Apprentissage actif de modèle de MDP

Mauricio Araya-López[1], Olivier Buffet[2],
Vincent Thomas[1], and François Charpillet[2]

[1] Nancy Université / LORIA prénom.nom@loria.fr
[2] INRIA / LORIA prénom.nom@loria.fr

Résumé : Dans cet article, nous nous intéressons à un problème d'apprentissage actif consistant à déduire le modèle de transition d'un Processus de Décision Markovien (MDP) en agissant et en observant les transitions résultantes. Ceci est particulièrement utile lorsque la fonction de récompense n'est pas initialement accessible. Notre proposition consiste à formuler ce problème d'apprentissage actif en un problème de maximisation d'utilité dans le cadre de l'apprentissage par renforcement bayésien avec des récompenses dépendant de l'état de croyance. Après avoir présenté trois critères de performance possibles, nous en dérivons des récompenses dépendant de l'état de croyance que l'on pourra utiliser dans le processus de prise de décision. Comme le calcul de la fonction de valeur bayésienne optimale n'est pas envisageable pour de larges horizons, nous utilisons un algorithme simple pour résoudre de manière approchée ce problème d'optimisation. Malgré le fait que la solution est sous-optimale, nous montrons expérimentalement que notre proposition est néanmoins efficace dans un certain nombre de domaines.

1 Introduction

L'apprentissage dans des processus de décision markovien (MDP pour *Markov Decision Process*) consiste souvent à maximiser l'utilité totale pour un problème donné. Cependant, apprendre le modèle de transition d'un MDP indépendamment de la fonction d'utilité—si elle existe—peut être une tâche très utile dans certains domaines. Par exemple, cela peut permettre d'apprendre le modèle de transition dans un *processus batch*, lorsqu'on est dans un premier temps intéressé par choisir les bonnes actions optimisant la récolte d'information, puis dans un second temps, par recevoir des récompenses (Şimşek & Barto, 2006). De plus, dans certains cas, nous n'avons pas accès à la fonction d'utilité comme lorsqu'on construit des modèles pour la simulation ou qu'on raffine un modèle existant. Dans ces derniers cas, on souhaite seulement trouver le meilleur modèle possible, peut importe ce qui en sera fait ensuite.

Apprendre le modèle d'un MDP stochastique est une tâche simple lorsqu'on dispose d'une politique. Dans ce cas, l'historique des transitions fournit les données nécessaires et trouver les paramètres optimaux pour la distribution sur les modèles choisie peut se faire en utilisant le maximum de vraisemblance. Par contre, trouver la politique qui explore de manière optimale un MDP dans le but d'acquérir la meilleure distribution sur les modèles possibles peut être un problème complexe.

Dans cet article, nous nous sommes intéressés à un problème d'apprentissage actif consistant à apprendre de bonnes distributions de probabilités sur les modèles en agissant dans un MDP dont la fonction d'utilité est inaccessible. A notre connaissance, il existe peu de recherches qui se concentre sur le problème d'apprentissage actif de modèles de MDP stochastiques arbitraires (Szepesvári, 2009). Parmi ces travaux, on trouve l'apprentissage de réseaux bayésiens dynamiques (DBN - Dynamic Bayesian Networks) pour représenter des MDP factorisés (Jonsson & Barto, 2007), mais ces techniques ne produisent qu'un seul réseau candidat qui se révèle être assez éloigné du vrai modèle.

Pour résoudre ce problème d'apprentissage actif, nous proposons de se placer dans le cadre de l'apprentissage par renforcement bayésien (BRL pour *Bayesian Reinforcement Learning*) en y ajoutant des récompenses dépendant de l'état de croyance. Dans un premier temps, nous formulons ce problème d'apprentissage actif comme un problème de maximisation d'utilité en utilisant des récompenses dépendant de l'état de croyance qui aura été déduit. Nous définissons ensuite des

critères de performance afin de mesurer la qualité des distributions de probabilité produites par les différentes politiques. A partir de ces critères, nous dérivons les récompenses dépendant de l'état de croyance qui seront utilisées pour construire les politiques d'exploration. Enfin, comme le calcul de la fonction de valeur bayésienne optimale est impossible en pratique, nous résolvons ce problème de manière sous-optimale en utilisant une technique **myopic** nommée *exploit*.

Les récompenses dépendant de l'état de croyance ont été utilisées par le passé en tant que méthodes heuristiques pour les POMDP (Processus de Décision Markovien Partiellement Observable). Par exemple, dans la navigation côtière (Roy & Thrun, 1999), la convergence est accélérée en utilisant un *reward shaping* fondé sur un critère lié à la quantité d'information. De plus, les POMDP utilisant des récompenses basées sur l'état de croyance ont été étudiés récemment par Araya-López *et al.* (2010) qui montre qu'on peut y appliquer les algorithmes classiques des POMDP à condition d'effectuer quelques modifications. Malheureusement, ces techniques ne peuvent pas être directement appliquées à l'apprentissage par renforcement bayésien, de la même manière que les algorithmes classiques POMDP ne peuvent pas être appliqués au BRL : le type particulier des états de croyance n'est pas adapté à ces algorithmes.

Le reste de l'article est organisé de la manière suivante. Dans la section 2, nous présentons un état de l'art de l'apprentissage par renforcement bayésien et des algorithmes qui ont été proposés dans ce cadre. Ensuite, dans la section 3, nous introduisons la méthodologie utilisée pour résoudre ce problème d'apprentissage actif en tant que problème d'apprentissage par renforcement bayésien avec des récompenses dépendant de l'état de croyance. Nous présenterons en particulier les critères de performance choisis et leurs récompenses dérivées respectives. Dans la section 4, nous présenterons les résultats des différentes expériences que nous avons menées sur différents modèles de MDP issus de l'état de l'art. Enfin, dans la section 5, nous conclurons et présenterons quelques perspectives futures à ces travaux.

2 État de l'art

2.1 Apprentissage par renforcement

Formellement, un Processus de Décision Markovien (MDP) (Puterman, 1994) est défini par un tuple $\langle \mathcal{S}, \mathcal{A}, T, R \rangle$ où, à chaque pas de temps, le système se trouvant dans l'état $s \in \mathcal{S}$ (*l'espace d'état*), l'agent effectue une action $a \in \mathcal{A}$ (*l'espace d'action*) qui a pour conséquence (1) une transition vers l'état s' selon la *fonction de transition* $T(s, a, s') = Pr(s'|s, a)$ et (2) une *récompense* scalaire $r(s, a)$ obtenue pendant cette transition. L'apprentissage par renforcement (AR) (Sutton & Barto, 1998) est le problème consistant à trouver une politique de décision—associant à chaque état une action, $\pi : \mathcal{S} \mapsto \mathcal{A}$—optimale en interagissant avec le système quand le modèle (T et R) est inconnu. Un critère de performance classique est l'espérance de retour γ-pondéré

$$V_\pi(s) = E_\pi \left[\sum_{t=0}^{\infty} \gamma^t R(S^t, A^t, S^{t+1}) | S^0 = s \right],$$

où $\gamma \in [0, 1]$ désigne un facteur d'actualisation. Lorsqu'une politique optimale est suivie, cette fonction de valeur vérifie l'équation d'optimalité de Bellman (Bellman, 1954) (pour chaque $s \in \mathcal{S}$) :

$$V^*(s) = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} T(s, a, s') \left[R(s, a, s') + \gamma V^*(s') \right],$$

et le calcul de cette fonction de valeur optimale permet de déduire une politique optimale en se comportant de manière gloutonne, c'est à dire, en choisissant les actions dans $\arg\max_{a \in \mathcal{A}} Q^*(s, a)$ (la fonction de valeur état-action Q_π étant définie par $Q_\pi(s, a) = \sum_{s' \in \mathcal{S}} T(s, a, s') \left[R(s, a, s') + \gamma V_\pi(s') \right]$).

