales candidates à l'itération suivante:

$$(x^{n+1}, u^{n+1}) \in \underset{(x,u)\in\mathcal{X}\times\mathcal{U}}{\arg\min} \left\{ \max_{\substack{(r,y)\in\mathbb{R}\times\mathcal{X}\ s.t.(x,u,r,y)\in\mathcal{C}(\mathcal{F}_n) \\ \mathbf{u}\in\Pi(\mathcal{F}_n\cup\{(x,u,r,y)\},x_0)}} \delta^{\mathbf{u}}\left(\mathcal{F}_n\cup\{(x,u,r,y)\},x_0\right) \right\}$$

où

$$\delta^{\mathbf{u}}(\mathcal{F}_n, x_0) = U^{\mathbf{u}}(\mathcal{F}_n, x_0) - L^{\mathbf{u}}(\mathcal{F}_n, x_0)\ .$$

Etant donné les propriétés de convergence des bornes, on conjecture que la séquence $\left(\Pi\left(\mathcal{F}_n, x_0\right)\right)_{n\in\mathbb{N}}$ converge vers l'ensemble des politiques optimales en un nombre fini d'itérations. La démonstration de cette propriété théorique est remise à de futurs travaux. On propose ci-dessous une première validation expérimentale.

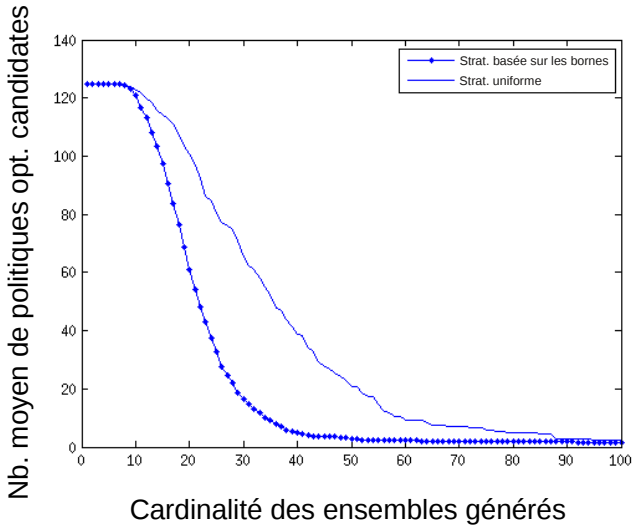

Figure 8. Evolution du nombre moyen de politiques optimales candidates en fonction de la taille de l'ensemble des transitions générées avec notre stratégie et avec une stratégie uniforme (moyennes empiriques sur 50 tests)

4.3. Illustration

On considère le problème suivant. La dynamique du système et la fonction de récompense sont définies de la manière suivante :

$$\begin{aligned} f(x,u) &= x+u, \\ \rho(x,u) &= x+u\,. \end{aligned}$$

L'espace d'états est inclus dans $\mathbb{R}$. L'espace d'action est

$$\mathcal{U} = \{-0.20, -0.10, 0, +0.10, +0.20\}.$$

On considère un horizon temporel $T = 3$, ce qui engendre un ensemble de $5^3 = 125$ differentes politiques de décision. L'état initial est $x_0 = -0.65$. De manière immédiate, on observe que la politique optimale est unique et consiste à prendre l'action $+0.20$ trois fois de suite.

Dans les expériences suivantes, les valeurs $\underset{(x,u)\in\mathcal{X}\times\mathcal{U}}{\arg\min}$ et $\underset{(r,y)\in\mathbb{R}\times\mathcal{X}\ s.t.(x,u,r,y)\in\mathcal{C}(\mathcal{F}_n)}{\max}$, dont le calcul est complexe, sont approchées par des méthodes de recherche aléatoire (c'est-à-dire en générant un ensemble de solutions aléatoires et en prenant l'optimum sur l'ensemble). La stratégie itérative est amorcée avec des ensembles de $n = 5$ transitions (une transition par décision):

$$\left\{ \begin{array}{l} (0, -0.20, \rho(0,-0.20), f(0,-0.20))\,, \\ (0, -0.10, \rho(0,-0.10), f(0,-0.10))\,, \\ (0, 0, \rho(0,0), f(0,0))\,, \\ (0, 0.10, \rho(0,0.10), f(0,0.10))\,, \\ (0, 0.20, \rho(0,0.20), f(0,0.20)) \end{array} \right\}$$

et on génère itérativement des transitions supplémentaires avec notre stratégie. Notre stratégie est comparée avec une stratégie d'échantillonnage uniforme (partant des mêmes ensembles initiaux de transitions). On donne à la figure 8 l'évolution de la moyenne empirique du nombre de politique(s) optimale(s) candidate(s) (moyennée sur 50 tests) en fonction de la cardinalité des ensembles de transitions ainsi générés [2]. On observe que notre stratégie d'échantillonnage permet d'éliminer les politiques non optimales plus rapidement que la stratégie d'échantillonnage uniforme. En particulier, on observe qu'en moyenne, les ensembles de 40 transitions générés avec notre stratégie d'échantillonnage permettent de discriminer les politiques de manière aussi

