Problèmes décisionnels de Markov temporels : formalisation et résolution

Emmanuel Rachelson
ONERA-DCSD, Toulouse
ISAE, Toulouse
emmanuel.rachelson@onera.fr

Sous la direction de : Frédérick Garcia, Patrick Fabiani

Résumé

On s'intéresse aux problèmes de décision dans l'incertain fortement conditionnés par le temps. Cet article de synthèse présente les spécificités de ces problèmes, l'analyse de leurs caractéristiques pour leur formalisation ainsi que plusieurs propositions algorithmiques fondées sur la structure des problèmes temporels dans l'incertain.

1 Introduction

On s'intéresse aux problèmes de décision dans un environnement incertain décrit par un formalisme de processus Markoviens et fortement conditionné par la variable temporelle. De nombreux exemples de nature très différente illustrent cette situation d'incertitude et de dépendance au temps : la coordination décentralisée de deux agents autonomes (drones hélicoptères par exemple), la décision en environnement instationnaire (la gestion des flux d'avions dans un aéroport avec un trafic dépendant de l'heure), la planification de tâches parallèles et/ou concurrentes, etc.

Afin de représenter la dynamique de tels systèmes, le formalisme de Processus Décisionnels de Markov (MDP, [Puterman, 1994]) est couramment utilisé. Or ce formalisme considère à la base des processus stationnaires à pas de temps unitaire. Un problème décisionnel de Markov temporel est un problème MDP dépendant du temps. Notre travail s'articule autour de l'analyse des différentes représentations MDP de la dépendance au temps et nous cherchons des méthodes exactes et approchées permettant de construire des politiques (stratégies en boucle fermée) en utilisant ces représentations pour formaliser les problèmes temporels stochastiques.

Lorsqu'on cherche à représenter la dépendance au temps dans le cadre MDP, on se heurte à plusieurs questions de fond : le temps est-il une variable d'état ? Comment représenter les différentes actions $attendre$? Qu'est-ce qui fait de la variable temporelle une variable à part ? Les problèmes instationnaires présentent souvent une structure particulière, comment capturer cette structure dans la représentation sur laquelle on va planifier ?

En section 2 on présente le problème plus en détail ; en se basant sur des exemples, on illustre les particularités de modélisation de la variable temporelle et les différences avec des variables d'état "classiques". Ces particularités débouchent, en section 3 sur une représentation MDP à temps continu et actions paramétriques. On montre alors que les algorithmes d'optimisation de MDP standards s'adaptent bien à ce nouveau cadre et on en donne un exemple avec le planificateur TMDPpoly. En section 4 on revient sur la nature des phénomènes instationnaires et on s'intéresse à la représentation des la concurrence et à la spécification de systèmes stochastiques concurrents. Sur cette base, on présente à la section 5 deux méthodes de recherche de politiques approchées pour ces processus décisionnels stochastiques et concurrents.

2 Représentation de la dépendance au temps dans le cadre stochastique Markovien

2.1 Origine du problème

Le problème d'instationnarité d'un processus décisionnel stochastique peut-être illustré de plusieurs façons différentes qui mettent en avant l'origine de l'instationnarité et la structure du problème.

Considérons en premier exemple le problème de navigation d'un agent devant se rendre de l'INRA à l'ONERA en optimisant son temps de trajet. En fonction de l'heure de la journée le trafic est plus ou moins dense et les sections de trajet soumises à des aleas quant au résultat (routes fermées) et à la durée de parcours (embouteillages). Un exemple similaire a été présenté par [Boyan et Littman, 2001]. Le problème de navigation d'un drône hélicoptère entre deux sites d'un feu de forêt peut également se modéliser comme un tel problème instationnaire avec des chemins dont la traversabilité est fonction du temps comme sur la figure 1. Dans cet exemple, la dépendance au temps est explicite, le temps est continu et l'horizon d'instationnarité est borné. On verra en section 3 comment on peut optimiser une stratégie sur une représentation adaptée à ces problèmes.