Les algorithmes classiques d'apprentissage par renforcement soit (1) estiment directement la fonction état-action optimale Q^* (*model-free RL*) soit (2) apprennent T et R pour calculer V^* ou Q^* (*model-based RL*). Cependant, dans les deux cas, une difficulté majeure consiste à choisir correctement les actions à exécuter de manière à trouver un compromis entre exploiter la connaissance actuelle et explorer l'environnement pour acquérir plus d'information.

2.2 Apprentissage par renforcement bayésien à base de modèle

Nous considérons dans cette partie l'*apprentissage par renforcement bayésien à base de modèle* (Strens, 2000), à savoir, un apprentissage par renforcement *model-based* pour lequel la connaissance sur le modèle—désormais une variable aléatoire $\boldsymbol{M}$—est représentée en utilisant une distribution de probabilité—généralement structurée—sur les modèles de transition possibles. [1] Une distribution $Pr(\boldsymbol{M}^0)$ qui doit être spécifiée initialement, est mise à jour selon la loi de Bayes après chaque nouvelle transition (s, a, s') :

$$Pr(\boldsymbol{M}^{t+1}|\boldsymbol{M}^0, h^{t+1}) = Pr(\boldsymbol{M}^{t+1}|\boldsymbol{M}^t, s^t, a^t, s^{t+1}) Pr(\boldsymbol{M}^t|\boldsymbol{M}^0, h^t),$$

où $h^t = s^0, a^0, \cdots, s^{t-1}, a^{t-1}, s^t$ correspond à l'historique des couples état-action jusqu'au temps t. Cette variable aléatoire est habituellement connue comme la *croyance* sur le modèle et définit un *Belief-MDP* avec un espace d'état infini. Résoudre de manière optimale ce *Belief-MDP* est impossible en pratique à cause de la complexité croissante selon l'horizon de planification, mais formuler le problème de l'apprentissage par renforcement selon une approche bayésienne constitue une manière appropriée de gérer le dilemme exploration/exploitation. Bien que les algorithmes POMDP classiques traitent des *belief-MDP*, il n'est pas possible d'en tirer directement parti à cause du type particulier de l'espace des croyances. D'autres techniques approchées—offline ou online—ont ainsi été proposées, permettant d'obtenir des résultats théoriques pour un certain nombre de cas. Plusieurs techniques d'approximation ont été proposées pour le BRL et comme présenté dans (Asmuth *et al.*, 2009), la plupart des approches appartiennent à l'une de ces catégories, de la plus simple à la plus complexe, approches *undirected*, approches *myopic* et approches *belief-lookahead*.

Les approches **Undirected** ne prennent pas en compte l'incertitude sur le modèle pour choisir la prochaine action. Ainsi, elles ne raisonnent pas sur le gain d'information possible. Ces approches consistent souvent à choisir des actions aléatoires de manière occasionnelle pour favoriser l'exploration, comme en utilisant des stratégies d'exploration ϵ-greedy ou softmax, les Q-valeurs calculées étant basées sur le modèle le plus probable. Ces algorithmes convergent habituellement à la limite vers la fonction de valeur optimale, mais sans garantie quant au temps de convergence.

Les approches **Myopic** choisissent la prochaine action afin de réduire l'incertitude sur le modèle. Certaines résolvent le MDP le plus probable à l'instant t en y ajoutant une récompense d'exploration favorisant les transitions dont les modèles sont les moins connus, comme dans R-MAX (Brafman & Tennenholtz, 2003), BEB (Kolter & Ng, 2009), ou en ajoutant des des récompenses basées sur la variance (Sorg *et al.*, 2010). Un autre type d'approche, utilisé dans BOSS (Asmuth *et al.*, 2009), consiste à résoudre, quand le modèle a suffisamment changé, une estimation optimiste du véritable MDP sous-jacent (obtenue en fusionnant de nombreux modèles échantillonnés). Pour certains de ces algorithmes comme BOSS et BEB, il existe des garanties selon lesquelles, avec une grande probabilité, la fonction de valeur est proche de la fonction de valeur optimale après un nombre donné d'échantillonnages. Cependant, ces algorithmes peuvent arrêter d'explorer après un certain temps, empêchant la convergence vers la fonction de valeur optimale.

Les approches **Belief-lookahead** cherchent à trouver le compromis optimal entre exploration et exploitation. Il est en effet possible (Duff, 2002) de reformuler l'apprentissage par renforcement bayésien comme un problème consistant à résoudre un POMDP pour lequel l'état courant est le couple $\omega = (s, \boldsymbol{m})$, où s est l'état courant observable et $\boldsymbol{m}$ est le modèle caché. Chaque transition (s, a, s') est une observation qui fournit de l'information concernant $\boldsymbol{m}$. Ces approches sont peu nombreuses du fait de leur besoin en calcul très importants, mais quelques approches comme BEETLE (Poupart *et al.*, 2006) ont été proposées. Une autre technique consiste à développer l'arbre des croyances et à exécuter une approche *Branch and Bound* pour élaguer certaines branches et éviter une expansion infinie (Dimitrakakis, 2008). Cependant, les difficultés pour calculer un majorant et la nécessité de grandes capacités de calcul font que cette technique est très lente par rapport aux approches *myopic* ou *undirected*.

