Décision séquentielle pour la perception active : ρPOMDP *versus* POMDP

Caroline P Carvalho Chanel[1,2], Florent Teichteil-Königsbuch[2] et Patrick Fabiani[2]

[1] Université de Toulouse - ISAE - Institut Supérieur de l'Aéronautique et de l'Espace
[2] Onera - The French Aerospace Lab
2, av. Edouard Belin FR-31055 Toulouse Cedex 4
Prénom.Nom@onera.fr

Résumé : Cet article propose une étude du compromis entre la prise d'information et la décision dans un cadre applicatif qui se rapporte à une mission d'exploration, où l'agent interagit avec son environnement pour identifier l'état caché du système. Dans ce problème de décision séquentielle pour la perception, il est possible de faire reposer la fonction de récompense sur une mesure de l'incertitude sur l'état de croyance de l'agent (Araya-López *et al.*, 2010; Candido & Hutchinson, 2011; Eidenberger & Scharinger, 2010). Sa forme est donc différente de celle utilisée dans le cadre classique des POMDP qui est, pour sa part, basée sur la paire état-action. Nous comparons donc deux approches d'optimisation des politiques pour ce type de problème. D'une part nous proposons un critère mixte qui couple une mesure de l'incertitude sur l'état de croyance avec les récompenses définies par les paires état-action et nous développons un schéma algorithmique de résolution pour ce critère. D'autre part, nous proposons d'ajouter au modèle des états but fictifs au moyen des actions de classification afin de revenir à une modélisation sous-forme de POMDP classique (critère non mixte). Une étude comparative de ces approches est ici présentée afin de vérifier leur équivalence en termes de prise d'informations. Les résultats nous mènent à conclure que ces approches sont non seulement comparables et équivalentes en termes de réduction d'incertitude, mais aussi, qu'elles peuvent être utilisées en parfaite complémentarité de façon à permettre : de caractériser une politique correspondant aux taux acceptables des bonnes et mauvaises classifications et de déterminer les bonnes valeurs des coûts et des récompenses du modèle POMDP classique.

Mots-clés : Planification pour la perception, POMDP, robotique mobile.

1 Introduction

Dans la décision séquentielle sous incertitude et observabilité partielle, l'agent a besoin d'acquérir de l'information sur l'environnement afin de mener à bien sa mission. La perception de l'environnement est donc fondamentale, afin que l'acteur estime et corrige son état après l'exécution d'une action. Dans ce cas, la politique résultant de l'optimisation du problème doit intégrer une gestion du compromis entre la prise d'information d'une part et la décision *finale* qui l'amènera à achever sa mission d'autre part.

La perception active peut être définie par la prise de décision de l'agent en tenant compte des effets de ses actions sur ses capacités de perception de l'environnement (Spaan, 2008; Mihaylova *et al.*, 2002; Dutta Roy *et al.*, 2004). Elle vise donc à maximiser la connaissance de l'agent sur l'environnement.

(Smith & Simmons, 2004; Spaan & Lima, 2009; Araya-López *et al.*, 2010; Eidenberger & Scharinger, 2010) modélisent le problème de la perception active sous forme de Processus Décisionnel de Markov Partiellement Observable (POMDP). Ce modèle propose un schéma général applicable à la problématique de planification de tâches de perception et de décision à long terme. Toutefois, la modélisation du problème de décision séquentielle pour la perception active sous forme de POMDP semble être plus ou moins directe : (Spaan & Lima, 2009), (Araya-López *et al.*, 2010) et (Smith & Simmons, 2004) présentent différents cas d'application où le but de la mission robotique peut ou non se modéliser directement dans la fonction de récompense du POMDP.

Nous nous sommes intéressés à l'étude d'un cas d'application particulier : la mission robotique consiste à détecter et identifier les modèles de voitures qui sont présents dans la scène où le robot aérien évolue sous observabilité partielle. Cela revient à identifier l'état caché du système robot-environnement. Il est à noter que ce type d'application, qui se ramène à un problème de perception active pure, a été traité de

différentes façons dans la littérature : critère de performance à court terme en tenant compte du coût des actions (Eidenberger *et al.*, 2009; Eidenberger & Scharinger, 2010), ou avec un critère à long terme toutefois sans tenir compte de coûts des actions (Araya-López *et al.*, 2010; Araya López, 2013). Nous proposons dans cet article *deux approches* différentes pour traiter ce problème de perception active *en tenant compte des coûts des actions avec un critère de performance à long terme.*

Pour cette mission, la modélisation du but à atteindre dans la fonction de récompense peut dépendre de l'état de croyance de l'agent. En ce sens, nous pouvons nous appuyer sur une classe particulière de POMDP dénotée ρPOMDP (Araya-López *et al.*, 2010), en proposant un critère mixte : basé d'une part sur une mesure de l'incertitude de l'état de croyance de l'agent, et d'autre part sur l'espérance de récompenses définies par les paires état-action. Ceci constitue un axe de recherche novateur (Araya López, 2013). Nous proposons aussi un schéma algorithmique pour la résolution du critère mixte d'optimisation, qui est obtenu par programmation dynamique. Ce critère mixte permet de quantifier le compromis entre la prise d'information et la décision, en tenant compte des coûts des actions.

D'autre part, on pourrait affirmer que l'acquisition d'information est toujours un moyen, pas une fin, et donc un problème bien défini de décision avec observabilité partielle qui *doit être* modélisé sous forme d'un POMDP classique (Spaan & Lima, 2009). Nous pensons en effet que si l'on ajoute au modèle des états buts fictifs (au moyen d'actions de classification), un tel critère mixte basé sur une mesure de l'incertitude de l'état de croyance ne serait plus nécessaire dans de nombreux cas pratiques de perception active pure. Un tel critère mixte permettrait en fait d'ajuster les récompenses d'un modèle POMDP classique (critère non mixte) équivalent afin d'obtenir un tel ou un tel taux de bonnes classifications. Ces deux approches pourraient être ainsi considérées comme complémentaires.

POMDP et perception active

Dans ce type d'application l'objectif de mission est généralement de minimiser l'incertitude sur l'état de croyance de l'agent. Toutefois, dans un certain nombre de cas, il semble difficile d'adapter le cadre général des POMDP à cet objectif, car il implique de modifier le modèle de récompense $r(s, a)$ (Spaan & Lima, 2009) de façon à y incorporer les gains d'information apportés par les différents états des capteurs.

(Araya-López *et al.*, 2010) a proposé une extension du modèle POMDP. Il définit une fonction de récompense ρ basée sur l'état de croyance de l'agent autonome, c'est-à-dire telle que ρ ne dépende plus uniquement de la paire état-action. La fonction de récompense est également associée par ailleurs à une mesure d'information de l'état de croyance.

Suivant ce même objectif, (Eidenberger *et al.*, 2009; Eidenberger & Scharinger, 2010) présentent un autre exemple où le modèle POMDP est utilisé pour formaliser le problème de perception active. Le critère d'optimisation est alors fondé sur la théorie de l'information et les coûts des différentes actions. La méthode de résolution proposée n'utilise pas d'algorithmes issus de l'état de l'art du POMDP, puisque les distributions de probabilité sont représentées par des ensembles de gaussiennes, et que le critère d'optimisation n'est plus linéaire par morceaux. De plus, la méthode évalue la valeur des actions prises de manière gloutonne, c'est-à-dire que le choix d'action se ramène à l'action qui rapporte immédiatement plus d'information de façon moins coûteuse.

Nous retenons donc que ces travaux permettent de définir des critères d'optimisation mixtes, basés sur une récompense attribuée aux paires état-action $r(s, a)$ d'une part, et sur la théorie de l'information, reposant généralement sur l'entropie de l'état de croyance, ou l'information mutuelle d'autre part. Dans l'article de (Araya-López *et al.*, 2010) aucune évaluation des politiques n'a été présentée ; dans (Araya López, 2013) une implémentation algorithmique semblable à la notre a été proposée, toutefois, l'évaluation d'un critère mixte comme celui que nous proposons est suggérée en tant que piste de recherche future ; et, (Eidenberger *et al.*, 2009; Eidenberger & Scharinger, 2010) proposent seulement une résolution partielle du modèle, pour l'obtention d'une stratégie myope (à 1 coup).

Ainsi, nous nous intéressons à définir une fonction de récompense qui dépend d'une mesure d'information contenue dans l'état de croyance et du coût des actions. Ainsi, nous nous approchons du critère utilisé dans (Eidenberger *et al.*, 2009; Eidenberger & Scharinger, 2010). Nous tenons compte toutefois d'un critère ainsi que d'une méthode de résolution à long terme.

Cet article s'organise ainsi : tout d'abord nous présentons le cadre formel des POMDP ainsi que le modèle ρPOMDP. Ensuite nous détaillerons la mission d'application. Le critère mixte ρPOMDP proposé dans cette article est alors défini, ainsi que le cadre algorithmique utilisé pour l'évaluation de ce critère mixte. Après, nous présenterons le modèle POMDP classique équivalent pour cette mission d'application, dans lequel on ajoute des buts fictifs au moyen d'actions supplémentaires. Par suite, nous comparerons les différentes politiques obtenues pour les deux modèles. Nous finaliserons par la conclusion et les perspectives.

2 Cadre formel : POMDP

Formellement, un POMDP est défini comme un n-uplet $\langle S, A, \Omega, T, O, R, b_0 \rangle$ où : S est un ensemble d'états, A un ensemble d'actions, Ω un ensemble d'observations. Pour tout instant de décision $t \in \mathbb{N}$, $T : S \times A \times S \to [0; 1]$ est une fonction de transition entre les états où $\forall a \in A, \forall s_t \in S$, et $\forall s_{t+1} \in S$ on définit : $T(s_t, a, s_{t+1}) = p(s_{t+1} \mid a, s_t)$; $O : \Omega \times S \to [0; 1]$ une fonction d'observation où $\forall o_t \in \Omega, \forall a \in A$, et $\forall s_t \in S$, on définit : $O(o_t, s_t) = p(o_t | s_t, a)$; $R : S \times A \to \mathbb{R}$ une fonction de récompenses associées aux couples état-action $r(s, a)$; et b_0 une distribution de probabilité sur les états initiaux, appelée *état de croyance*. On note Δ l'ensemble des distributions de probabilités sur les états, appelé aussi espace (continu) d'états de croyance.