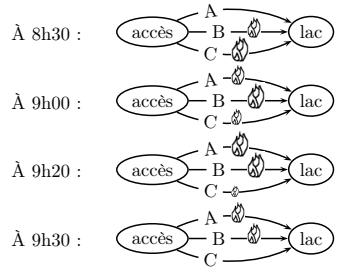

FIGURE 1 – Traversée d'un feu de forêt

En revanche, si on considère comme second exemple le

problème de la coordination décentralisée de deux agents autonomes hétérogènes il s'avère qu'une représentation directe de l'instationnarité présente deux problèmes majeurs. Dans le processus de communication et négociation présenté dans [Rachelson, 2007] les agents échangent des messages permettant à chacun d'intégrer les effets des actions de l'autre comme des éléments d'instationnarité de l'univers. Or cette intégration implique les deux désavantages suivants :

1. A chaque fois qu'un message est reçu, elle nécessite un lourd mécanisme d'adaptation et d'ingénierie sur les fonctions de transition et de récompense dans la représentation MDP propre à chaque agent.
2. Elle perd la représentation structurée en processus concurrents couplés. Par exemple, le déroulement parallèle de la propagation de l'incendie, de l'exécution du plan de l'autre agent et des changements possibles d'objectifs sont chacun des processus concurrents simples qui sont couplés via l'état global du système.

Afin de mieux décrire cet aspect structuré de l'instationnarité, on introduit le problème de gestion d'une ligne de métro. Dans ce problème jouet, on considère une ligne de métro composée de n quais. Des passagers arrivent individuellement sur chaque quai avec une période aléatoire dépendant de l'heure de la journée et de la position de la station en ville. Une fois mis sur les rails, les métros font leur tournée de façon automatique. Le problème de décision est très simple, il s'agit de déterminer une politique de mise en service, de retrait et de réparation des rames qui maximise le profit de l'exploitant de la ligne sur le long terme. Ce problème présente une forte composante incontrôlable et un grand nombre de processus parallèles agissant sur le même état global du système. Un problème similaire est l'affectation de taxiways aux avions en partance et à l'arrivée dans un aéroport.

Ce problème présente l'instationnarité sous un angle beaucoup plus structuré. Elle ne provient plus uniquement de la synthèse dans un cadre MDP continu des phénomènes dépendants du temps : la principale cause de l'instationnarité et de la dépendance au temps provient de la concurrence des processus aléatoires que l'on considère. Comme illustré par [Cushing *et al.*, 2007] dans le cas déterministe, c'est de la concurrence que provient la difficulté de la planification temporelle. On s'attachera à modéliser cet aspect en section 4 et à en tirer parti en section 5. Il est toutefois important de noter dès à présent que la représentation structurée de la concurrence des processus permet de décrire de façon factorisée des espaces d'états (et les dynamiques associées) très grands qui nécessiteraient un immense travail d'ingénierie pour parvenir à une représentation entièrement intégrée en un unique processus instationnaire.

2.2 Temps discret vs. temps continu ?

Une des premières questions se posant pour la problématique de planifier en fonction du temps est de connaître la granularité de la variable t. Pour les problèmes mentionnés précédemment, il apparait naturellement que la valeur exacte de la date à laquelle on décide peut-être critique. Par exemple pour la traversée du feu de forêt ou pour l'affectation de taxiways, les dates de décision sont soumises à aléas et la meilleure date de décision peut-être située entre deux points de discrétisation. A l'inverse, sur certaines périodes la stratégie peut-être proche du stationnaire et l'on augmente alors inutilement la taille de l'espace d'état en imposant une granularité donnée à la variable temporelle.