Parmi les diverses représentations possibles pour les croyances sur le modèle, nous avons choisi d'utiliser une distribution de Dirichlet indépendante pour chaque couple état-action. Nous représenterons l'une d'entre elle à la date t par une statistique suffisante : un vecteur d'entiers positifs $\boldsymbol{\theta}^t_{s,a}$ où $\boldsymbol{\theta}^t_{s,a}(s')$ correspond au nombre de fois où la transition (s, a, s') a été observée en

1. A partir de maintenant, nous ne considérerons pas d'incertitude sur la fonction de récompense puisque celle-ci peut s'exprimer comme une incertitude sur la fonction de transition.

incluant $\boldsymbol{\theta}^0_{s,a}(s')$ observations effectuées *a priori*. L'état de croyance peut ainsi s'écrire $\omega = (s, \boldsymbol{\theta})$, où $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{s,a}, \forall s, a\}$. Le MDP résultant correspond à ce que nous appellerons un MDP *Belief-Augmented* (BAMDP), à savoir, un *belief-MDP* particulier pour lequel l'état de croyance peut se décomposer en deux parties : l'état du système (visible) et la croyance sur le modèle (caché). Une transition (s, a, s') conduit à une mise à jour bayésienne du modèle, $\boldsymbol{\theta}'$ différant de $\boldsymbol{\theta}$ seulement par $\boldsymbol{\theta}'_{s,a}(s') = \boldsymbol{\theta}_{s,a}(s') + 1$. De plus, du fait des propriétés des distributions de Dirichlet, la fonction de transition du BAMDP $T(\omega, a, \omega')$ est donnée par

$$Pr(\omega'|\omega, a) = \frac{\boldsymbol{\theta}_{s,a}(s')}{\|\boldsymbol{\theta}_{s,a}\|_1}.$$

D'autres choix sont possibles. On pourrait par exemple prendre explicitement en compte le fait que différentes actions peuvent partager le même modèle en factorisant certaines distributions de Dirichlet ou permettre à l'algorithme d'identifier de telles structures en utilisant les distributions de Dirichlet combinées avec les processus du *restaurant chinois* ou du *buffet indien* (Asmuth *et al.*, 2009).

Pour résumer, l'apprentissage par renforcement bayésien transforme le problème consistant à prendre des décisions face à un modèle inconnu en un problème consistant à prendre des décisions lorsque l'état du système contient des paramètres non observés. Trouver le bon compromis entre exploration et exploitation revient à résoudre un BAMDP étant donnée une croyance a priori définie par les paramètres $\boldsymbol{\theta}^0$.

3 Apprentissage actif de MDP en utilisant l'apprentissage par renforcement bayésien

Dans ce article, nous nous sommes intéressés à apprendre le modèle d'un MDP en observant les transitions de manière online, mais aussi en choisissant les actions à exécuter à chaque pas de temps. Il en découle un problème de prise de décision, pour lequel la meilleure politique doit sélectionner l'action optimisant le processus d'apprentissage. Pour une politique donnée, le processus d'apprentissage est direct en utilisant le cadre bayésien, puisque la maximisation de la vraisemblance pour les distributions de Dirichlet jointes correspond à la séquence de mises à jour bayésiennes des paramètres $\boldsymbol{\theta}$ décrits dans la section 2.2. Par contre, la sélection de la politique optimale dépendra du critère utilisé pour comparer deux distributions de Dirichlet jointes produites à partir de deux politiques différentes. Parmi les options possibles, nous avons retenu trois critères de performance qui seront décrits dans la section suivante.

Pour trouver la politique optimale, nous pouvons transformer la tâche d'apprentissage actif en un problème d'apprentissage par renforcement bayésien, dont les récompenses peuvent être dérivées du critère de performance choisi. En d'autres termes, nous proposons d'étendre la définition classique de l'apprentissage par renforcement bayésien à des récompenses dépendant des paramètres $\boldsymbol{\theta}$. L'équation de Bellman s'exprime alors de la manière suivante :

$$V(\boldsymbol{\theta}, s) = \max_a \left[\sum_{s'} Pr(s'|s, a, \boldsymbol{\theta})(\rho(s, a, s', \boldsymbol{\theta}) + \gamma V(\boldsymbol{\theta}', s')) \right], \tag{1}$$

avec $\boldsymbol{\theta}'$ correspondant au vecteur paramètre décrivant la croyance *a posteriori* après la mise à jour bayésienne effectuée pour s, a et s' et $\rho(s, a, s', \boldsymbol{\theta})$ [2] correspondant à la récompense immédiate dépendant de la croyance. Dans le cadre de cette formulation, le problème d'apprentissage actif de modèles de MDP peut se résoudre de manière optimale en utilisant les techniques de programmation dynamique comme l'itération sur la valeur. Cependant, de manière analogue à l'apprentissage par renforcement bayésien, calculer la fonction de valeur est impossible en pratique à cause d'une expansion infinie de l'arbre, et, même si le facteur d'actualisation pourrait aider à en calculer une approximation, le facteur de branchement est déjà trop large pour des problèmes relativement petits.

2. Le fonction de récompense ρ peut s'exprimer sans utiliser $\boldsymbol{\theta}'$ puisque $\boldsymbol{\theta}'$ est fonction de $\boldsymbol{\theta}$, s, a et s'

Le problème que nous traitons dans cet article, peut être vu comme un problème à horizon fini, dont l'objectif est d'explorer optimalement le MDP étant donné un nombre fixe de pas de temps. En s'approchant de l'horizon, l'algorithme devrait se comporter de la manière la plus gloutonne possible parce qu'il n'y a plus de récompense à partir de ce point. Au contraire, si le problème est défini pour un horizon infini, n'importe quelle politique qui visiterait tous les couples état-action un nombre infini de fois serait optimale à la limite. C'est pourquoi le facteur γ est nécessaire pour encourager les récompenses reçues au début. En utilisant un facteur γ—et un paramètre de convergence associé ϵ—nous pouvons traiter des problèmes à horizon indéfini ou à horizon fini à large horizon. Par la suite, nous nous focaliserons sur des problèmes à horizon fini, mais les résultats peuvent naturellement s'étendre aux problèmes à horizon infini avec un γ et un ϵ donnés.

3.1 Critères de performance

En supposant que le modèle réel est inconnu, il est nécessaire de définir un moyen pour comparer deux distributions produites par des politiques différentes. Parmi l'ensemble complet des critères, nous avons sélectionné trois critères pour cet article : la *différence de variance*, la *différence d'entropie* et la *distance de Bhattacharyya*.

3.1.1 Différence de variance

Le premier critère est fondé sur l'intuition simple qu'une distribution apprise possédant une plus petite variance est une distribution plus précise. Ainsi, nous chercherons à construire les politiques produisant les distributions possédant les plus faibles variances. La variance d'une distribution à plusieurs variables sur les modèles possibles correspond à une matrice fortement creuse de taille $|\mathcal{S}|^2|\mathcal{A}| \times |\mathcal{S}|^2|\mathcal{A}|$, mais nous considérerons que la somme des variances marginales (la diagonale de la matrice) est suffisante comme métrique. La variance du i-ème élément d'une distribution de Dirichlet est donnée par

$$\sigma^2(X_i|\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}_i(\|\boldsymbol{\alpha}\|_1 - \boldsymbol{\alpha}_i)}{\|\boldsymbol{\alpha}\|_1^2(\|\boldsymbol{\alpha}\|_1 + 1)}.$$

Nous considérerons par la suite que la *différence de variance* entre une distribution paramétrée par $\boldsymbol{\theta}^t$ après t applications de la loi de Bayes et la distribution donnée a priori $\boldsymbol{\theta}^0$ correspond à la somme des différences des variances marginales, à savoir

$$D_V(\boldsymbol{\theta}^t, \boldsymbol{\theta}^0) = \sum_s \sum_a \sum_{s'} (\sigma^2(X_{s'}|\boldsymbol{\theta}^0_{s,a}) - \sigma^2(X_{s'}|\boldsymbol{\theta}^t_{s,a})).$$