A chaque pas de temps t, l'agent choisit une action $a \in A$ étant donné l'état de croyance $b_t \in \Delta$, qui l'amène stochastiquement à un nouvel état $s_{t+1} \in S$; ensuite, l'agent perçoit une observation bruitée $o \in \Omega$. L'agent met à jour son *état de croyance*, grâce à la règle de Bayes, ce qui dépend de l'action réalisée, de l'observation reçue, et est fonction de l'état de croyance précédent ($s' = s_{t+1}$ suit l'état $s = s_t$).

$$b_a^o(s') = \frac{p(o|s') \sum_{s \in S} p(s'|s, a) b(s)}{\sum_{s \in S} \sum_{s'' \in S} p(o|s'') p(s''|s, a) b(s)} \tag{1}$$

L'objectif de la résolution d'un POMDP est de construire une politique, c'est-à-dire une fonction $\pi : \Delta \to A$, qui maximise un critère de performance. En robotique, où des buts symboliques de mission sont représentés par des récompenses numériques, il est généralement convenable d'optimiser l'espérance de la somme pondérée pour tout état de croyance initial (Cassandra *et al.*, 1996; Spaan & Vlassis, 2004) :

$$V^\pi(b) = E_\pi \left[\sum_{t=0}^{\infty} \gamma^t \sum_{s \in S} r(s_t, \pi(b_t)) b_t(s)) \middle| b_0 = b \right] \tag{2}$$

où γ est le facteur d'actualisation de la valeur. La valeur de la politique optimale π^* est définie par la fonction de valeur optimale qui satisfait l'équation d'optimalité de Bellman :

$$V^*(b) = \max_{a \in A} \left[\sum_{s \in S} r(s, a) b(s) + \gamma \sum_{o \in O} p(o|a, b) V^*(b_a^o) \right] \tag{3}$$

Cette fonction de valeur est linéaire par morceaux et convexe (Smallwood & Sondik, 1973), i.e, à l'instant $n < \infty$, la fonction de valeur V_n peut être représentée par des hyperplans sur Δ, nommés α-vecteurs. Un α-vecteur et l'action associée $a(\alpha_n^i)$ définissent une région dans l'espace de croyance pour lequel ce vecteur maximise V_n. Donc, la valeur d'un état de croyance peut être définie comme $V_n(b) = \max_{\alpha_n^i \in V_n} b \cdot \alpha_n^i$. Et la politique optimale à cette étape est $\pi_n(b) = a(\alpha_n^b)$.

Des algorithmes récents de résolution hors ligne, comme PBVI (Pineau *et al.*, 2003), HSVI2 (Smith & Simmons, 2005), et SARSOP (Kurniawati *et al.*, 2008), approchent la fonction de valeur pour un ensemble borné d'états de croyance B, où $B \subset \Delta$. Ces algorithmes implémentent différentes heuristiques pour explorer l'espace d'états de croyance, et différentes techniques pour mettre à jour la fonction de valeur V, par un opérateur de $backup$. La fonction de valeur est représentée par un ensemble de α-vecteurs, et contient un nombre limité à $|B|$ de α-vecteurs.

2.1 Le modèle ρPOMDP

ρPOMDP (Araya-López *et al.*, 2010) est une extension du modèle POMDP qui permet d'exprimer explicitement la réduction de l'incertitude sur certaines variables d'état à partir d'une fonction de récompense ρ. La modification majeur du modèle repose sur cette fonction de récompense qui est définie typiquement sur l'espace d'état de croyance. Sa forme diffère de la fonction classique $r(b, a) = \sum_{s \in S} r(s, a) b(s)$, puisqu'elle est plus générale. Autrement dit, elle peut ne pas dépendre de $r(s, a)$ et dépendre seulement d'une mesure de l'incertitude de b : $\rho(b) = \log_2(|S|) + \sum_{s \in S} b(s) \log_2(b(s))$, par exemple.

Il a été démontré que résoudre le ρPOMDP est possible. Plus précisément, (Araya-López *et al.*, 2010) démontre que la convexité de la fonction de valeur d'un POMDP est préservée pour un horizon de longueur N, lorsque la fonction de récompense ρ est convexe (avec $V_0 = 0$). Par contre, il semble nécessaire de donner une attention spéciale à la (possible) non linéarité de cette fonction ρ, afin de pourvoir utiliser les algorithmes de résolution exacte et approchée de l'état de l'art des POMDP moyennant quelques modifications. Il est donc possible d'approximer ρ avec une fonction linéaire par morceaux et convexe.

3 La mission d'exploration

Nous considérons un Véhicule Aérien Inhabité (UAV) qui doit détecter et reconnaître de cibles dans un environnement incertain et partiellement observable. La mission consiste à détecter et à identifier les voitures présentes dans les différentes zones qui constituent l'environnement. L'objectif est alors d'identifier l'état caché du système. En raison de l'observabilité partielle des objets dans la scène, nous modélisons cette mission sous forme de POMDP. L'UAV peut réaliser des actions de haut niveau de perception. Nous ne faisons aucune hypothèse sur le nombre de voitures présentes, ni sur leur nature. Nous considérons toutefois une seule voiture par zone. Les zones sont extraites par une pré-mission à partir d'un algorithme de traitement d'images lors d'un balayage du terrain.

Le nombre total d'états du POMDP dépend de plusieurs variables d'état qui sont discrétisées selon : le nombre de zones (N_z), le nombre d'altitudes de vol (N_h), le nombre de modèles de voiture (N_{models}) dans la base de données, ainsi qu'éventuellement un état terminal qui caractérise la fin de la mission. Nous supposons que les voitures, c'est-à-dire les cibles peuvent être dans n'importe quelle zone de l'environnement, et peuvent aussi représenter n'importe quel modèle connu de la base de données. Dans notre application, la base de données comporte 3 modèles possibles, soit $\{model_A,\ model_B,\ model_C\}$. Les variables d'états sont telles que :

— z, avec N_z valeurs possibles, qui indique la position de l'hélicoptère autonome ;
— h, avec N_h valeurs possibles, qui indique l'altitude de vol de l'hélicoptère autonome ;
— $Id_{Ta_{z_1}}$ (respectivement $Id_{Ta_{z_2}}$, $Id_{Ta_{z_3}}$, etc), avec $N_{models}+1$ valeurs possibles, qui indique l'identité ou l'absence d'un modèle de cible dans la zone 1 (respectivement dans la zone 2, dans la zone 3, etc.)

Ainsi, le nombre total d'états est donné par : $|S| = N_z \cdot N_h \cdot (N_{models}+1)^{N_z}$

Les actions de l'UAV sont : changer de zone, changer d'altitude, changer d'angle de vue. Le nombre d'actions de changement de zone dépend du nombre de zones (resp. d'altitudes) considérées. Ces actions sont appelées $go\_to(\hat{z})$ (resp. $go\_to(\hat{h})$), où $\hat{z}$ (resp. $\hat{h}$) sont la zone (resp. altitude) de destination. L'action de changement d'angle de vue, change l'angle d'observation d'une zone par rapport au centre de cette même zone. Donc, le nombre total d'actions est : $|A| = N_z + N_h + 1$.

Fonctions de transition et de récompense : basés sur des missions préalables de l'UAV, nous considérons les différentes actions du modèle comme déterministes. Toutefois le problème est toujours un POMDP puisque les observations des modèles de voitures sont probabilistes. Il a été démontré que la complexité de résolution d'un POMDP est essentiellement liée aux observations probabilistes (Sabbadin *et al.*, 2007).

Chacune des actions du modèle engendre, à la fin de son exécution, une observation par rapport à la zone dans laquelle l'UAV se trouve. Une fois l'action exécutée, une image est extraite automatiquement et donne ensuite lieu à une traitement d'image qui renvoie un des symboles considérés dans le modèle d'observation. Comme la caméra est fixe, il est important de contrôler les déplacements de l'UAV dans le but d'observer différentes parcelles de l'environnement.

action $go\_to(\hat{z})$ *(*$go\_to(\hat{h})$*) :* cette action amène l'UAV à la zone (resp. altitude) désirée. Le coût associé à cette action dépend de la consommation de carburant qui repose sur la distance parcourue par l'UAV. Les coordonnées de zones sont connues après le balayage du terrain, et sont utilisées dans le calcul de ce coût que nous appelons $C_{z,\hat{z}}$ (resp. $C_{h,\hat{h}}$). Avec $C_{z,\hat{z}} > C_{h,\hat{h}}$.

action $change\_view$ *:* cette action est une action de haut niveau qui change l'angle de vue de l'UAV. Il est a noter que nous ne faisons aucune hypothèse sur l'orientation des voitures dans les différentes zones. En ce sens cette action ne change pas l'état du POMDP, puisque celui-ci dépend uniquement des variables z et h. Toutefois, cette action de haut niveau, qui est traduite par le composant de l'architecture robotique en charge de l'exécution des actions, fait l'UAV changer son angle de vue par rapport au centre de la zone observée (environs 10 dégrées). Ceci nous permet de ne pas expliciter plusieurs états du POMDP, c'est-à-dire nous n'avons pas besoin de modéliser l'orientation des voitures par des variables d'état, en restant assez réalistes. Le coût C_v de cette action dépend de l'angle de déplacement, et est généralement inférieur aux coûts de changement de zone et d'altitude ($C_{z,\hat{z}} > C_{h,\hat{h}} > C_v$).

Fonction d'observation : le modèle POMDP requiert une description probabiliste fidèle des effets des actions et des observations. Ceci est souvent difficile à obtenir en pratique. Dans notre application le modèle d'observation a été automatiquement appris à partir des images collectées lors de vols préalables. Ces images nous ont permis de réaliser un apprentissage hors ligne (en laboratoire) supervisé de ce modèle d'observation. Nous avons utilisés ici la même procédure que celle décrite dans (Carvalho Chanel *et al.*, 2012) afin de définir une probabilité a priori des possibles sorties symboliques du traitement d'image pour

un état donné.

Comme dans (Carvalho Chanel *et al.*, 2012), l'observation des modèles de voitures est basé sur l'algorithme de traitement d'images de (Saux & Sanfourche, 2011), qui est actuellement embarqué sur l'architecture de l'UAV. Pour notre application, 5 observations sont possibles ($|\Omega| = 5$) au dessus de la zone courante : *{voiture non détectée, voiture détectée mais non identifiée, voiture identifiée comme modèle A, voiture identifiée comme modèle B, voiture identifiée comme modèle C}*.