Le problème de la coordination temporelle de processus ou d'agents dans l'incertain a été abordé dans les travaux de [Mausam *et al.*, 2005] pour le traitement de MDP concurrents à temps discret, ou dans les travaux de [Bernstein *et al.*, 2000] pour la planification multi-agent dans l'incertain. Ces approches sont centrées sur la coordination d'actions d'agents synchrones sur un temps discrétisé de façon régulière et unitaire. Nous abordons le problème de la coordination par l'aspect temporel et cherchons à nous affranchir de cette notion de discrétisation. On considérera donc par la suite que nos problèmes de décision sont à temps continu.

2.3 Une variable d'état continue comme un autre ?

Lorsqu'on spécifie un problème de planification temporelle dans l'incertain on n'a d'information sur l'instationnarité que sur un certain horizon temporel. Au delà de cet horizon, on considère le problème stationnaire ou cyclique. Il semble qu'on pourrait alors considérer le temps comme une variable d'état continue comme une autre. Cependant, deux différences importantes font du temps une variable d'état à part : d'une part le principe de causalité et d'autre part le problème de l'horizon de planification dans le cadre MDP.

Un problème décrit comme dépendant explicitement du temps se caractérise par le fait qu'on ne peut y revenir en arrière, c'est le principe de causalité. Traiter la variable temporelle comme une variable d'état continue quelconque revient donc à négliger cet aspect. Introduire une variable temporelle explicite conditionne donc la dynamique du système : un problème à temps explicite (dans l'état) est donc un problème Markovien sans cycle.

Par ailleurs, il est commun de raisonner à horizon infini dans le cadre MDP. Cet horizon correspond au nombre de coups que l'on s'autorise à jouer avant la fin de l'exécution de la stratégie. Cet horizon correspond donc à un nombre d'actions et non à une borne temporelle. On peut travailler à horizon temporel borné et à horizon de planification (en termes de nombre d'actions) infini. Afin d'étendre le cadre MDP existant (dont on rappelle la description au début de la section suivante), on s'intéresse aux problèmes de MDP temporels à horizon temporel et de planification infinis. Or les variables continues classiques d'un MDP sont nécessairement bornées. Considérer un horizon temporel inifini implique donc de considérer une variable d'état temporelle non-bornée, ce qui sort du cadre MDP classique. La section 3 montre comment on peut étendre le cadre MDP et ses méthodes aux MDP temporels à horizon infini.

3 Variable temporelle continue et actions paramétriques

3.1 Généralisation des MDP au temps continu et actions paramétriques

Un MDP ([Puterman, 1994]) est décrit de façon classique par un quintuplet $< S, A, p, r, T >$ où S représente l'espace

d'état (souvent discret, dénombrable, parfois continu, borné), A représente l'espace des actions, $p(s'|s,a)$ décrit la probabilité d'effectuer une transition vers l'état s' sachant qu'on entreprend l'action a dans l'état s. $r(s,a)$ est un modèle de récompense associé aux transitions permettant par la suite d'exprimer le critère d'optimisation de la politique que l'on cherche. Enfin T est l'ensemble des pas de décision que l'on s'autorise à faire ; en horizon infini, T est isomorphe à $\mathbb{N}$. On se dote alors d'un critère et on cherche une politique π (fonction de S dans A) qui maximise ce critère. Les critères qui nous intéressent ici sont le critère γ-pondéré qui s'exprime : $V_\gamma^\pi = E\left(\sum_{\delta\in T}\gamma^\delta r(s_\delta,\pi(s_\delta))\right)$ et le critère total correspondant à $\gamma = 1$ mais qui n'a pas nécessairement de valeur finie. Pour un critère γ-pondéré, la fonction de valeur de la politique optimale vérifie :

$$V^\pi(s) = max\, a \in A r(s,a) + \gamma \sum_{s'\in S} p(s'|s,a)V^\pi(s') \quad (1)$$