3.1.2 Différence d'entropie

Une autre mesure usuelle pour les distributions de probabilité est l'entropie qui mesure l'incertitude d'une variable aléatoire. Calculer l'incertitude sur la croyance semble être un moyen naturel de quantifier la qualité de la distribution. L'entropie d'une variable aléatoire à plusieurs composantes, chacune étant définie par des distributions de Dirichlet avec les paramètres $\boldsymbol{\alpha}$, est donnée par

$$H(\boldsymbol{\alpha}) = \log(\boldsymbol{\alpha}) + (\|\boldsymbol{\alpha}\|_1 - N)\psi(\|\boldsymbol{\alpha}\|_1) - \sum_{j=1}^{N}((\boldsymbol{\alpha}_j - 1)\psi(\boldsymbol{\alpha}_j),$$

où N désigne la dimension du vecteur $\boldsymbol{\alpha}$ et $\psi(\cdot)$ la fonction digamma.

Dans le cas d'une distribution de Dirichlet jointe sur les modèles possibles, il est possible d'utiliser l'indépendance entre les couples état-action pour calculer l'entropie de chaque couple état-action de manière séparée.

Proposition 1
L'entropie d'une distribution $\boldsymbol{M}$ sur les modèles possibles définie par les paramètres $\boldsymbol{\theta}$ correspond à la somme des entropies pour chaque couple état-action : $H(\boldsymbol{\theta}) = \sum_{s,a} H(\boldsymbol{\theta}_{s,a})$.

Preuve

$$
\begin{aligned}
H(\boldsymbol{\theta}) &= -\int Pr(\boldsymbol{m}|\boldsymbol{\theta}) \log(Pr(\boldsymbol{m}|\boldsymbol{\theta}))d\boldsymbol{m}) \\
&= \underbrace{\int_{\Delta}\int_{\Delta}\dots\int_{\Delta}}_{|\mathcal{S}|\times|\mathcal{A}|} \prod_{s,a} [Pr(\boldsymbol{m}_{s,a}|\boldsymbol{\theta}_{s,a})d\boldsymbol{m}_{s,a}] \sum_{\hat{s},\hat{a}} \log(Pr(\boldsymbol{m}_{\hat{s},\hat{a}}|\boldsymbol{\theta}_{\hat{s},\hat{a}})) \\
&= \sum_{\hat{s},\hat{a}} \int_{\Delta} Pr(\boldsymbol{m}_{\hat{s},\hat{a}}|\boldsymbol{\theta}_{\hat{s},\hat{a}}) \log(Pr(\boldsymbol{m}_{\hat{s},\hat{a}}|\boldsymbol{\theta}_{\hat{s},\hat{a}}))d\boldsymbol{m}_{\hat{s},\hat{a}} \underbrace{\int_{\Delta}\int_{\Delta}\dots\int_{\Delta}}_{|\mathcal{S}|\times|\mathcal{A}|-1} \prod_{s\neq\hat{s},a\neq\hat{a}} [Pr(\boldsymbol{m}_{s,a}|\boldsymbol{\theta}_{s,a})d\boldsymbol{m}_{s,a}] \\
&= \sum_{\hat{s},\hat{a}} \int_{\Delta} Pr(\boldsymbol{m}_{\hat{s},\hat{a}}|\boldsymbol{\theta}_{\hat{s},\hat{a}}) \log(Pr(\boldsymbol{m}_{\hat{s},\hat{a}}|\boldsymbol{\theta}_{\hat{s},\hat{a}}))d\boldsymbol{m}_{\hat{s},\hat{a}} \\
&= \sum_{s,a} H(\boldsymbol{\theta}_{s,a}).
\end{aligned}
$$

□

Ainsi, la *différence d'entropie* entre une distribution paramétrée par $\boldsymbol{\theta}^t$ après t applications de la mise à jour de Bayes et la distribution a priori est

$$D_H(\boldsymbol{\theta}^t, \boldsymbol{\theta}^0) = H(\boldsymbol{\theta}^0) - H(\boldsymbol{\theta}^t).$$

3.1.3 Distance de Bhattacharyya

Les deux mesures décrites précédemment calculent la différence d'une certaine propriété entre deux distributions, cette propriété représentant la quantité d'information ayant été apprise. Dans ce contexte, la théorie de l'information fournit différentes notions de quantité d'information comme celle de Chernoff, de Shannon, de Fisher ou de Kolmogorov. Comme énoncé dans (Rauber *et al.*, 2008), les trois dernières notions de quantité d'information ne sont pas adaptées pour des distributions de Dirichlet, soit parce qu'il n'existe pas de solution analytique soit parce que ces quantités ne sont pas définies pour certaines distributions de Dirichlet paramétrées par $\boldsymbol{\theta}$.

La distance de Chernoff [3] permet de comparer deux distributions définies par les paramètres $\boldsymbol{\alpha}$ et $\boldsymbol{\alpha}'$ à partir de la notion d'information de Chernoff. Cette distance est définie par

$$C_\lambda(\boldsymbol{\alpha}, \boldsymbol{\alpha}') = -\log\left(\int_{\mathcal{X}} Pr(x|\boldsymbol{\alpha})^\lambda Pr(x|\boldsymbol{\alpha}')^{1-\lambda} dx\right),$$

avec $\lambda \in [0, 1]$. Dans le cas de distributions de Dirichlet, la distance de Chernoff s'exprime sous la forme

$$C_\lambda(\boldsymbol{\alpha}, \boldsymbol{\alpha}') = -\log\left(\frac{B(\lambda\boldsymbol{\alpha} + (1-\lambda)\boldsymbol{\alpha}')}{B(\boldsymbol{\alpha})^\lambda B(\boldsymbol{\alpha}')^{1-\lambda}}\right),$$

où $B(\cdot)$ désigne la fonction *beta* généralisée. Cette expression peut être calculée en utilisant des sommes de fonctions $\log\Gamma(\cdot)$ comme présenté dans (Rauber *et al.*, 2008).

Le choix du paramètre λ a un impact important sur le calcul de cette distance pour les distributions de Dirichlet car ce calcul peut nécessiter des évaluations de $\Gamma(\cdot)$ pour des paramètres non entiers qui sont difficiles à approcher. Heureusement, lorsque $\lambda = 1/2$, la distance de Chernoff devient la *distance de Bhattacharyya*, qui présente l'avantage que $\Gamma(n + 1/2)$ possède une forme analytique connue.

Comme pour l'entropie, il convient de noter que la distance de Chernoff (et par extension, la distance de Bhattacharyya) peut être calculée pour chaque couple état-action indépendamment des autres.

Proposition 2
La distance de Chernoff entre deux distributions $\boldsymbol{M}$ et $\boldsymbol{M}'$ ayant pour paramètres $\boldsymbol{\theta}$ et $\boldsymbol{\theta}'$ correspond à la somme des distances de Chernoff pour chaque couple état-action : $C_\lambda(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sum_{s,a} C_\lambda(\boldsymbol{\theta}_{s,a}, \boldsymbol{\theta}'_{s,a})$.