4 Critère mixte

Sur l'hypothèse que la fonction de récompense est effectivement convexe, nous proposons un critère mixte qui couple à la fois les récompenses associées aux paires état-action avec les récompenses associées à une mesure de l'incertitude de l'état de croyance. Ce critère a été initialement proposé dans (Carvalho Chanel *et al.*, 2010), par contre le schéma algorithmique utilisé ainsi que l'application robotique pour l'évaluation de ce critère ici sont différents. Il est à noter que l'approche présentée ici est un cas particulier du modèle général ρPOMDP.

Dans la suite nous définissons ce critère mixte, et nous démontrons que la politique associée à ce critère peut se calculer par programmation dynamique. Nous présentons par la suite une implémentation d'un algorithme basé sur un schéma d'optimisation d'itération approchée sur la valeur, que nous avons adapté au cas de ce critère mixte.

Définition 1 (Critère mixte)
Soit π une politique définie sur l'état de croyance. Nous définissons alors un critère d'optimisation qui se décompose en : l'espérance mathématique de la somme pondérée des récompenses attribuées aux actions choisies à laquelle s'ajoute l'espérance mathématique de la somme pondérée des entropies (négatives) des états de croyance successifs. Ces deux termes sont ensuite pondérés par une constante $\lambda \in [0, 1]$:

$$J^{\pi}(b) = (1-\lambda)V^{\pi}(b) + \lambda H^{\pi}(b), \text{ avec} \tag{4}$$

$$V^{\pi}(b) = E_{\pi}\left[\sum_{t=0}^{\infty}\gamma^t r(b_t, \pi(b_t))\middle| b_0 = b\right] \text{ et } H^{\pi}(b) = E_{\pi}\left[\sum_{t=0}^{\infty}\gamma^t H(b_t)\middle| b_0 = b\right]$$

Notons que $V^{\pi}(b)$ est le critère γ-pondéré des POMDP, puisque : $r(b_t, \pi(b_t)) = \sum_{s\in S} b_t(s) r(s, \pi(b_t))$. Notons aussi, que $H^{\pi}(b)$ correspond au critère de performance déjà proposée par (Deinzer *et al.*, 2003). En revanche, nous noterons ici l'entropie de l'état de croyance telle que : $H(b) = \sum_{s\in S} b(s)\log(b(s))$, de sorte que celle-ci est négative (pas de signe de moins). Cela permet à la fonction de valeur J^{π} d'être convexe. De plus, l'entropie définie en tant que telle, joue le rôle d'une pénalisation dans le critère puisque $H(b) \leq 0, \ \forall b \in \Delta$. Ainsi, la somme des deux termes forme une fonction convexe qui correspond à la définition du critère de performance donné par (Mihaylova *et al.*, 2002).

Il nous est alors possible de définir une équation de Bellman relative à ce critère, comme cela est démontré dans le théorème suivant :

Théorème 1 (Equation de Bellman pour le critère mixte.)
Le principe de Bellman appliqué au critère mixte pour une politique stationnaire π est donné par :

$$J^{\pi}(b) = (1-\lambda)r(b,\pi) + \lambda H(b) + \gamma \sum_{o\in\Omega} p(o|b,\pi) J^{\pi}(b^o_{\pi}) \tag{5}$$

preuve dans l'annexe A.1.

Proposition 1 (Équation de valeur optimale pour le critère mixte.)
La fonction de valeur optimale $J^*(b)$ est la solution de l'équation de Bellman :

$$J^*(b) = \max_{a\in A}\left\{(1-\lambda)r(b,a) + \lambda H(b) + \gamma \sum_{o\in\Omega} p(o|b,a) J^*(b^o_a)\right\} \tag{6}$$

preuve dans l'annexe A.2.

L'application de la programmation dynamique à l'équation de Bellman 5 permet de calculer une politique où la récompense immédiate se trouve pénalisée par l'entropie de l'état de croyance. Cette pénalisation sera d'autant plus importante que la distribution sur les états est peu précise. Ce calcul permet d'introduire les opérateurs de Bellman et de programmation dynamique relatifs à ce critère mixte.

Définition 2 (Opérateur de Bellman pour le critère mixte)
L'opérateur de Bellman, noté $\mathcal{W}^\pi : \mathbb{R}^\Delta \to \mathbb{R}^\Delta$ *sachant que* $\mathbb{R}^\Delta \Leftrightarrow \{f : \Delta \to \mathbb{R}\}$. *Donc, pour tout* $J \in \mathbb{R}^\Delta$ *est défini par :*

$$\mathcal{W}^\pi J(b) = (1-\lambda)\sum_{s\in S} r(s,\pi)b(s) + \lambda \sum_{s\in S} b(s)\log(b(s)) + \gamma \sum_{o\in\Omega} p(o|b,\pi)J(b^o_\pi). \tag{7}$$

L'opérateur de programmation dynamique $\mathcal{W} : \mathbb{R}^\Delta \to \mathbb{R}^\Delta$ *est défini tel que :*

$$\mathcal{W}J(b) = \max_{a\in A}\left[(1-\lambda)\sum_{s\in S} r(s,a)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \gamma\sum_{o\in\Omega} p(o|b,a)J(b^o_a)\right]. \tag{8}$$

Bien que toujours convexe, le critère mixte n'est plus une fonction linéaire. Comme cela a été déjà discuté auparavant, (Araya-López *et al.*, 2010) remarque que si la fonction convexe de récompense n'est pas linéaire, il est possible de l'approximer à partir d'une fonction linéaire par morceaux et convexe (PWLC). Grâce à cela, nous pouvons utiliser des algorithmes approchés issus de l'état de l'art du domaine POMDP en les adaptant.

4.1 Approximation linéaire de la fonction de récompense du critère mixte

Il est possible de démontrer que la fonction de récompense peut être représentée par un α-vecteur au point b^*, à partir d'une approximation linéaire du premier ordre de celle-ci. De la sorte que l'intégration de l'entropie dans la fonction de récompense n'empêchera pas de paramétrer la fonction de valeur par des α-vecteurs aux points $b^* \in \Delta$.

Nous définissons donc α_c, l'α-vecteur classique qui maximise la valeur $V(b)$, pour $b \in \Delta$. Cet α-vecteur est constitué des récompenses associées aux paires état-action et permet de modéliser la fonction de valeur $V(b)$ sous la forme vectorielle telle que : $V(b) = \alpha_c \cdot b$.

La prise en compte de l'entropie modifie l'expression de la récompense immédiate qui devient : $\rho(b,a) = (1-\lambda)\sum_{s\in S} r(s,a)b(s) + \lambda\sum_{s\in S} b(s)\log b(s)$ et, sous forme vectorielle nous définissons cette récompense immédiate par : $\rho(b,a) = \alpha_m \cdot b$. Si l'on cherche à dériver la nouvelle récompense immédiate $\rho(b) = \alpha_m \cdot b$, pour toute action a, on obtient :

$$\rho(b) = \alpha_m \cdot b = ((1-\lambda)\alpha_c + \lambda\log(b)) \cdot b \tag{9}$$

$$\frac{\partial\rho(b)}{\partial b} = (1-\lambda)\alpha_c + \lambda(\log b + \underline{\log}\, e) \tag{10}$$

où $\log b \equiv [\log b(s_1) \ldots \log b(s_n)]$ est le vecteur dont la i-ième composante vaut $\log b(s_i)$, et $\underline{\log}\ e \equiv [\log e \ldots \log e]$ un vecteur pour lequel toutes les composantes valent $\log e$. Ainsi, nous vérifions au passage que la dérivée de ρ par rapport à b n'est pas linéaire. Cependant, un développement en série au premier ordre permet d'obtenir une approximation linéaire au voisinage d'un point $b^* \in \Delta$. Dans ce cas, ρ et son gradient par rapport à b sont évalués au point $b^* \in \Delta$.

$$\rho(b) = \rho(b^*) + \frac{\partial\rho(b)}{\partial b}|_{b=b^*} \cdot (b - b^*) \tag{11}$$

$$= ((1-\lambda)\alpha_c + \lambda(\log b^*)) \cdot b + \lambda\underline{\log}\, e \cdot b - \lambda\underline{\log}\, e \cdot b^* \tag{12}$$

En utilisant la condition que pour tout b, la somme des composantes vaux 1 ($\sum_i b_i = 1$) étant donné que b est une distribution de probabilité, on voit que $\underline{\log} e \cdot b = \log e$ et donc $(\lambda\underline{\log}\, e \cdot b - \lambda\underline{\log}\, e \cdot b^*) = 0$. Ainsi il vient :

$$\rho(b) = ((1-\lambda)\alpha_c + \lambda(\log b^*)) \cdot b, \text{ sous forme vectorielle } \rho(b) = \alpha_m^{b^*} \cdot b \tag{13}$$

Ceci démontre que nous pouvons utiliser $\alpha_m^{b^*}$ comme α-vecteur de tout point b au voisinage de b^*. Ce résultat est important puisqu'il démontre que les algorithmes de type *point-based* peuvent être appliqués mais au prix d'une approximation linéaire des variations de J. Il apparaît également que pour chaque $b^* \in \Delta$ exploré l'algorithme devra conserver l'α-vecteur associé.

Comme nous voudrons résoudre et implémenter le ρPOMDP reposant sur le critère proposé. Pour faire cela, nous avons dû modifier l'algorithme PBVI (Pineau *et al.*, 2003) afin de s'accommoder à cette nouvelle formulation. Le choix de l'algorithme PBVI se justifie par le fait qu'il offre une méthode d'exploration stochastique de l'espace des états de croyance, ce qui évite de modifier les heuristiques de recherche dans cet espace qui sont utilisées habituellement par d'autres algorithmes, par exemple, tels que HSVI (Smith & Simmons, 2005) et SARSOP (Kurniawati *et al.*, 2008).