On souhaite introduire un cadre de formalisation souple permettant de spécifier des actions à la fois continues et discrètes. Par ailleurs, afin d'éviter les problèmes d'énumération sur les espaces continus, on opte pour une représentation de l'état sous forme factorisée (sous formes de variables). On considère ensuite qu'on a un espace discret dénombrable d'actions mais que ces dernières sont paramétrées et que les paramètres peuvent être indifféremment continus ou discrets. Ces éléments permettent de définir un MDP factorisé à temps continu et à actions paramétriques (XMDP) comme défini dans [Rachelson *et al.*, 2007]. Un XMDP est un sextuplet $< S, A(X), p, r, T >$ où :

- S est un espace d'états Borélien décrivant des variables d'état continues ou discrètes.
- A est un espace d'actions décrivant un jeu fini d'actions $a_i(x)$ avec x un vecteur de paramètres prenant ses valeurs dans un espace X. De ce fait, l'espace d'actions $A(X)$ du problème est un espace hybride.
- p est une densité de probabilités donnant $p(s'|s,a(x))$.
- r est une fonction de récompense donnant $r(s,a(x))$.
- T est un ensemble de périodes de décision.

On se dote d'un critère γ-pondéré temporel exprimant - comme dans le cas classique - la dépréciation des récompenses obtenues dans l'avenir et permettant d'assurer le caractère fini du critère :

$$V_\gamma^\pi(s,t) = E_{(s,t)}^\pi\left\{\sum_{\delta=0}^{\infty}\gamma^{t_\delta - t} r_\pi(s_\delta,t_\delta)\right\} \quad (2)$$

On a alors montré ([Rachelson *et al.*, 2008]) que la politique optimale était donnée par une équation de Bellman étendue :

$$V^*(s,t) = \max_{a\in A}\sup_{x\in X}\left\{r(s,t,a(x)) + \int_{\substack{t'\in\mathbb{R}\\ s'\in S}} \gamma^{t'-t}p(s',t'|s,t,a(x))V^*(s',t')ds'dt'\right\} \quad (3)$$

L'existence de cette équation prouve que l'on peut adapter facilement les algorithmes classiques d'optimisation de politiques pour MDP, notamment les algorithmes d'itération de la valeur ou de la politique.

La difficulté inhérente aux problèmes à temps explicite et espaces d'états et d'actions hybrides réside dans la résolution des intégrales et de l'opérateur sup dans l'équation 3. Dans le cadre des problèmes où la seule variable temporelle est continue et sous hypothèses fortes, le cadre TMDP permet un traitement exact et direct de cette équation.

3.2 XMDP et horizon fini : application au cadre TMDP

Les Time-dependent MDP (TMDP) constituent un formalisme dédié au traitement de MDP dépendants du temps. Nous avons prouvé qu'un TMDP est en fait un XMDP doté d'un critère total et présentant le conditions nécessaires pour que ce critère existe. Un TMDP est décrit dans [Boyan et Littman, 2001] par :

- un espace d'états discret S,
- un espace d'actions discret A,
- un ensemble de réalisations M, chaque réalisation μ étant constituée de :
  - un état d'arrivée x'_μ,
  - une distribution sur la durée de la réalisation ou sur sa date de fin P_μ,
  - un booléen indiquant si P_μ porte sur une durée ou une date de fin.
- une fonction de transition $L(\mu|s,a,t)$,
- une fonction de récompense $r(\mu,t,\tau)$,
- une fonction de coût instantané d'attente $K(s,t)$.