3. La distance de Chernoff n'est pas une distance à proprement parler car elle ne respecte pas toujours l'inégalité triangulaire.

Preuve

$$
\begin{aligned}
C_\lambda(\boldsymbol{\theta}, \boldsymbol{\theta}') &= -\log\left(\int Pr(\boldsymbol{m}|\boldsymbol{\theta})^\lambda Pr(\boldsymbol{m}|\boldsymbol{\theta}')^{1-\lambda} d\boldsymbol{m}\right) \\
&= -\log\left(\underbrace{\int_\Delta \int_\Delta \cdots \int_\Delta}_{|\mathcal{S}|\times|\mathcal{A}|} \prod_{s,a} \left[Pr(\boldsymbol{m}_{s,a}|\boldsymbol{\theta}_{s,a})^\lambda Pr(\boldsymbol{m}_{s,a}|\boldsymbol{\theta}'_{s,a})^{1-\lambda} d\boldsymbol{m}_{s,a}\right]\right) \\
&= -\log\left(\prod_{s,a}\left[\int_\Delta Pr(\boldsymbol{m}_{s,a}|\boldsymbol{\theta}_{s,a})^\lambda Pr(\boldsymbol{m}_{s,a}|\boldsymbol{\theta}'_{s,a})^{1-\lambda} d\boldsymbol{m}_{s,a}\right]\right) \\
&= \sum_{s,a} C_\lambda(\boldsymbol{\theta}_{s,a}, \boldsymbol{\theta}'_{s,a}).
\end{aligned}
$$

□

Ainsi, la *distance de Bhattacharyya* entre une distribution paramétrée par $\boldsymbol{\theta}^t$ après t mises à jour bayésiennes et la distribution initiale $\boldsymbol{\theta}^0$ peut être définie par

$$D_B(\boldsymbol{\theta}^t, \boldsymbol{\theta}^0) = \sum_{s,a} C_{1/2}(\boldsymbol{\theta}^t_{s,a}, \boldsymbol{\theta}^0_{s,a}).$$

Ce dernier critère de performance peut être interprété comme une mesure de la *"répartition des visites des couples état-action"*. Si la connaissance sur le modèle est favorisée dans certaines parties de l'espace d'état, alors la distance de Bhattacharyya du reste du modèle sera faible et produira ainsi une faible performance. En d'autres termes, ce critère a tendance à privilégier les distributions possédant de l'information sur le modèle en totalité plutôt que sur certaines parties de celui-ci.

3.2 Récompenses dérivées

Afin de définir des récompenses dépendant de la croyance nécessaires pour l'équation 1, nous utiliserons les critères de performance définis précédemment pour en déduire une expression analytique des fonctions de récompense immédiates. Dans le cas d'un horizon fini, la définition de la récompense est directe : les critères de performance peuvent être directement utilisés comme récompense au dernier pas de temps, tandis que toutes les autres récompenses sont égales à 0. Mais résoudre ce problème nécessiterait de considérer toute l'expansion de l'arbre jusqu'à une profondeur égale à l'horizon, ce qui s'avère extrêmement coûteux.

Donc, nous avons proposé un moyen permettant de définir la récompense immédiate à chaque pas de temps. Comme les récompenses sont définies sur chaque transition (s, a, s') et sur la croyance courante paramétrée par $\boldsymbol{\theta}$, nous utiliserons la mise à jour bayésienne pour calculer la différence de performance entre la croyance courante et la croyance a posteriori après mise à jour. Il s'agit d'une technique standard de *reward shaping* qui permet de décomposer une fonction potentiel—dans ce cas, le critère de performance—en des récompenses immédiates à chaque pas de temps, tout en conservant l'optimalité de la politique générée (Ng *et al.*, 1999).

Formellement, une récompense dérivée est définie par

$$\rho(s, a, s', \boldsymbol{\theta}^t) = D(\boldsymbol{\theta}^{t+1}, \boldsymbol{\theta}^0) - D(\boldsymbol{\theta}^t, \boldsymbol{\theta}^0),$$

où $\boldsymbol{\theta}^{t+1}$ sont les paramètres a posteriori après la transition (s, a, s'). Il convient de noter que la mise à jour de Bayes ne modifie qu'un couple état-action par mise à jour, ce qui signifie que seul un composant de la distribution, celui lié au couple état-action impliqué, est modifié à chaque mise à jour. Ceci permet de simplifier le calcul de la performance associée à une transition. De plus, les différences de *variance* et d'*entropie* respectent la propriété selon laquelle $D(\boldsymbol{\theta}^t, \boldsymbol{\theta}^0) + D(\boldsymbol{\theta}^{t+1}, \boldsymbol{\theta}^0) = D(\boldsymbol{\theta}^t, \boldsymbol{\theta}^{t+1})$, ce qui signifie que les récompenses dérivées pour ces deux critères peuvent être facilement calculées de la manière suivante

$$\rho(s, a, s', \boldsymbol{\theta}^t) = D(\boldsymbol{\theta}^t, \boldsymbol{\theta}^{t+1}), \tag{2}$$

en s'affranchissant de la dépendance à la croyance a priori.

Même si la distance de *Bhattacharyya* ne respecte pas cette propriété, par simplicité, nous utiliserons néanmoins des récompenses dérivées exprimées selon la forme de l'équation 2 sachant que nous ne préservons plus l'optimalité dans ce cas particulier.

Ainsi, après quelques calculs simples mais fastidieux, il est possible de définir les récompenses immédiates liées à la *variance*, l'*entropie* et la distance de *Bhattacharyya*. Pour présenter ces expressions, nous utiliserons les variables intermédiaires $x = \boldsymbol{\theta}_{s,a}(s')$ et $y = \|\boldsymbol{\theta}_{s,a}\|_1$ pour toutes les récompenses et $z = \|\boldsymbol{\theta}_{s,a}\|_2^2$ pour la récompense liée à la variance.

$$\rho_V(s,a,s',\boldsymbol{\theta}) = \frac{1}{y+1} - \frac{z}{y^2(y+1)} + \frac{2x-y+z}{(y+1)^2(y+2)},$$
$$\rho_H(s,a,s',\boldsymbol{\theta}) = \log\left(\frac{y}{x}\right) + \frac{|\mathcal{S}|+1}{y} - \sum_{j=x}^{y}\frac{1}{j},$$
$$\rho_B(s,a,s',\boldsymbol{\theta}) = \log\left[\frac{\Gamma(x)\sqrt{x}}{\Gamma(x+1/2)}\right] - \log\left[\frac{\Gamma(y)\sqrt{y}}{\Gamma(y+1/2)}\right].$$

Nous souhaitons aussi ajouter à ces fonctions de récompense, une fonction de récompense simple et arbitraire, motivée par le bonus d'exploration de BEB (Kolter & Ng, 2009), qui se concentre sur le gain d'information attendu en effectuant une transition à partir d'un couple état-action. Ainsi, cette récompense liée au nombre de fois où les couples état-action ont été visités—qu'on nommera *compte des couples état-action*—peut se définir simplement par

$$\rho_S(s,a,s',\boldsymbol{\theta}) = \frac{1}{y},$$

Cette dernière récompense n'est pas définie à partir d'un critère de performance mais se révèle très efficace (et simple à calculer) pour différents problèmes.