Algorithme 1: PBVI pour le critère mixte

entrée : POMDP, N, $N_{\mathcal{B}}$
sortie : fonction de valeur J

1 $\mathcal{B} \leftarrow b_0$;
2 **while** $|\mathcal{B}| < N_{\mathcal{B}}$ **do**
3 Étendre $\mathcal{B}$;
4 Initialiser $J_0 \leftarrow \emptyset$, $n = 0$;
5 **repeat**
6 $n = n + 1$;
7 **if** $J_{n-1} \neq \emptyset$ **then**
8 Calculer toutes les projections $\Gamma_m^{a,o}$ de J_{n-1} (cf. (Pineau *et al.*, 2003)) ;
9 **for** $b \in \mathcal{B}$ **do**
10 $J_n \leftarrow \bigcup backup_m(b)$;
11 **until** $n > N$ ou $||\max_{\alpha_n \in V_n} \alpha_n \cdot b - \max_{\alpha_{n-1} \in V_{n-1}} \alpha_{n-1} \cdot b|| < \epsilon, \forall b \in \mathcal{B}$;
12 **return** J_n ;

4.2 Adaptation de l'algorithme PBVI

L'algorithme PBVI repose sur une méthode d'optimisation de type *point-based*. Celle-ci suppose qu'en mettant à jour non seulement la valeur mais aussi le gradient (le α-vecteur) pour chaque $b \in \mathcal{B}$, la politique calculée pourra être utilisée pour d'autres points de l'espace de croyance qui n'appartiennent pas à l'ensemble B. L'ensemble des états de croyance est composé de points considérés comme atteignables en suivant une politique d'action arbitraire depuis un état de croyance initial b_0. Comme la plupart des algorithmes de type *point-based* procèdent à une mise à jour de la fonction de valeur du POMDP à partir d'un sous-ensemble d'états de croyance $\mathcal{B} \in \Delta$, nous considérerons que les ensembles de points utilisés par l'approximation linéaire d'une part et l'algorithme d'autre part sont les mêmes (Araya-López *et al.*, 2010)

L'algorithme 1 décrit la procédure pour calculer les nouveaux α-vecteurs représentés par l'approximation linéaire du premier ordre. Avant de présenter le nouvel opérateur de mise à jour de la valeur – *backup* – nous souhaitons aborder un point technique qui nous semble essentiel dans le calcul du logarithme de b.

L'entropie définie négative, n'est pas une fonction lipschitzienne, c'est-à-dire une fonction continue à variation et à dérivée bornées. On voit clairement que $\log x \to -\infty$ quand $x \to 0$. Ceci pose un problème numérique lors de la résolution. Afin de s'en affranchir, nous proposons de borner inférieurement les composantes $\log b(s_i) = -\infty$ du vecteur $\log b$ et de définir un seuil, de façon à approcher la composante $(-\infty)$ du gradient de la valeur d'un état de croyance par une constante suffisamment négative (assurer la pente). Il est à noter que cela ne change pas la valeur de l'état de croyance, étant donné que $x \log x \to 0$ quand $x \to 0$. Ainsi : $\log b(s_i) = \log b(s_i),\ \text{si } b(s_i) > 10^{-300}$, sinon $\log b(s_i) = \log 10^{-300}$.

L'adaptation de l'algorithme PBVI (cf. l'algorithme 1) implique donc une modification de la mise à jour de la valeur ($backup_m$) (ligne 10). Dans ce cas, nous rappelons que la récompense au voisinage d'un point $b \in \mathcal{B}$ peut être approchée par :

$$r_m^a = ((1-\lambda)r_a + \lambda \log b) \cdot b = \alpha_m^{b,a} \cdot b \tag{14}$$

où $r_a(s) = r(s,a)$. De cette façon le nouvel opérateur de $backup$ peut être défini par :

$$backup_m(b) \quad = \quad \arg\max_{\alpha_b^a \in \Gamma_b^a} b \cdot \alpha_b^a, \text{ avec :} \tag{15}$$

$$\Gamma_b^a \quad \leftarrow \quad r_m^a + \gamma \sum_o \arg\max_{\alpha_i^{a,o} \in \Gamma_m^{a,o}} b \cdot \alpha_i^{a,o} \tag{16}$$

Les projections $\Gamma_m^{a,o}$ sont calculées de la même façon que dans PBVI (Pineau *et al.*, 2003). Nous précisons une fois de plus que le calcul de mise à jour de la valeur, réalisée pour ce critère, impose que $|J_n| = |\mathcal{B}|$, c'est-à-dire que le nombre de α-vecteurs qui constitue la fonction de valeur à l'instant n est égale au nombre d'états de croyance atteignables. Ceci est dû à l'approximation linéaire du premier ordre qui permet de déterminer le gradient $J(b)$ pour $b \in \mathcal{B}$, à partir d'un α-vecteur particulier qui dépend du vecteur $\log b$.

Comme pour PBVI, la génération de points $b \in \mathcal{B}$ est ici faite de façon stochastique. Toutefois, cette génération de points doit être réalisée au préalable à toute résolution et non au fur et à mesure des itérations. Nous souhaitons ainsi éviter que des erreurs d'approximation de la valeur des états de croyance, dû à l'approximation du premier ordre, viennent fausser l'ensemble de la démarche.

Nous notons aussi que la valeur associée à un état de croyance dans cette méthode de résolution peut en fait osciller au delà d'un certain seuil. Ceci a été vérifié notamment par l'expérience lors de nos tests en simulation. Même si l'opérateur de Bellman est une contraction dans l'espace des fonctions de valeur qui conduit in fine à un point-fixe pour ce critère mixte, pour la méthode de résolution ici décrite, et qui n'est pas exacte, l'on peut seulement garantir que l'erreur d'approximation peut être bornée étant donné la densité d'un ensemble $\mathcal{B}$. (Araya-López *et al.*, 2010) démontre que l'on peut réduire la borne d'erreur finale pour la rendre aussi petite que l'on souhaite : plus dense est l'ensemble $\mathcal{B}$ plus l'erreur sera petite.

4.3 Évaluation par variation de paramètre : Approche multi-critère

Pour évaluer le critère mixte, des politiques stationnaires ont été calculées pour différents valeurs du facteur de pondération λ. De la sorte, nous analysons le critère défini précédemment selon une approche multi-objectif (Deb, 2001). En effet, celui-ci peut être vu comme un compromis entre l'acquisition d'information que l'on cherche à maximiser et le coût des actions nécessaires à mettre en œuvre pour cette prise d'information. L'objectif consiste donc à construire le front de Pareto associé à ces deux critères antagonistes et à déterminer l'ensemble de solutions non-dominées, c'est-à-dire toutes les solutions meilleures que les autres sur au moins un objectif. Lorsqu'on est en présence de plusieurs objectifs, il est rare qu'une solution permette de les optimiser simultanément. La plupart du temps, l'amélioration d'un objectif se fait au détriment d'un autre (Deb, 2001).

Le front de Pareto constitue un outil d'aide au choix d'une solution en fonction des critères que l'on cherche à minimiser ou maximiser. La construction de celui-ci est ici réalisée en échantillonnant plus ou moins finement le paramètre de pondération λ et en résolvant pour chaque valeur de celui-ci le problème d'optimisation du critère mixte par programmation dynamique. Par conséquent, nous avons optimisé des politiques pour chaque valeur de λ en exploitant l'algorithme PBVI modifié. L'ensemble de résultats nous permet de tracer une approximation de la frontière de l'enveloppe convexe de l'ensemble des solutions, ou en quelque sorte, l'allure du Front de Pareto, sachant que chaque politique calculée correspond à la maximisation des critères pour une valeur de λ.

Nous avons réalisé l'évaluation multi-critère sur la mission d'exploration. L'agent autonome doit se déplacer dans les différentes zones pour obtenir de l'information par rapport à la zone observée. La diminution de l'incertitude de son état de croyance devient le but de la mission. Nous rappelons que le critère mixte permettra de gérer le compromis entre la prise d'informations de l'environnement et les coûts associés aux différentes actions. Ces coûts sont ici proportionnels aux distances parcourues par l'hélicoptère. Nous avons généré le modèle POMDP de cette mission d'exploration. Nous admettons que l'environnement est constitué de 3 zones distinctes séparées de 70 mètres les unes des autres. Initialement l'hélicoptère se trouve dans la zone 1 à une altitude de vol de 30 mètres, et aucune connaissance a priori sur l'identité ou la présence de cibles n'est disponible. Ce manque d'information se traduit par une distribution de probabilité uniforme sur les 64 états initiaux possibles.

Des politiques ont été calculées pour différentes valeurs de λ, et pour un ensemble d'états de croyance $\mathcal{B}$, tel que $|\mathcal{B}| = 5000$. Le temps moyen pour le calcul des politiques a été de 5 heures, pour $\epsilon \leq 2$. La valeur du seuil ϵ peut être considérée comme lâche, mais nous avons fait ce choix dû à l'approximation linéaire du premier ordre. Selon la valeur de λ choisie l'algorithme a tendance à osciller à partir d'un certain seuil. Ainsi, par l'expérience nous avons fixé le critère d'arrêt à 2.

Nous avons tracé le front de Pareto à partir de 12000 simulations de différentes politiques avec un horizon

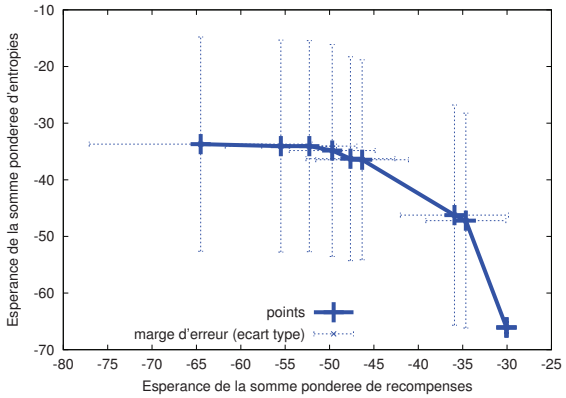
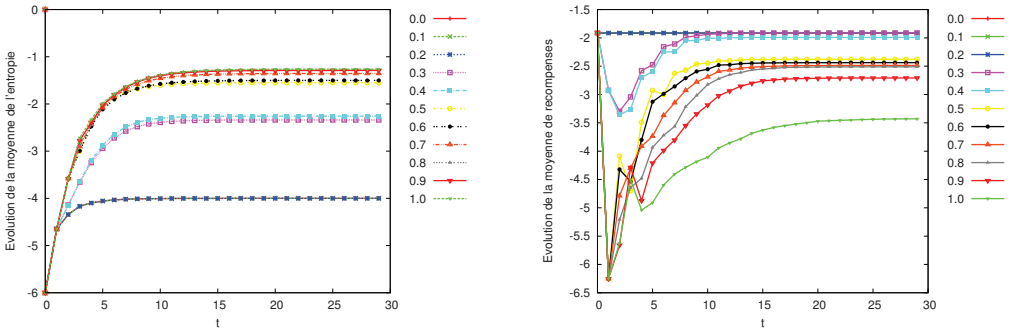


FIGURE 1 – Front de Pareto pour la mission d'exploration – $\lambda \in \{0, 0.1, 0.2, ..., 0.9, 1.0\}$.