2. On a choisi de représenter des résultats moyennés sur 50 tests plutôt que les résultats d'un test unique car (i) la variance des résultats obtenus avec la stratégie d'échantillonnage uniforme est grande et (ii) la variance des résultats obtenus avec notre stratégie n'est pas négligeable puisque la procédure utilisée pour approcher les opérateurs $\arg\min$ et $\max$ repose sur un générateur de nombres aléatoires.

efficace que les ensembles de 80 transitions générés par la stratégie uniforme. Précisons finalement que dans le cas précis de ce problème, il faudrait générer 5 + 25 + 125 = 155 transitions pour être certain d'éliminer toutes les politiques non optimales (en les essayant toutes).

5. Travaux connexes

Echantillonner de manière adéquate la dynamique et la fonction de récompense d'un système inconnu est un problème qui a déjà été abordé par de nombreux auteurs. L'approche développée dans (Ephsteyn *et al.*, 2008) est probablement celle qui se rapproche le plus de notre stratégie d'échantillonnage. Dans (Ephsteyn *et al.*, 2008), les auteurs proposent une stratégie itérative favorisant les zones de l'espace supposées influencer la politique de décision. Ces travaux sont menés dans un contexte stochastique, stationnaire et pour un espace d'état fini, alors que nous considérons ici des problèmes déterministes dans des espaces d'état continus.

La plupart des travaux en apprentissage par renforcement abordant le problème de la génération d'échantillons informatifs ont préféré des solutions visant à contrôler un système de manière à générer des informations pouvant être utilisées pour augmenter les performances des politiques de décision, tout en gardant potentiellement un comportement générateur de bonnes performances. Une approche classique pour aborder ce dilemme entre exploration et exploitation (Auer, 2003 ; Cohen *et al.*, 2007 ; Castronovo *et al.*, 2012) est d'adopter une politique de type $\epsilon-$greedy qui prend l'initiative, avec une probabilité donnée, de prendre une décision différente de celle suggérée par la politique supposée optimale (Thrun, 1992 ; Kaelbling, 1993 ; Sutton, Barto, 1998). Ce problème a été particulièrement bien étudié dans le cas de problèmes ayant un état unique (Bubeck *et al.*, 2009 ; Maes *et al.*, 2011).

Dans le domaine de la discrétisation adaptative pour la programmation dynamique, on peut également trouver des travaux qui proposent des stratégies se rapprochant de notre approche. Dans ces travaux, l'espace état-décision est itérativement discrétisé de sorte à mener rapidement à une politique de décision optimale (voir par exemple (Munos, Moore, 2002)). Cependant, la complexité – en termes de temps de calcul – de notre stratégie ne lui permet pas d'être une stratégie d'échantillonnage adaptative performante.

Enfin, on peut également mentionner le fait qu'identifier un sous-ensemble de petite taille de transitions à partir duquel on puisse apprendre une bonne politique de décision est un problème qui a déjà été traité dans des contextes différents du nôtre. Par exemple, (Ernst, 2005) propose une approche pour extraire un sous-ensemble particulièrement informatif de transitions à partir de l'estimation d'erreurs d'approximation dans des équations de Bellman. Dans (Rachelson *et al.*, 2011), où aucune contrainte sur le nombre total de transitions générées n'est fixée, les auteurs se concentrent sur l'identification d'un petit sous-ensemble de transitions et parviennent à apprendre via un algorithme BMRL une politique de décision optimale basée sur moins de 20 transitions, mais au prix de centaines de milliers de transitions générées.

6. Conclusions

Cet article présente deux stratégies d'échantillonnage itératives dont la finalité est de générer des ensembles de transitions informatifs dans le cas de la résolution de problèmes de contrôle optimal déterministes à espace d'états continu. Des expériences réalisées sur deux problèmes jouets ont donné des résultats prometteurs. En particulier, les stratégies proposées se montrent nettement plus efficaces que des stratégies d'échantillonnage uniformes.

Dans un premier temps, il serait intéressant d'étendre ces deux approches afin qu'elles soient opérationnelles dans des contextes stochastiques. Une première direction de recherche consiste à multiplier les échantillonnages pour chaque couple état-décision testé.

Ces premiers résultats encouragent à étendre l'analyse de ces deux approches. En particulier, dans le cas de la première stratégie, il serait intéressant d'étudier sous quelles conditions une modification de la politique causée par l'ajout d'une nouvelle transition correspond également à une amélioration réelle de la politique de décision. Il serait tout aussi intéressant de caractériser l'erreur de prédiction et son influence sur les détections de transitions qui n'apportent finalement pas de modifications de la politique courante. L'objectif de ces travaux serait d'identifier sous quelles hypothèses les ensembles de transitions ainsi générés pourraient converger vers des ensembles de transitions à partir desquels des politiques de décision (quasi-)optimales pourraient être apprises. Dans le cas de la deuxième stratégie, il serait intéressant d'étudier quelles propriétés théoriques peuvent être obtenues en vertu des propriétés de convergence des bornes. Enfin, les stratégies d'échantillonnage introduites dans cet article ont été spécifiées et expérimentées dans un contexte déterministe, avec un espace de décision discret et fini. Il serait intéressant d'étudier comment mettre en oeuvre ces stratégies dans des contextes différents.