Il convient de noter que ces récompenses sont des fonctions décroissantes avec le temps, ainsi le gain instantané d'information est toujours décroissant pour un couple état-action donné. De plus, ces récompenses tendent vers 0 à l'infini, signifiant qu'il n'y a alors plus rien à apprendre.

3.3 Résoudre BRL avec des récompenses dépendant de la croyance

Il est certain qu'il va falloir adapter les algorithmes introduits dans la section 2.2 pour prendre en compte des récompenses dépendant de l'état de croyance. Par exemple, BEETLE considère que les récompenses sont des scalaires dans la représentation polynomiale de la fonction de valeur, mais dans notre contexte, ces récompenses seront des fonctions multiples de monômes, ce qui rendra la planification *offline* encore plus complexe.

EXPLOIT, l'un des algorithmes *online* les plus simples pour l'apprentissage par renforcement bayésien, consiste à résoudre un simple MDP correspondant au modèle moyen courant, ou, en d'autres termes, à itérer sur la fonction de valeur bayésienne sans faire de mise à jour bayésienne du modèle. Ensuite, EXPLOIT exécute la meilleure action pour ce simple MDP, met à jour sa croyance en observant l'état d'arrivée et recommence en résolvant le MDP correspondant au nouveau modèle moyen. Pour des horizons plus larges, cela signifie que l'on sera de plus en plus proche de la politique optimale au fur et à mesure que la croyance évolue. Pour l'apprentissage par renforcement bayésien classique, cet algorithme revient à calculer un minorant de la fonction de valeur de la croyance actuelle, c'est donc un algorithme pessimiste en terme d'information (Dimitrakakis, 2008), et relativement glouton vis à vis des récompenses.

Lorsque les récompenses dépendent de la croyance, EXPLOIT s'exprime de la manière suivante

$$V(\boldsymbol{\theta},s) = \max_a \left[\sum_{s'} Pr(s'|s,a,\boldsymbol{\theta})\left(\rho(s,a,s',\boldsymbol{\theta}) + \gamma V(\boldsymbol{\theta},s')\right)\right],$$

où le MDP à résoudre est défini par un modèle de transition $T(s,a,s') = \boldsymbol{\theta}_{s,a}(s')/\|\boldsymbol{\theta}_{s,a}\|$ et une fonction de récompense $R(s,a,s') = \rho(s,a,s',\boldsymbol{\theta})$.

Si on s'intéresse désormais aux récompenses dépendant de la croyance présentées en section 3.2, qui ont des valeurs décroissantes avec l'évolution de la croyance, résoudre ce MDP ne fournit plus

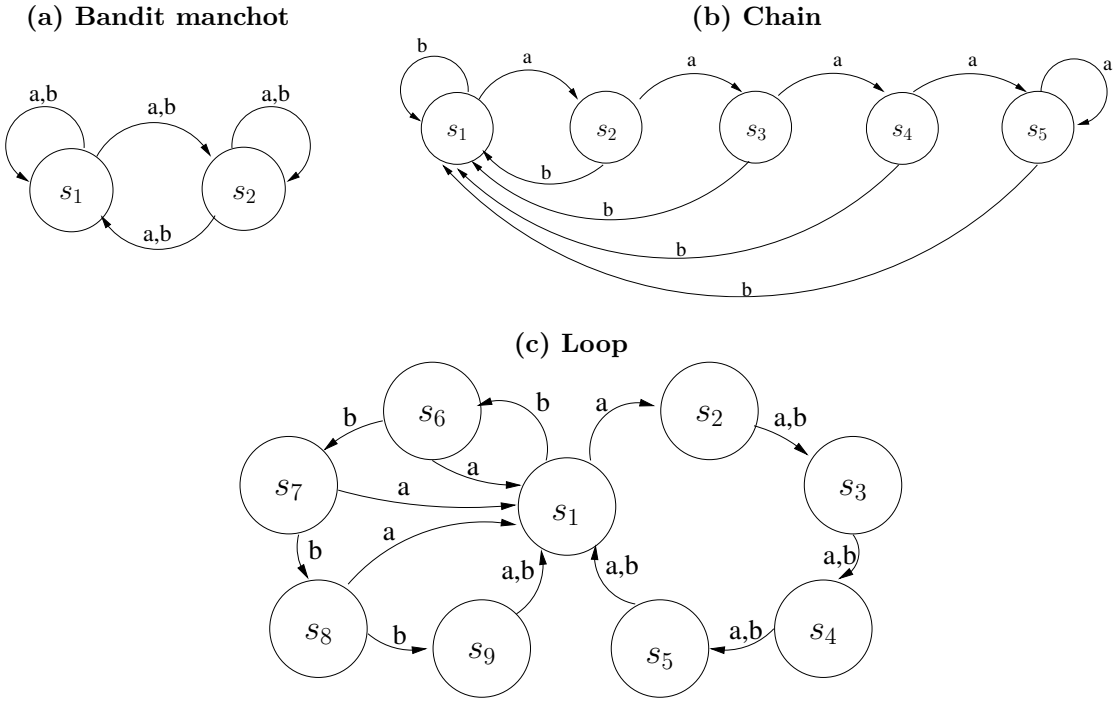

FIGURE 1 – Modèles MDP utilisés pour les expériences conduites (sans probabilités de transition).

un minorant—ni un majorant—de la fonction de valeur bayésienne, mais en fournit seulement une approximation. Cet algorithme simple exploitera l'information actuelle concernant la croyance pour explorer les parties du modèle où l'information est encore faible, et, malgré l'aspect sous-optimal de cette approximation, cet algorithme présente un bon comportement exploratoire comme nous le montrerons dans la prochaine section et se comporte de mieux en mieux pour des horizons de plus en plus larges.

4 Expériences

4.1 Conditions expérimentales

Nous avons choisi d'apprendre trois MDP de petite taille, issus de l'état de l'art concernant l'apprentissage bayésien par renforcement : le problème classique du *Bandit manchot* (Gittins, 1979), le *Chain problem* et le *Loop problem* (Strens, 2000). La figure 1 présente la structure de chaque problème (en omettant les probabilités de transition).

Dans le problème du **bandit manchot** à deux états (Figure 1-a), l'action a correspond au fait d'actionner un bras et b au fait d'actionner l'autre bras. L'état s_1 représente l'état d'arrivée associé à un succès et s_2 l'état associé à une défaite. Les résultats associés aux bras sont indépendants (ce que l'agent ignore) et le probabilités de succès sont différentes pour chaque bras : $Pr(a) = 0.8$ et $Pr(b) = 0.7$.