.



(a) Moyenne de l'entropie de l'état de croyance à chaque instant t.

(b) Moyenne de la récompense à chaque instant t.

FIGURE 2 – Moyennes de l'entropie de l'état de croyance et de récompenses.

de 30 étapes de décision. Le front de Pareto obtenu est tracé sur la figure 1. Nous vérifions, que l'amélioration de l'un des deux objectifs (récompenses sur les états ou entropie de l'état de croyance) se fait au détriment de l'autre. Sur cette même figure, est également superposé l'écart type de chaque critère, qui permet de mettre en évidence la grande variabilité d'une exécution à une autre (problème-dépendant).

L'évolution de la moyenne de l'entropie est tracée sur la figure 2(a), et l'évolution moyenne des récompenses est montré dans la figure 2(b)). Nous pouvons ainsi voir que plus le poids de l'entropie est élevé, plus elle se trouve effectivement optimisée, ceci au prix de gains moyens moins élevés. Ces moyennes sont calculées selon :

$$H_t = \frac{1}{k}\sum_{i=0}^{k}\sum_{s\in S} b_t(s)\log(b_t(s)) \quad \text{et} \quad V_t = \frac{1}{k}\sum_{i=0}^{k} r(s_t, \pi(b_t)) \tag{17}$$

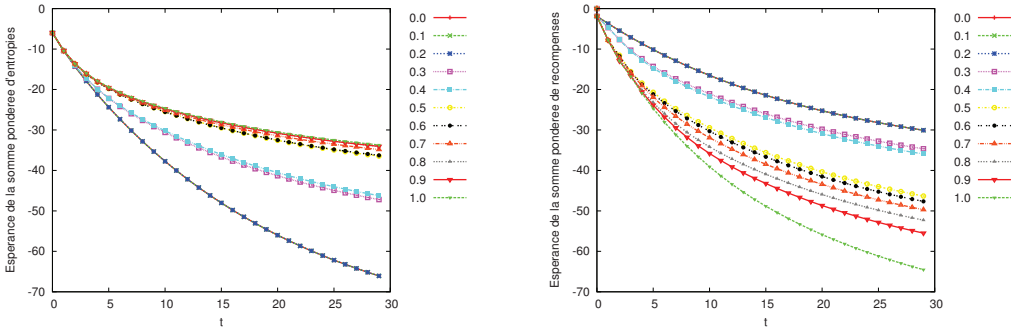
où k représente le nombre de simulations. Il est à noter que le simulateur connait l'état caché du système en attribuant les récompenses à chaque étape de décision.

Nous montrons aussi l'espérance de la somme pondérée de récompenses et d'entropies sur la figure 3 calculées comme :

$$H^{\pi}(b) = E_{\pi}\left[\sum_{i=0}^{t}\gamma^i H(b_i)|b_0 = b\right] \quad \text{et} \quad V^{\pi}(b) = E_{\pi}\left[\sum_{i=0}^{t}\gamma^i r(s_i, \pi(b_i))|b_0 = b, s_i\right] \tag{18}$$

sur un passé de longueur t.

Le comportement de l'agent sera différent en fonction de λ. Dans le cas où $\lambda = 0$, le critère optimisé se confond avec le critère classique du POMDP, qui ne prend pas en compte l'entropie associé à l'état de croyance, ce qui est confirmé par le fait que la politique se réduit aux deux actions (goto h_1 et goto h_2) moins coûteuses. L'entropie de l'état de croyance sur la zone 1, qui est la zone initialement observée, sera réduite ; par contre, comme aucune autre action de déplacement entre zones n'est présente dans la politique, l'entropie de l'état de croyance sur les autres zones ne sera pas réduite. Ceci explique le fait que la courbe rouge associée au symbole (–|–) sur la figure 2(a) reste bloquée à la valeur -4 à partir d'un certain temps (instant de décision t). Pour les cas où $\lambda = 0.1$ et 0.2, l'entropie n'a pas un poids suffisant vis-à-vis des récompenses associées au paires état-action moyennées sur l'état de croyance : ceci explique le fait que la politique confère au système le même comportement que pour $\lambda = 0$. Lorsque que nous faisons tendre λ vers 1, la politique fournit à l'agent un comportement de plus en plus investigateur malgré le coût des actions : plus l'agent acquiert de l'information, plus l'entropie de croyance est diminuée au prix d'un coût d'actions plus important.



(a) Espérance de la somme pondérée d'entropies de l'état de croyance pour chaque instant t.

(b) Espérance de la somme pondérée de récompenses pour chaque instant t.

FIGURE 3 – Espérance de la somme pondérée d'entropies de l'état de croyance et de récompenses.

Toutes ces évaluations nous mènent à conclure que la détermination du bon λ à utiliser dépendra de l'importance que le concepteur du système autonome souhaitera donner aux différents critères. De plus, pour ce critère mixte, le moment de prise de décision finale, c'est-à-dire l'instant auquel le rapport de l'état caché du système est effectué, revient au utilisateur au moyen d'un seuil d'arrêt lorsque celui-ci exécute la politique. Autrement dit, l'utilisateur devra fixer un seuil η à partir du quel il arrêtera l'exécution de la politique une fois que $b(s_i) > \eta$, de façon à identifier l'état caché s_i.

5 Modèle POMDP de la mission d'exploration avec des buts fictifs

Cette nouvelle approche consiste à ajouter au modèle POMDP classique des actions de rapport de l'état du système, appelées *report* s_i, de sorte qu'une action supplémentaire est désormais offerte à l'agent par état. Avec ces actions de type *report*, on peut vouloir se ramener à une modélisation sous forme de POMDP avec une fonction de récompense dépendant des paires état-action. Autrement dit, à la place de fixer η, on cherche à donner une récompense suffisamment grande au fait de rapporter correctement le véritable état. Plus précisément, nous rajoutons au modèle autant d'actions *report* qu'il y a d'états possibles.

action report s_i **:** cette action permet à l'hélicoptère autonome d'affirmer que l'état réel du système qu'il observe est l'état s_i. Cette action conduit l'agent à l'état terminal du POMDP.

— fonction de récompense : $R(s, \texttt{report}\ s_i) = \mathbb{I}_{\{s=s_i\}} R_r - \mathbb{I}_{\{s\neq s_i\}} C_r$, où $R_r > 0$ représente la récompense associée à l'action de rapporter l'état s_i lorsque celui-ci correspond effectivement à l'état caché du système que l'agent observe, et $C_r \gg 0$ le coût associé à cette action si l'état vrai n'est pas s_i.

Notons que, dans cette approche, il est nécessaire d'ajouter une action par état, de façon à garantir que l'agent ne soit récompensé que pour l'état s_i concerné par l'action *report* s_i. L'avantage d'un tel modèle est que nous pouvons appliquer les algorithmes existants pour résoudre le POMDP et faire ensuite des comparaisons avec les solutions obtenues par moyen d'un critère mixte.

D'autre part, nous rappelons que la complexité de la procédure de mise à jour exacte de la valeur est liée au nombre d'α-vecteurs qui composent V_{n-1}, au nombre d'actions $|A|$ et qu'elle dépend de manière exponentielle du nombre $|\Omega|$ d'observations : $|V_{n+1}| = |A||V_n|^{|\Omega|}$. Ainsi, même si le facteur $|A|$ est relativement négligeable au regard de $|\Omega|$, nous pouvons être amenés à ajouter un nombre important d'actions au modèle en plus des actions standards de déplacement et de changement d'angle de vue. En revanche, si on se ramène au POMDP classique on peut appliquer des algorithmes efficaces basés sur la recherche heuristique, tels que HSVI [1] (Smith & Simmons, 2005) et SARSOP [2] (Kurniawati *et al.*, 2008), qui focaliseront l'optimisation de la valeur sur un petit nombre d'actions prometteuses selon l'état de croyance.

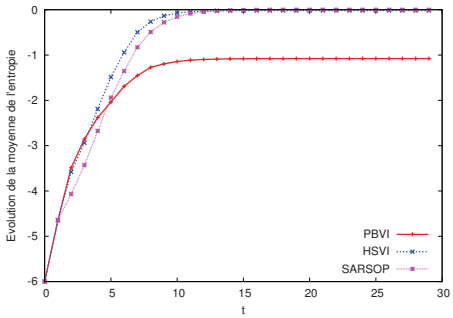
5.1 Évaluation du modèle modifié par ajout de buts fictifs

Pour évaluer cette approche, nous avons calculé des politiques à partir de différents algorithmes : PBVI (avec $|\mathcal{B}| = 5000$) que nous avons ré-implémenté ; HSVI et SARSOP. La durée de calcul de politiques a été limitée à 4 heures pour HSVI et SARSOP. Au bout de 4 heures de calcul ces deux algorithmes ont atteint un $\epsilon \leq 13$ avec $|V| = 47359$ pour HSVI, et $|V| = 14783$ pour SARSOP. Ensuite, nous avons utilisé cette valeur pour ϵ comme critère d'arrêt de l'algorithme PBVI. Les résultats ont été obtenus au bout de 43 heures de calcul, résultant en une fonction de valeur de taille $|V| = 2281$.

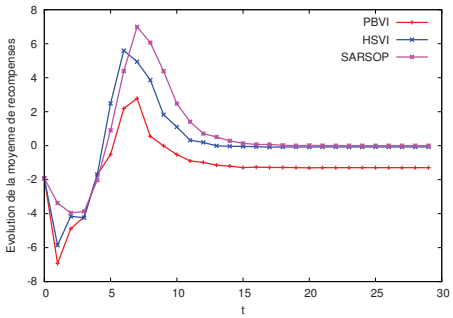
Nous pouvons constater que les algorithmes basés sur la recherche heuristique, c'est-à-dire HSVI et SARSOP, ont des performances supérieures à celle de l'algorithme PBVI qui repose sur une recherche stochastique. Il est indispensable de focaliser la recherche sur un nombre réduit d'actions. Nous avons augmenté le nombre d'états de croyance utilisés dans l'algorithme PBVI, c'est-à-dire $\mathcal{B}$ mais notre implémentation a atteint la limite de mémoire disponible [3]. Nous tenons à faire remarquer que le temps de calcul pour ces politiques est important, dû à la quantité d'actions à évaluer.