Nous avons montré qu'un TMDP était un XMDP à une unique action paramétrique (l'action $attendre$), et qu'ainsi, les équation d'optimalité fournies dans [Boyan et Littman, 2001] correspondaient bien à l'optimisation d'un critère total. Ces équations sont :

$$V(s,t) = \sup_{t'\geq t}\left(\int_t^{t'} K(s,\theta)d\theta + \overline{V}(s,t')\right) \quad (4)$$

$$\overline{V}(s,t) = \max_{a\in A} Q(s,t,a) \quad (5)$$

$$Q(s,t,a) = \sum_{\mu\in M} L(\mu|s,t,a)\cdot U(\mu,t) \quad (6)$$

$$U(\mu,t) = \begin{cases}\int_{-\infty}^{\infty} P_\mu(t')[R(\mu,t,t') + V(s'_\mu,t')]dt' \\ \int_{-\infty}^{\infty} P_\mu(t'-t)[R(\mu,t,t') + V(s'_\mu,t')]dt'\end{cases} \quad (7)$$

Si on suppose que les P_μ sont des distributions discrètes, que L est contante par morceaux en t et que r est décomposable en trois fonctions additives linéaires par morceaux en t, τ et t', alors on peut effectuer une résolution exacte des équations 4 à 7 forunissant une fonction de valeur linéaire par morceaux en t. On peut ainsi mettre en place un algorithme d'itération de la valeur pour la résolution exacte des TMDP.

3.3 Extension de la résolution des TMDP

Cependant, les hypothèses mentionnées plus haut sont très restrictives et nous avons cherché à les généraliser dans un

premier temps, puis à les relaxer pour chercher des solutions approchées.

Nous avons montré que dans le cadre d'une description sous forme de fonctions polynômiales des fonctions de transition et de récompense, il fallait nécessairement que les P_μ soient des distributions discrètes et les L des fonctions constantes par morceaux pour qu'une résolution exacte soit possible. Il faut par ailleurs que le modèle de récompense soit décrit par des fonctions polynômiales par morceaux de degré inférieur à 4. Cette preuve repose sur les propriétés de la fonctionnelle de Dirac et sur la théorie de résolution des équations polynômiales.

Par ailleurs, afin de généraliser la résolution des TMDP à des problèmes plus généraux, nous proposons une extension permettant de représenter P_μ, L et r sous forme de fonctions polynômiales par morceaux de degré quelconque. Cette extension, nommée TMDPpoly est basée sur le fait que, si l'on note a le degré de P_μ, b celui de r, c celui de L et que l'on suppose qu'à l'étape n le degré de V_n est inférieur à b, alors on peut calculer de façon approchée V_{n+1} sous la forme d'un polynôme de degré $a+b+c+1$. On effectue alors une étape d'interpolation par un polynôme défini par morceaux de degré b avant de réitérer à l'équation 7.

Malgré ce gain d'expressivité du modèle TMDP, la méthode de résolution basée sur l'itération de la valeur souffre des deux problèmes récurrents des formalismes MDP :

— D'une part, l'écriture complète du modèle - en particulier dans le cas d'états discrets énumérés - nécessite un fort travail préparatoire d'ingénierie qui grève l'utilisation de tels modèles.
— D'autre part le "curse of dimensionality" de Bellman [Bellman, 1957] rend le traitement d'espaces d'états de grande dimension très difficile.

Or, comme on l'a vu en section 2, les problèmes décisionnels de Markov temporels peuvent être structurés par l'analyse de la concurrence des événements et des processus. Plutôt que de synthétiser dans une fonction de transition globale les effets de processus concurrents simples, on se propose de les décrire séparément et d'analyser leur exécution couplée.

4 Instationnarité et concurrence

La démarche que nous avons adoptée repose sur l'idée de représenter formellement l'exécution de processus temporels à événements discrets concurrents puis d'introduire une couche décisionnelle permettant d'influer sur le système. On cherche alors à construire des algorithmes de recherche de politiques (éventuellement incomplètes) sur cette représentation formelle en s'appuyant sur des données de simulation.