Remerciements

Raphael Fonteneau remercie le F.R.S.-FNRS (Fonds de la Recherche Scientifique). Ce papier présente des résultats obtenus grâce au Pôle d'Attraction Inter-universitaire (PAI) belge DYSCO (Dynamical Systems, Control and Optimization) ainsi qu'au réseau européen d'excellence PASCAL2. La responsabilité scientifique demeure celle des auteurs.

Bibliographie

Auer P. (2003). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, vol. 3, p. 397 - 422.

Aurenhammer F. (1991). Voronoi diagrams − a survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)*, vol. 23, n° 3, p. 345-405.

Bubeck S., Munos R., Stoltz G., Szepesvári C. (2009). Online optimization in X-armed bandits. In *Advances in Neural Information Processing Systems 21 (NIPS 2009)*, p. 201-208. MIT Press.

Busoniu L., Babuska R., De Schutter B., Ernst D. (2010). *Reinforcement Learning and Dynamic Programming using Function Approximators*. Taylor & Francis CRC Press.

Castronovo M., Maes F., Fonteneau R., Ernst D. (2012, June). Learning exploration/exploitation strategies for single trajectory reinforcement learning. In *10th european workshop on reinforcement learning (EWRL 2012)*. Edinburgh, Scotland.

Cohen J., McClure S., Yu A. (2007). Should I stay or should I go? How the human brain manages the trade-off between exploitation and exploration. *Philosophical Transactions of the Royal Society B 29*, vol. 362, n° 1481, p. 933-942.

Ephsteyn A., Vogel A., DeJong G. (2008). Active reinforcement learning. In *Proceedings of the 25th international conference on machine learning (ICML 2008)*, vol. 307.

Ernst D. (2005). Selecting concise sets of samples for a reinforcement learning agent. In *Proceedings of the third international conference on computational intelligence, robotics and autonomous systems (CIRAS 2005)*. Singapore.

Ernst D., Geurts P., Wehenkel L. (2005). Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, vol. 6, p. 503-556.

Fonteneau R. (2011). Contributions to Batch Mode Reinforcement Learning. *PhD Thesis, University of Liège*.

Fonteneau R., Murphy S., Wehenkel L., Ernst D. (2010). Generating informative trajectories by using bounds on the return of control policies. In *Proceedings of the workshop on active learning and experimental design 2010 (in conjunction with AISTATS 2010)*.

Fonteneau R., Murphy S., Wehenkel L., Ernst D. (2011a). Active exploration by searching for experiments falsifying an already induced policy. In *Proceedings of the 2011 IEEE symposium on adaptive dynamic programming and reinforcement learning (IEEE ADPRL 2011)*, p. 40-47.

Fonteneau R., Murphy S. A., Wehenkel L., Ernst D. (2011b). Towards min max generalization in reinforcement learning. In *Agents and artificial intelligence: International conference, ICAART 2010, valencia, spain, january 2010, revised selected papers. series: Communications in computer and information science (ccis)*, vol. 129, p. 61-77. Springer, Heidelberg.

Ingersoll J. (1987). *Theory of financial decision making*. Rowman and Littlefield Publishers, Inc.

Kaelbling L. (1993). *Learning in Embedded Systems*. MIT Press.

Maes F., Wehenkel L., Ernst D. (2011, September). Automatic discovery of ranking formulas for playing with multi-armed bandits. In *9th european workshop on reinforcement learning (EWRL 2011)*. Athens, Greece.

Munos R., Moore A. (2002). Variable resolution discretization in optimal control. *Machine Learning*, vol. 49, p. 291-323.

Murphy S. (2003). Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society, Series B*, vol. 65(2), p. 331-366.

Murphy S. (2005). An experimental design for the development of adaptive treatment strategies. *Statistics in Medicine*, vol. 24, p. 1455-1481.

Ormoneit D., Sen S. (2002). Kernel-based reinforcement learning. *Machine Learning*, vol. 49, n° 2-3, p. 161-178.

Rachelson E., Schnitzler F., Wehenkel L., Ernst D. (2011). Optimal sample selection for batch-mode reinforcement learning. In *3rd international conference on agents and artificial intelligence (ICAART 2011).*

Riedmiller M. (2005). Neural fitted Q iteration - first experiences with a data efficient neural reinforcement learning method. In *Proceedings of the sixteenth european conference on machine learning (ECML 2005)*, p. 317-328. Porto, Portugal.

Sutton R., Barto A. (1998). *Reinforcement Learning*. MIT Press.

Thrun S. (1992). The role of exploration in learning control. In D. White, D. Sofge (Eds.), *Handbook for intelligent control: Neural, fuzzy and adaptive approaches.* Van Nostrand Reinhold.

Viterbi A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, vol. 13, n° 2, p. 260- 269.