Dans la figure 4, nous montrons la moyenne obtenue pour 12000 simulations à chaque étape de décision pour l'entropie négative et la moyenne de récompenses (équation 17). Nous montrons dans la figure 4(a) l'entropie qui permet d'évaluer l'incertitude de l'état de croyance de l'agent, illustrant ainsi l'évolution de la connaissance de celui-ci. Nous pouvons ainsi vérifier que l'incertitude de l'état de croyance de l'agent tend vers zéro au fur et à mesure qu'il acquiert de l'information. Dans la figure 4(a), l'entropie moyenne de l'état de croyance tend vers zéro pour HSVI et SARSOP, et vers -1 pour PBVI. Ceci peut être expliqué par le fait

1. disponible en `http://www.cs.cmu.edu/~trey/zmdp/`
2. disponible en `http://bigbird.comp.nus.edu.sg/pmwiki/farm/appl/`
3. La machine que nous avons utilisé est un Intel Duo Core2 avec 2Gb de mémoire et 2.13GHz.



(a) Moyenne de l'entropie de l'état de croyance à chaque instant t.



(b) Moyenne de la récompense à chaque instant t.

FIGURE 4 – Moyennes de l'entropie de l'état de croyance et des récompenses.

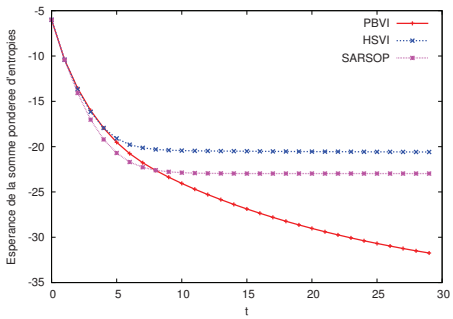
que l'algorithme PBVI, en étant limité en nombre d'états de croyance, ne dispose pas d'actions optimisées pour certains états de croyance potentiellement rencontrés lors des simulations. Il peut se retrouver ainsi dans l'incapacité d'appliquer une action optimale à certains états de croyance dans certaines simulations. L'action utilisée est alors une action sous-optimale qui ne permet pas d'acquérir l'information manquante vis-à-vis de certains états de croyance dans lesquels se retrouve l'agent.

Dans la figure 4(b) l'évolution moyenne des récompenses nous montre que l'agent décide de rapporter l'état caché du système au bout de 10 étapes de décision en moyenne. Ceci est mis en évidence par le pic observé sur les fonctions de récompense entre les étapes de décision 5 et 10. La politique obtenue par l'algorithme PBVI a une courbe de valeur inférieure à celles des algorithmes HSVI et SARSOP. Cette observation s'explique par la même analyse que celle faite précédemment. Nous avons tracé les courbes jusqu'à $t = 30$ pour montrer les résultats moyens.

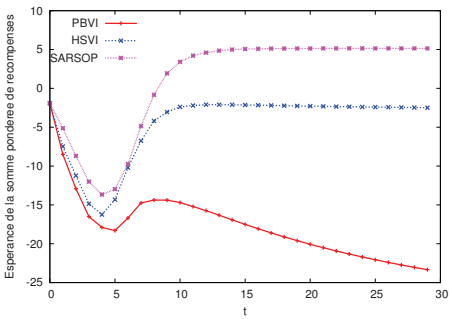
Il est à noter que le moment de prise de décision *finale* (action *report* s) est déterminé par le choix *a priori* de modélisation des C_r et R_r, ce qui correspond à une alternative au choix *a priori* d'une valeur de seuil η sur l'état de croyance.

Dans la figure 5, nous montrons les courbes de l'espérance de la somme pondérée d'entropies négatives H^π et de récompenses V^π (équation 18) pour chaque horizon t. L'espérance de la somme pondérée des récompenses est en effet le critère optimisé par la politique du POMDP lorsque celle-ci est simulée un nombre suffisant de fois. Ce critère est généralement utilisé comme mesure de la performance d'une politique. D'autre part, l'espérance de la somme pondérée des entropies met en évidence la vitesse de convergence de la croyance de l'agent. Une fois de plus les résultats caractérisant l'algorithme PBVI sont inférieurs à ceux de HSVI et SARSOP, toujours pour la même raison : PBVI est limité en nombre d'états de croyance et ne dispose donc pas d'actions optimales pour certains états de croyance rencontrées au cours des simulations.

Ces résultats nous permettent de conclure que cette approche, par ajout de buts fictifs par moyen d'actions supplémentaires *rend possible* l'utilisation directe du critère classique ainsi que des algorithmes classiques de résolution des POMDP. Nous pouvons modéliser des récompenses sur ces actions *report* par des paires état-action, et résoudre ainsi le problème de perception active avec le formalisme classique de POMDP. Le point faible de la méthode concerne le nombre d'états qui conditionne de la même manière celui des actions. Ceci peut être un facteur limitant, au regard de l'algorithme de résolution employé. Toutefois, nous avons vérifié que l'utilisation des algorithmes efficaces basés sur la recherche heuristique nous permet de surmonter ce problème. L'autre point faible de cette approche est qu'il faut choisir a priori la structure de récompense R_r et C_r pour ensuite évaluer le comportement en termes de taux de bonnes classifications. Ce



(a) Espérance de la somme pondérée d'entropies de l'état de croyance pour chaque instant t.



(b) Espérance de la somme pondérée de récompenses pour chaque instant t.

FIGURE 5 – Espérance de la somme pondérée des entropies de l'état de croyance et des récompenses.

choix de récompenses R_r et C_r, même en étant en rapport avec les coûts de déplacement, reste empirique.

6 Comparaison entre les approches : ajout d'actions et critère mixte

Étant donné que le critère mixte que nous avons proposé a été implémenté en modifiant l'algorithme PBVI, nous désirons comparer les stratégies obtenues sur la base de cet algorithme. Cette comparaison repose sur, d'une part, la stratégie obtenue à partir du critère mixte pour différentes valeurs de la pondération λ et la stratégie obtenue pour le modèle modifié qui intègre des actions de type *report*, d'autre part. Du fait des spécificités et des différences importantes entre les deux approches, le recours à l'algorithme de résolution PBVI uniquement permet de mener une comparaison objective des politiques obtenues. Pour les deux approches nous avons fixé la taille de l'ensemble d'états de croyance à 5000.

En revanche, pour l'approche par ajout de buts fictifs, la performance de l'algorithme PBVI est relativement pauvre comparée à celle des autres algorithmes. Afin de contourner cette limitation, nous avons moyenné nos résultats sur l'ensemble des trajectoires pour lesquelles la politique a rapporté un état dans le temps de mission imparti, fixé à 15 étapes de décision.

Pour le critère mixte, le moment de prise de décision finale, c'est-à-dire l'instant auquel le rapport de l'état caché du système est effectué, revient à l'utilisateur (expert) lorsque celui-ci met en œuvre la politique. Nous savons que ce choix est arbitraire, nous rappelons ici, que le choix de la valeur de R_r et C_r est également arbitraire. Ainsi, nous comparons par ailleurs différents seuils de prise de décision finale. Cette décision finale s'appuie sur l'état de croyance de l'agent. Nous supposons qu'une fois que l'état de croyance de l'agent à atteint certain niveau, c'est-à-dire que $b(s) > \eta$), l'agent décidera de rapporter l'état s concerné et recevra en conséquence une récompense (selon que la décision soit bonne ou mauvaise), stoppant ainsi l'application de la politique. La récompense attribuée est la même qui a été fixée pour le modèle par ajout d'actions. On obtient ainsi une moyenne de la somme pondérée de récompenses qui peut être comparée à celle du modèle par ajout d'actions. Il est à noter que nous avons ainsi une comparaison qui nous semble juste entre deux manières relativement arbitraires de modéliser le comportement de l'agent, qui est obtenu par résolution d'un modèle classique (ajout d'action) ou modifié (critère mixte) de POMDP.

Éléments de comparaison

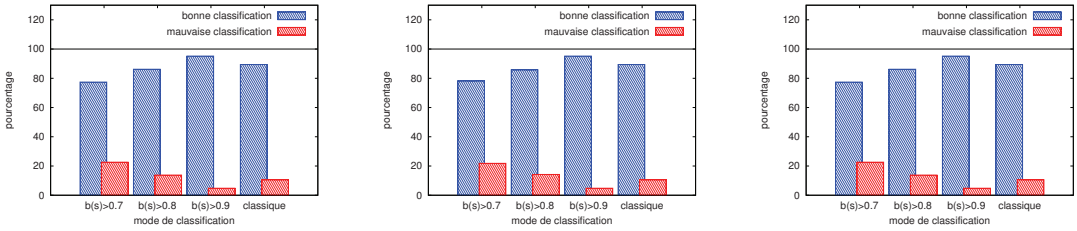
Pour comparer les approches, nous avons évalué trois critères de performance : (1) le pourcentage de classifications bonnes (resp. mauvaises classifications), c'est-à-dire le nombre relatif de fois que l'agent a correctement classé l'état caché du système (resp. de manière incorrecte) ; (2) l'évolution de la moyenne de l'entropie de l'état de croyance et l'espérance de la somme pondérée des entropies, afin de vérifier la vitesse de convergence de la croyance de l'agent ; (3) l'évolution de la moyenne des récompenses et l'espérance de la somme pondérée de celle-ci, afin de vérifier le coût engendré globalement par la politique ;

Le résultats moyens ont été calculés pour 1000 simulations de la politique à partir d'un état de croyance initial qui correspond à une distribution de probabilité uniforme. De plus, pour cette comparaison nous avons choisi trois valeurs du facteur de pondération λ, $\lambda = \{0.5, 0.7, 0.9\}$, et trois seuils de prise de décision finale, $b(s) > 0.7$, $b(s) > 0.8$, $b(s) > 0.9$.

6.1 Résultats

Il est montré dans la figure 6 les pourcentages de bonne et de mauvaise classifications. Les différentes colonnes sont associées aux seuils utilisés pour la classification pendant la simulation de la politique obtenue pour le critère mixte et au mot clef *classique* qui se réfère au modèle modifié par ajouts d'actions qui optimise le critère classique des POMDP.