4.1 Processus Markoviens concurrents

Les processus semi-Markoviens (SMP) constituent une classe particulière de processus de Markov. Ils décrivent une chaîne de Markov dans laquelle le temps de séjour dans un état n'est pas unitaire (chaîne de Markov classique) ou exponentiel (chaîne de Markov à temps continu) mais suit une distribution quelconque respectant la propriété de Markov. De la même manière qu'en ajoutant un espace d'actions sur une chaîne de Markov, on crée un MDP, on peut définir un processus décisionnel semi-Markovien (SMDP). L'unique différence avec les MDP réside dans l'existence d'une fonction $F(\tau|s,a)$ donnant la distribution sur la durée de séjour dans un état, en fonction de l'état, de l'action entreprise et indépendemment de l'état d'arrivée. On peut montrer que l'optimisation d'une politique sur un SMDP se ramène à celle d'un MDP. Afin de décrire la concurrence de plusieurs processus semi-Markoviens, on peut utiliser le formalisme des SMP généralisés (GSMP, [Glynn, 1989]) qui décrivent des SMP concurrents couplés. Enfin, en ajoutant, parmi les différents événements concurrents d'un GSMP, des actions que l'on peut choisir, on définit un processus décisionel semi-Markovien généralisé (GSMDP, [Younes et Simmons, 2004]). Un GSMP peut être décrit par :

— un espace d'états E,
— un ensemble d'événements E,
— un ensemble de compteurs associés aux événements c_e représentant le temps avant déclenchement.
— une fonction de durée $F(\tau|s,e)$
— une fonction de transition $P(s'|s,\tau,e)$

La dynamique d'un GSMP est décrite par ses compteurs : dans un état s, il existe un sous-ensemble d'événements E_s dits "actifs" (pour les autres, $c_e=\infty$), on prend l'événement pour lequel c_e est le plus petit, on décrémente tous les autres compteurs de la quantité c_e et on tire l'état suivant selon la distribution $P(s'|s,c_e,e)$. Dans l'état suivant s', on désactive les événements n'appartenant pas à $E_{s'}$, on tire une valeur de compteur pour les événements qui s'activent (dont éventuellement l'événement qui vient de se délencher) et on continue le processus. Ecrire un GSMDP consiste simplement à créer un événement contrôlable a concurrent des autres événements et prenant ses valeurs dans un espace A.

L'exemple du métro ou de l'affectation de taxiways sont illustratifs de l'expressivité des GSM(D)P pour représenter des processus complexes comme composés de processus simples concurrents. On peut également noter que dans le problème de la coordination d'agents, le fait de représenter l'autre agent comme un processus concurrent couplé annule simplifie la représentation du problème de décision.

Afin de replacer le formalisme GSM(D)P dans le cadre plus général de la simulation de systèmes à événements discrets, nous nous sommes intéressés au formalisme de Spécification des Systèmes à Evénements Discrets DEVS.

4.2 Simulation de processus à événements discrets

Le formalisme DEVS ([Zeigler *et al.*, 2000]) est un langage de spécification de systèmes à événements discrets. Indépendant du formalisme utilisé, il couvre les systèmes déterministes ou stochastiques, les réseaux de Pétri, les automates cellulaires, etc. DEVS est une représentation évoluée d'un automate temporel, séparant les processus et les hiérarchisant. L'élément de base de la représentation DEVS est le modèle atomique. Ce dernier est connecté via des ports à d'autres modèles, atomiques ou composites. Un modèle composite est un ensemble de ports connectés aux entrées et sorties des sous-modèles atomiques. L'état d'un modèle est interne et non-partagé par défaut. Un modèle atomique est enfin constitué de l'octuplet :

— X l'ensemble des événements d'entrée du modèle
— Y l'ensemble des événements de sortie du modèle
— S l'espace d'états internes du modèle
— δ_{ext} la fonction de transition externe dépendant des événements d'entrée et de l'état du modèle
— δ_{int} la fonction de transition interne ne dépendant que de l'état du modèle
— δ_{con} la fonction de résolution de conflits entre événements internes et externes
— λ la fonction de sortie
— ta la fonction d'avancement du temps lorsqu'un événement se produit