Dans le **chain problem** à 5 états (Figure 1-b), chaque état est connecté à l'état s_1 en effectuant l'action b et chaque état s_i est connecté à l'état suivant s_{i+1} en effectuant l'action a, sauf l'état s_5 qui est connecté à lui-même. A chaque pas de temps, l'action réellement exécutée est différente de celle choisie par l'agent avec une probabilité de 0.2.

Le dernier problème est le **Loop MDP** à 9 états (Figure 1-c). Il est constitué de deux boucles de longueur 5. Prendre l'action a de manière répétée fait traverser la boucle droite. De manière analogue, prendre l'action b de manière répétée fait traverser la boucle gauche. Prendre l'action b dans la boucle droite est équivalent à prendre l'action a, mais prendre l'action a dans la boucle de gauche fait retourner à l'état initial s_1. Toutes les transitions sont déterministes.

Tous ces problèmes débutent toujours à l'état s_1, avec une distribution a priori uniforme. D'autres

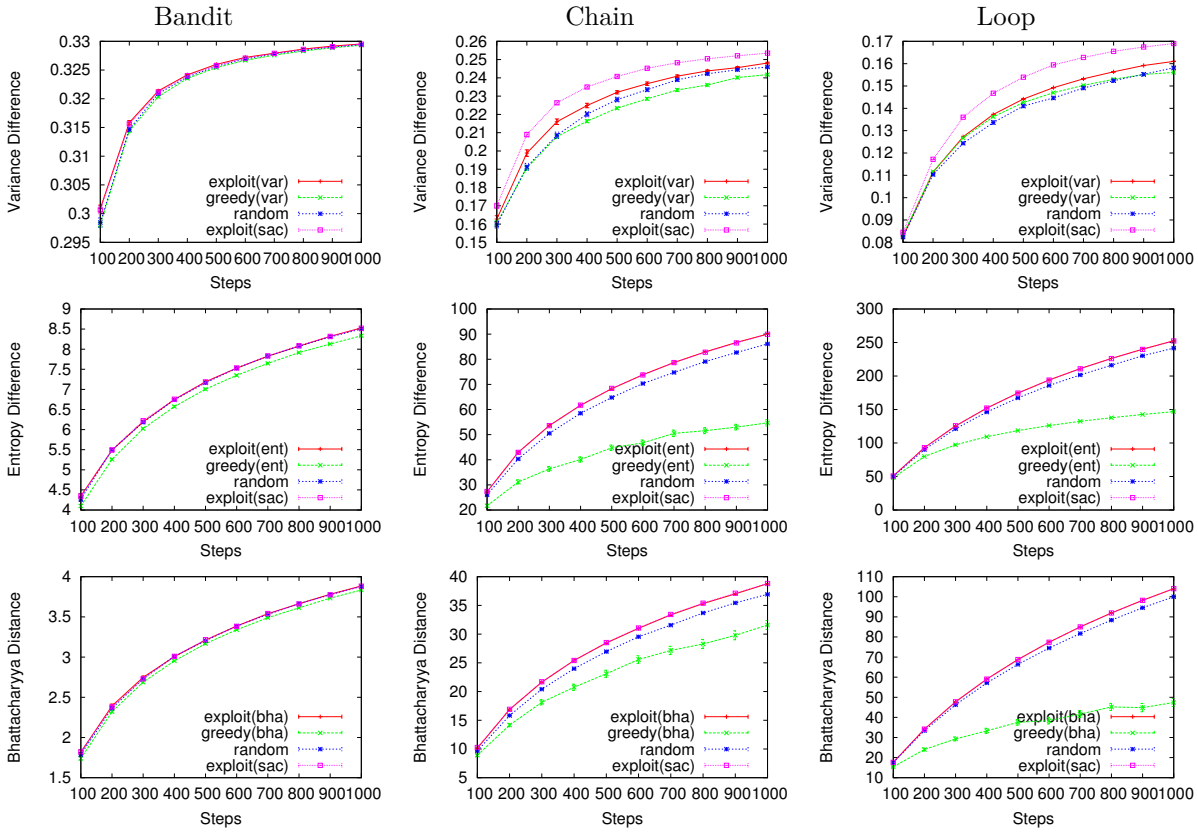

FIGURE 2 – Performance moyenne pour 100 essais en fonction de nombre de pas de temps considéré, pour chacun des trois problèmes et chacun des trois critères de performance. Pour chaque cas, les résultats obtenus pour la stratégie RANDOM (en vert), la stratégie GREEDY (en rouge) et pour l'algorithme EXPLOIT avec la récompense dérivée (en bleu) et la récompense utilisant le *compte état-action* (en cyan) sont présentés.

distributions initiales peuvent être utilisés—comme des distributions a priori informatives ou structurées—mais par simplicité, nous ne considérerons que des distributions a priori uniformes dans cet article.

Pour évaluer le comportement d'EXPLOIT, nous avons considéré deux autres politiques, à savoir une politique aléatoire RANDOM et une politique gloutonne GREEDY. La politique RANDOM choisit de manière homogène une action aléatoire à chaque pas de temps, tandis que la politique gloutonne GREEDY choisit l'action avec la récompense immédiate attendue la plus importante.

Pour chaque problème, les trois critères de performance ont été testés. Les récompenses utilisées pour EXPLOIT et GREEDY sont les récompenses dérivées respectives de la section 3.2, à savoir ρ_V, ρ_H et ρ_B en fonction du critère de performance évalué. Nous avons aussi testé la récompense basée sur le *compte des couples état-action* ρ_S pour chaque critère et chaque expérience.

Afin de résoudre de manière approchée les MDP à horizons finis issus de EXPLOIT, nous avons utilisé des MDPs à horizon infini car, pour de larges horizons, la quantité de calcul nécessaire pour résoudre le problème par programmation dynamique est trop importante. Ainsi, à chaque étape de l'algorithme EXPLOIT, nous cherchons la politique optimale correspondant à un MDP d'horizon infini avec les paramètres $\gamma = 0.95$ et $\epsilon = 0.01$, sachant que nous avons légèrement modifié le problème, et qu'en conséquence, avons ajouté une erreur à l'approximation bayésienne globale. Même si ϵ dépend dans l'absolu des valeurs de récompense, la valeur choisie de ϵ (parmi les différents ordres de grandeur que nous avons testés dans $[0.00001, 1]$) ne modifie pas les résultats.

4.2 Résultats

Nous avons testé toutes les stratégies sur chacun des problèmes pour les premiers pas de temps (entre 100 et 1000) et ce pour chaque critère de performance. La figure 2 présente les performances moyennes pour 100 essais tracées avec leur intervalle de confiance respectif de 90%.

Pour le *problème du bandit manchot*, il n'y a pas beaucoup de différence entre les algorithmes et le comportement est globalement le même pour tous les critères. Cela vient du fait que la politique optimale pour explorer un MDP entièrement connecté revient globalement à choisir de manière équitable l'action à effectuer parmi les actions disponibles, ce qui s'avère proche de la politique RANDOM.