Ces résultats montrent que selon le seuil spécifié par l'utilisateur les deux approches – critère mixte et ajout d'actions – peuvent être considérées comme équivalentes, en particulier pour un seuil $b(s) > 0.8$.



(a) Pourcentage de bonne et de mauvaise classification pour $\lambda = 0.5$.

(b) Pourcentage de bonne et de mauvaise classification pour $\lambda = 0.7$.

(c) Pourcentage de bonne et de mauvaise classification pour $\lambda = 0.9$.

FIGURE 6 – Pourcentage de bonnes et de mauvaises classifications selon la valeur de λ.

De plus, pour le seuil $b(s) > 0.9$, le pourcentage de bonnes classifications est légèrement supérieur à celui obtenu à partir du critère classique relatif au modèle modifié. Il est à relever que ce résultat n'est pas nouveau. Il est connu de la communauté scientifique qui travaille dans le domaine de la perception active. Le recours à un critère mixte, comme celui défini par (Mihaylova *et al.*, 2002), permet en effet d'obtenir ce genre de performances. Toutefois, notre analyse comparative entre les deux approches est nouvelle à notre connaissance et n'a pas fait l'objet de publications.

Il est aussi à noter que le pourcentage de bonnes classifications est toujours supérieur au seuil d'arrêt choisi. De plus ces résultats permettent de confirmer que la modélisation sous forme de POMDP classique (ajout d'actions) rend complètement implicite le "réglage" du seuil de décision au travers du choix du rapport entre R_r et C_r. Pour un simple utilisateur il est plus naturel de fixer un taux de bonnes/mauvaises classifications que l'on souhaite, que de définir le rapport de récompense (R_r et C_r).

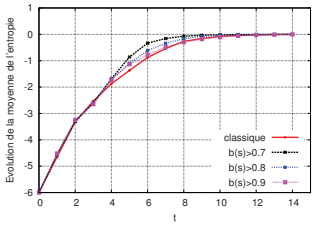
Ainsi, nous pensons que cette comparaison est un outil très constructif : l'analyse du comportement d'une politique obtenue par moyen d'un critère mixte avec un seuil de décision nous aide à définir le *bon* rapport entre R_r et C_r afin de respecter un niveau de bonnes classifications. Nous avons fixé cette récompense à 50 et le coût associé à une mauvaise classification à 100. La figure 6 montre que l'on peut chercher à faire mieux (cf. $b(s) > 0.9$), plus la récompense sera grande (50) et plus les coûts seront importants (-100) plus l'agent sera exigeant sur son niveau de croyance pour la prise de décision *finale* de classification.

La figure 7 regroupe les résultats concernant l'évolution de la moyenne de l'entropie de l'état de croyance et des récompenses. Cette comparaison est encore plus délicate à réaliser que la précédente, puisque dans le cas du critère mixte les récompenses associées à la classification ne sont pas prises en compte lors de l'optimisation. Les récompenses artificielles concernant les bonnes et les mauvaises classifications ont été ajoutées lors de la simulation de la politique.

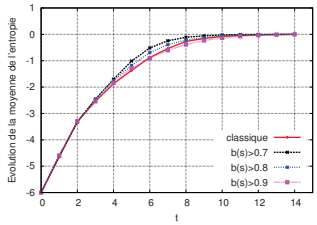
L'évolution de la moyenne de l'entropie de l'état de croyance suit quasiment les mêmes variations quelle que soit la valeur de la pondération λ. Ceci n'est pas surprenant étant donné que, pour des valeurs de λ supérieures à 0.5 (figure 2(a)), la vitesse de convergence de l'entropie reste la même. Ceci tend à démontrer une fois de plus que les deux approches – critère mixte et ajout d'actions – sont équivalentes.

La différence entre ces deux approches est illustrée sur les courbes qui représentent la moyenne des récompenses à chaque instant t (figures 8(d), 8(e) et 8(f)). Sur ces figures nous pouvons observer que l'instant de prise de décision *finale* change selon le seuil fixé (pic des fonctions). Plus l'on est exigeant vis-à-vis de la probabilité $b(s)$ associée à l'état caché, plus l'agent déclenchera sa décision *finale* tard.

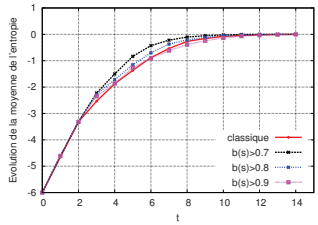
Sur la figure 8, est tracée l'espérance de la somme pondérée des entropies et des récompenses pour les différentes valeurs de la pondération λ (selon équation 18). L'espérance de la somme pondérée des récompenses atteint une valeur supérieure pour le critère mixte pour les valeurs de $\lambda = 0.5$ et $\lambda = 0.7$ avec le seuil de décision supérieur à 0.7. Ceci peut être expliqué par le fait que les coûts associés aux déplacements de l'agent pèsent plus lors du calcul de la politique. Ainsi, l'agent cherchera de l'information de manière moins coûteuse que dans le cas où $\lambda = 0.9$ par exemple.



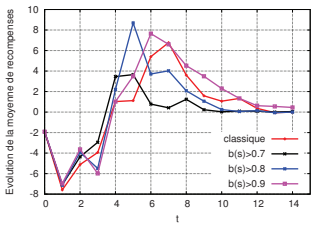
(a) Moyenne de l'entropie de l'état de croyance à chaque instant t pour $\lambda = 0.5$.



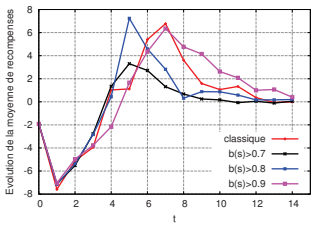
(b) Moyenne de l'entropie de l'état de croyance à chaque instant t pour $\lambda = 0.7$.



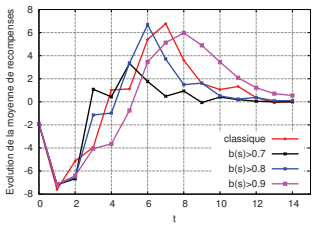
(c) Moyenne de l'entropie de l'état de croyance à chaque instant t pour $\lambda = 0.9$.



(d) Moyenne de récompenses à chaque instant t pour $\lambda = 0.5$.

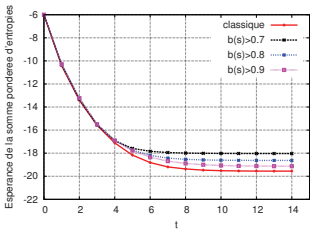


(e) Moyenne de récompenses à chaque instant t pour $\lambda = 0.7$.

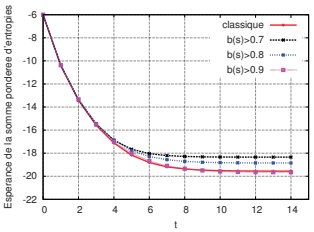


(f) Moyenne de récompenses à chaque instant t pour $\lambda = 0.9$.

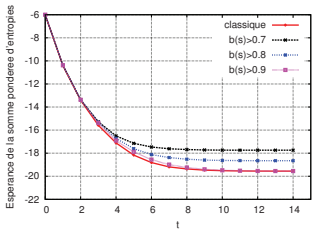
FIGURE 7 – Évolution moyenne de l'entropie de l'état de croyance et de récompenses selon la valeur de λ.



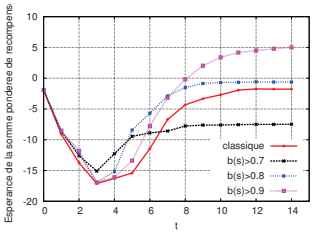
(a) Espérance de la somme pondérée d'entropies de l'état de croyance pour chaque instant t pour $\lambda = 0.5$.



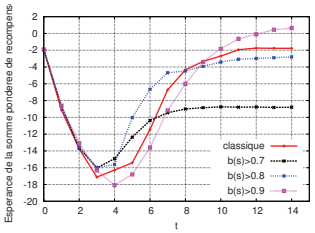
(b) Espérance de la somme pondérée d'entropies de l'état de croyance pour chaque instant t pour $\lambda = 0.7$.



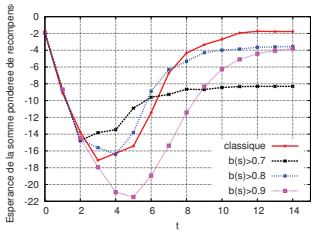
(c) Espérance de la somme pondérée d'entropies de l'état de croyance pour chaque instant t pour $\lambda = 0.9$.



(d) Espérance de la somme pondérée de récompenses pour chaque instant t pour $\lambda = 0.5$.



(e) Espérance de la somme pondérée de récompenses pour chaque instant t pour $\lambda = 0.7$.



(f) Espérance de la somme pondérée de récompenses pour chaque instant t pour $\lambda = 0.9$.

FIGURE 8 – Espérance de la somme pondérée d'entropies et de récompenses selon la valeur de λ.

7 Conclusion et perspectives

Nous pouvons conclure que l'utilisation d'un critère mixte, non-linéaire, est pertinente dans la réalisation de missions exploratoires. Dans ces missions, le but de l'agent est de réduire l'incertitude de son état de croyance. L'utilisation d'un critère non-linéaire que l'on optimise à partir des algorithmes issus de l'état-de-l'art de POMDP est possible, mais l'utilisateur (expert) de la méthode devra, soit utiliser des approximations linéaires du premier ordre basées sur un grand nombre d'état de croyance afin d'estimer au mieux le gradient de la fonction de valeur, soit mettre en place un algorithme de résolution dédié, pour lequel la fonction de valeur ne serait plus paramétrée par des α-vecteurs.

Nous avons aussi proposé une modélisation de la mission d'exploration de manière à ce que le critère optimisé soit le critère classique du POMDP. Notre approche résulte dans un modèle pour lequel on ajoute des buts fictifs au moyen d'actions dites *report s* pour chaque état s du système. Autrement dit l'agent décideur sera récompensé si et seulement si, il réalise l'action *report s* quand l'état s est l'état véritable du système. L'optimisation de ces actions (buts) ne dépend plus de manière directe d'une mesure de l'incertitude de l'état de croyance puisque la récompense associée peut être modélisée pour les paires état-action. La politique appliquant ces actions réduira de manière implicite l'incertitude de l'état de croyance. L'ajout d'actions (buts) peut paraître un facteur limitant de l'approche, mais le retour vers une modélisation classique nous permet d'utiliser des algorithmes efficaces basés sur la recherche heuristique, qui focalisent l'optimisation vers les actions les plus prometteuses. De plus, grâce à la comparaison des deux approches on peut proposer une structure des récompenses et des coûts du modèle POMDP classique telle que la politique obtenue corresponde à un taux de bonne classification comparable à celui obtenue à partir du critère mixte.

Ainsi, le critère mixte permet d'implémenter et d'optimiser une stratégie ρPOMDP qui respecte un taux de bonnes et mauvaises classifications défini comme le seuil de décision pour l'exécution d'une action *report s*. Toutefois, la détermination de la pondération λ reste arbitraire. D'autre part, le rapport entre récompenses et coûts dans une modélisation classique définit de manière implicite les taux de bonnes et de mauvaises classifications, qui ne peuvent être identifiés que par des simulations de la politique obtenue. D'une certaine manière l'approche par critère mixte peut être utilisée comme un outil constructif pour la détermination du *bon* rapport entre les récompenses et les coûts en regard des objectifs de taux de bonnes et mauvaises classifications : (i) choix du paramètre η par l'utilisateur ; puis (ii) réglage du paramètre λ juste suffisant par l'expert ; puis (3) ajustement des récompenses sur les états terminaux dans le modèle classique par l'expert. Cela nous permet de conclure que ces approches sont non seulement comparables e équivalentes en termes de réduction d'incertitude mais aussi complémentaires en termes le réglage des récompenses et des coûts.

Références

ARAYA LÓPEZ M. (2013). *Des algorithmes presque optimaux pour les problèmes de décision séquentielle à des fins de collecte d'information*. PhD thesis, Université de Lorraine.

ARAYA-LÓPEZ M., BUFFET O., THOMAS V. & CHARPILLET F. (2010). A POMDP Extension with Belief-dependent Rewards. *Advances in Neural Information Processing Systems*, **23**.

CANDIDO S. & HUTCHINSON S. (2011). Minimum uncertainty robot navigation using information-guided POMDP planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, p. 6102–6108.

CARVALHO CHANEL C., FARGES J., TEICHTEIL-KÖNIGSBUCH F. & G.INFANTES (2010). POMDP solving : what rewards do you really expect at execution ? In *Proc. of the 5th Starting AI Researchers' Symposium*.

CARVALHO CHANEL C., TEICHTEIL-KÖNIGSBUCH F. & LESIRE C. (2012). POMDP-based online target detection and recognition for autonomous UAVs. In *20th European Conference on Artificial Intelligence (ECAI). Including Prestigious Applications of Artificial Intelligence (PAIS) and System Demonstrations Track*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, p. 955–960 : IOS Press.

CASSANDRA A., KAELBLING L. & KURIEN J. (1996). Acting under uncertainty : Discrete Bayesian models for mobile-robot navigation. In *Proceedings of IEEE/RSJ*.

DEB K. (2001). *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons Hoboken, NJ.

DEINZER F., DENZLER J. & NIEMANN H. (2003). Viewpoint selection-planning optimal sequences of views for object recognition. *Lecture notes in computer science*, p. 65–73.

DUTTA ROY S., CHAUDHURY S. & BANERJEE S. (2004). Active recognition through next view planning : a survey. *Pattern Recognition*, **37**(3), 429–446.

EIDENBERGER R., GRUNDMANN T. & ZOELLNER R. (2009). Probabilistic action planning for active scene modeling in continuous high-dimensional domains. In *IEEE International Conference on Robotics and Automation (ICRA)*, p. 2412–2417.

EIDENBERGER R. & SCHARINGER J. (2010). Active perception and scene modeling by planning with probabilistic 6d object poses. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, p. 1036–1043.

KURNIAWATI H., HSU D. & LEE W. (2008). SARSOP : Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *Proc. RSS*.

MIHAYLOVA L., LEFEBVRE T., BRUYNINCKX H., GADEYNE K. & SCHUTTER J. D. (2002). Active sensing for robotics – a survey. In *5th Intl Conf. On Numerical Methods and Applications*, p. 316–324.

PINEAU J., GORDON G. & THRUN S. (2003). Point-based value iteration : An anytime algorithm for POMDPs. In *Proc. of International Joint Conference on Artificial Intelligence (IJCAI)*.

SABBADIN R., LANG J. & RAVOANJANAHARY N. (2007). Purely epistemic markov decision processes. In *Proceedings of the 22nd national conference on Artificial intelligence - Volume 2*, p. 1057–1062 : AAAI Press.

SAUX B. & SANFOURCHE M. (2011). Robust vehicle categorization from aerial images by 3d-template matching and multiple classifier system. In *7th International Symposium on Image and Signal Processing and Analysis (ISPA)*, p. 466–470.

SMALLWOOD R. & SONDIK E. (1973). The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, p. 1071–1088.

SMITH T. & SIMMONS R. (2004). Heuristic search value iteration for POMDPs. In *Proc. UAI*.

SMITH T. & SIMMONS R. (2005). Point-based POMDP algorithms : Improved analysis and implementation. In *Proc. UAI*.

SPAAN M. (2008). Cooperative Active Perception using POMDPs. *Association for the Advancement of Artificial Intelligence - AAAI*.

SPAAN M. & LIMA P. (2009). A decision-theoretic approach to dynamic sensor selection in camera networks. In *Int. Conf. on Automated Planning and Scheduling*, p. 279–304.

SPAAN M. & VLASSIS N. (2004). A point-based POMDP algorithm for robot planning. In *IEEE International Conference on Robotics and Automation (ICRA)*.

A Annexe

A.1 Preuve théorème 1

À partir de l'équation 4 on peut remplacer $V^\pi(b)$ et $H^\pi(b)$ par leurs expressions sous forme de somme pondérée. La convergence des différentes sommes est assurée par le facteur $\gamma \in [0,1[$ sachant que $r(b,\pi)$ et $H(b)$ sont bornées. On a donc :

$$J^\pi(b) = (1-\lambda)E_\pi\left[\sum_{t=0}^{\infty}\gamma^t r(b_t,\pi)\middle| b_0=b\right] + \lambda E_\pi\left[\sum_{t=0}^{\infty}\gamma^t H(b_t)\middle| b_0=b\right] \quad (19)$$

$$J^\pi(b) = (1-\lambda)E_\pi\left[r(b_0,\pi)|b_0=b\right] + (1-\lambda)E_\pi\left[\sum_{t=1}^{\infty}\gamma^t r(b_t,\pi)\middle| b_0=b\right] + \\ +\lambda E_\pi\left[H(b_0)|b=b_0\right] + \lambda E_\pi\left[\sum_{t=1}^{\infty}\gamma^t H(b_t)\middle| b_0=b\right] \quad (20)$$

$$J^\pi(b) = (1-\lambda)\sum_{s\in S} r(s,\pi)b(s) + (1-\lambda)E_\pi\left[\sum_{t=1}^{\infty}\gamma^t r(b_t,\pi)\middle| b_0=b\right] + \\ +\lambda\sum_{s\in S} b(s)\log(b(s)) + \lambda E_\pi\left[\sum_{t=1}^{\infty}\gamma^t H(b_t)\middle| b_0=b\right] \quad (21)$$

$$J^\pi(b) = (1-\lambda)\sum_{s\in S} r(s,\pi)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \\ +\gamma(1-\lambda)\sum_{o\in\Omega} p(o|b,\pi)E_\pi\left[\sum_{t=0}^{\infty}\gamma^t r(b_t,\pi)\middle| b_0=b_\pi^o\right] + \\ \gamma\lambda\sum_{o\in\Omega} p(o|b,\pi)E_\pi\left[\sum_{t=0}^{\infty}\gamma^t H(b_t)\middle| b_0=b_\pi^o\right] \quad (22)$$

$$J^\pi(b) = (1-\lambda)\sum_{s\in S} r(s,\pi)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \\ +\gamma\sum_{o\in\Omega} p(o|b,\pi)\left((1-\lambda)V^\pi(b_\pi^o) + \gamma\lambda H^\pi(b_\pi^o)\right) \quad (23)$$

$$J^\pi(b) = (1-\lambda)\sum_{s\in S} r(s,\pi)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \gamma\sum_{o\in\Omega} p(o|b,\pi)J^\pi(b_\pi^o) \quad (24)$$

Pour une politique stationnaire π donnée, la fonction de valeur J^π satisfait donc l'équation de Bellman 5. L'équation 5 montre que J^π correspond au critère γ-pondéré classique dans lequel la récompense est ajoutée à l'entropie de l'état de croyance courant. On est ramené à un problème de maximisation du critère pondéré précédent qui peut s'interpréter comme le calcul de récompenses *artificielles*. Celles-ci sont égales aux récompenses réelles $\sum_{s\in S} r(s,a)b(s)$ auxquelles s'ajoutent les entropies $H(b)$.

A.2 Preuve de la proposition 1

Nous avons :

$$J^\pi(b) = (1-\lambda)\sum_{s\in S} r(s,\pi)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \gamma\sum_{o\in\Omega} p(o|b,\pi)J^\pi(b_\pi^o), \quad (25)$$

et aussi, comme la convergence des différentes sommes est assurée par le facteur $\gamma \in [0,1[$, on a pour toute politique $\pi = (a,\pi')$:

$$J^*(b) = \max_\pi\left\{(1-\lambda)E_\pi\left[\sum_{t=0}^{\infty}\gamma^t r(b_t,\pi(b_t))\middle| b_0=b\right] + \lambda E_\pi\left[\sum_{t=0}^{\infty}\gamma^t H(b_t)\middle| b_0=b\right]\right\}$$

$$J^*(b) = \max_{(a,\pi')}\left[(1-\lambda)\sum_{s\in S} r(s,a)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \gamma\sum_{o\in\Omega} p(o|b,a)J^{\pi'}(b_a^o)\right]$$

$$J^*(b) = \max_{a\in A}\left[(1-\lambda)\sum_{s\in S} r(s,a)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \gamma\sum_{o\in\Omega} p(o|b,a)\max_{\pi'} J^{\pi'}(b_a^o)\right]$$

$$J^*(b) = \max_{a\in A}\left[(1-\lambda)\sum_{s\in S} r(s,a)b(s) + \lambda\sum_{s\in S} b(s)\log(b(s)) + \gamma\sum_{o\in\Omega} p(o|b,a)J^*(b_a^o)\right]$$