Les *Chain problem* et *Loop problem* possèdent des dynamiques plus complexes, qui nécessitent une exploration intelligente. Cela peut s'observer pour les trois critères : la politique GREEDY présente une faible performance puisqu'une stratégie à long terme est nécessaire pour arriver dans certains états. Même si la stratégie RANDOM se comporte relativement bien, les stratégies fondées sur l'information dépassent cette technique simple pour les trois critères. Aussi bien pour le critère fondé sur l'entropie que pour la distance de Bhattacharyya, le *compte état-action* se comporte aussi bien que la récompense dérivée et pour le critère basé sur la variance, cette récompense simple se comporte mieux que la récompense dérivée. Cela signifie (i) que les récompenses dérivées peuvent être remplacées par cette récompense simple sans perte de performance (ii) que la récompense dérivée de la variance conduit à une faible performance par rapport à son propre critère. En effet, des expériences croisées entre les récompenses et les critères ont montré que les politiques construites à partir des trois fonctions de récompenses ρ_S, ρ_H et ρ_B se comportent aussi bien pour tous les critères sauf la politique construite à partir de ρ_V qui est à chaque fois moins performante.

Concernant la variabilité des résultats, on peut observer que la stratégie GREEDY est la seule stratégie présentant une variabilité significative, alors que les autres ne présentent qu'une très faible variabilité après quelques centaines de pas de temps. En particulier, le *Chain problem* produit des résultats avec plus de variabilité que le *Loop problem*, à cause des fonctions de transition stochastiques, mais, même dans ce cas, la différence entre les techniques fondées sur l'information et la stratégie RANDOM est plus grande que l'intervalle de confiance.

5 Conclusions et travaux futurs

Nous avons présenté une façon appropriée de modéliser le problème de l'apprentissage actif d'un modèle de MDP stochastique avec des dynamiques arbitraires en transformant ce problème en un problème de maximisation d'utilité dans le cadre de l'apprentissage bayésien par renforcement en utilisant des récompenses dépendant de la croyance. A cette fin, nous avons utilisé trois critères de performance qui sont habituellement utilisés pour comparer des distributions de probabilité, à savoir, la variance, l'entropie et la distance de Bhattacharyya. Pour chaque critère de performance, nous avons dérivé une fonction de récompense dépendant de la croyance de manière à ce que, dans les deux premiers cas, les récompenses accumulées correspondent exactement au critère de performance. Nous avons aussi présenté une fonction de récompense simple fondée sur des travaux antérieurs concernant l'apprentissage par renforcement bayésien. Même si ces formulations permettent—en théorie—de résoudre le problème de manière optimale, l'impossibilité à calculer la fonction de valeur bayésienne optimale conduit à utiliser des algorithmes sous-optimaux comme EXPLOIT. Les expériences que nous avons conduites montrent qu'utiliser cette simple technique produit de meilleurs résultats que de sélectionner les actions de manière aléatoire, ce qui constitue la technique de base pour explorer des modèles de MDP inconnu. Nos expériences montrent aussi qu'il n'est pas nécessaire de choisir des récompenses dérivées complexes (au moins pour EXPLOIT) pour obtenir de bons résultats, un *compte état-action* se comporte aussi bien (voire mieux) que les récompenses dérivées théoriques.

Cependant, ce travail laisse plusieurs questions ouvertes concernant les possibilités liées à l'apprentissage actif de modèles de MDP formulé dans le cadre de l'apprentissage par renforcement bayésien. Par exemple, analyser les raisons pour lesquelles les récompenses fondées sur un *compte état-action* donnent les mêmes résultats que les récompenses théoriquement dérivées pourrait nous aider à comprendre comment améliorer les performances d'EXPLOIT. Explorer d'autres techniques d'apprentissage par renforcement bayésien pourrait aussi nous aider à raffiner les politiques *myo-*

pic proposées dans cet article, ou peut être que d'autres techniques *myopic* pourraient produire de meilleurs résultats. En particulier, utiliser des techniques optimistes sur la fonction de valeur comme BOSS (Asmuth *et al.*, 2009), n'aura pas le même impact sur les récompenses dépendant de la croyance, car ces récompenses évoluent au cours de l'exécution.

Références

Araya-López M., Buffet O., Thomas V. & Charpillet F. (2010). A POMDP extension with belief-dependent rewards. In *Advances in Neural Information Processing Systems 23 (NIPS-10)*.

Asmuth J., Li L., Littman M., Nouri A. & Wingate D. (2009). A Bayesian sampling approach to exploration in reinforcement learning. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI'09)*.

Bellman R. (1954). The theory of dynamic programming. *Bull. Amer. Math. Soc.*, **60**, 503–516.

Brafman R. & Tennenholtz M. (2003). R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, **3**, 213–231.

Şimşek O. & Barto A. G. (2006). An intrinsic reward mechanism for efficient exploration. In *Proceedings of the 23rd international conference on Machine learning*, ICML'06, p. 833–840, New York, NY, USA : ACM.

Dimitrakakis C. (2008). Tree exploration for bayesian rl exploration. In *CIMCA/IAWTIC/ISE*, p. 1029–1034.

Duff M. (2002). *Optimal learning : Computational procedures for Bayes-adaptive Markov decision processes*. PhD thesis, University of Massachusetts Amherst.

Gittins J. C. (1979). Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society*, **41**(2), 148–177.

Jonsson A. & Barto A. (2007). Active learning of dynamic bayesian networks in markov decision processes. In *Proceedings of the 7th International Conference on Abstraction, Reformulation, and Approximation*, SARA'07, p. 273–284, Berlin, Heidelberg : Springer-Verlag.

Kolter J. & Ng A. (2009). Near-Bayesian exploration in polynomial time. In *Proceedings of the Twenty-Sixth International Conference on Machine Learning (ICML'09)*.

Ng A. Y., Harada D. & Russell S. (1999). Policy invariance under reward transformations : Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*, p. 278–287 : Morgan Kaufmann.

Poupart P., Vlassis N., Hoey J. & Regan K. (2006). An analytic solution to discrete Bayesian reinforcement learning. In *Proceedings of the Twenty-Third International Conference on Machine Learning (ICML'06)*.

Puterman M. (1994). *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. Wiley-Interscience.

Rauber T., Braun T. & Berns K. (2008). Probabilistic distance measures of the Dirichlet and Beta distributions. *Pattern Recognition*, **41**(2), 637–645.

Roy N. & Thrun S. (1999). Coastal navigation with mobile robots. In *Advances in Neural Information Processing Systems 12*, p. 1043–1049.

Sorg J., Singh S. & Lewis R. (2010). Variance-based rewards for approximate Bayesian reinforcement learning. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*.

Strens M. J. A. (2000). A bayesian framework for reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML'00)*, p. 943–950.

Sutton R. & Barto A. (1998). *Reinforcement Learning : An Introduction*. MIT Press.

Szepesvári C. (2009). *Reinforcement Learning Algorithms for MDPs – A Survey*. Rapport interne TR09-13, Department of Computing Science, University of Alberta.