Cependant, l'intégration de GSMP dans le formalisme DEVS pose un problème majeur : un GSMP est constitué de processus indépendants couplés via l'état commun du système. En revanche, un ensemble de modèles DEVS correspondent à un ensemble d'agents ayant chacun son état propre et ne le partageant qu'au travers des ports d'entrée et de sortie. Nous avons donc exploré deux voies d'intégration de GSMP dans DEVS (et dans les plates-formes logicielles basées sur DEVS) : la première implique l'écriture d'un GSMP comme un unique modèle DEVS, interfaçable avec d'autres modèles DEVS et gérant les événements GSMP concurrents en interne. L'autre implique un modèle très communicant où chaque événement GSMP est un modèle DEVS et où les valeurs des variables d'état concernant les préconditions et les états partagés transitent entre modèles à chaque transition d'un modèle. Sur cette seconde représentation, nous avons proposé une technique de réduction du volume des communications, cependant ce dernier demeure très important.

Il existe plusieurs plate-formes de simulation conçues sur les bases théoriques et opérationnelles du formalisme DEVS. Afin de pouvoir simuler des GSMP de façon vérifiable, nous avons développé une extension GSMP à la plate-forme VLE ([Quesnel, 2006]). Un GSMDP contrôlé par une politique est un GSMP ; en le simulant, on obtient une évaluation de cette politique. C'est l'usage que l'on fait de GSMP-VLE et l'approche sur laquelle on se base par la suite.

4.3 Processus décisionnels semi-Markoviens généralisés

Introduire des événements contrôlables non-concurrents dans un GSMP est l'approche suivie par [Younes et Simmons, 2004] dans la définition des GSMDP. Or la dynamique d'un GSMDP basée uniquement sur son espace d'état n'est pas Markovienne. [Nilsen, 1998] montre que si l'on met les compteurs dans l'état alors la dynamique d'un GSMP devient celle d'un SMP et est donc Markovienne. De même, si on stocke les compteurs dans l'état d'un GSMDP, on peut chercher des politiques optimales sous forme de politiques Markoviennes. Cependant, les compteurs de certains événements sont intrinsèquement inobservables, il n'est donc pas pertinent de construire un processus décisionnel en dépendant.

La principale approche de résolution des GSMDP proposée dans [Younes et Simmons, 2004] repose sur le fait d'approximer toutes les fonctions de transition par des distributions de type "phase-type". On peut alors, par réduction, ramener le problème à un MDP à temps continu et, par uniformisation ([Puterman, 1994]), à un MDP classique avec un espace d'état augmenté. Cette approche qui ne fait qu'une approximation puis une résolution exacte est cependant handicapée par la difficulté de prouver la pertinence de l'approximation et par le problèmes des espaces d'états de grande dimension.

Notre approche repose sur l'hypothèse qu'en rendant la variable temporelle observable dans le processus décisionnel, on peut construire une politique Markovienne dont la valeur est ϵ-optimale. On se propose donc d'ajouter une variable temporelle explicite dans le cadre GSMDP et de spécifier des politiques sur cet espace d'état augmenté. Afin de contrer le problème de l'exploration des espaces d'état de grande dimension, on se propose de s'appuyer sur la simulation de GSMP afin d'obtenir l'évaluation de nos politiques sur un sous-espace d'état pertinent et restreint.

5 Politiques approchées pour GSMDP instationnaires

5.1 Itération de la politique temporelle approchée

L'approche algorithmique que nous adoptons se situe dans la famille des algorithmes d'itération de la politique (PI). L'algorithme PI pour MDP alterne deux étapes. A l'étape n on suppose qu'on a une politique π_n, on évalue cette politique selon un critère donné, en utilisant l'équation 1 sans l'étape de maximisation. On obtient ainsi la valeur V^{π} de cette politique. On effectue alors une amélioration sur un coup dans chaque état en appliquant l'équation 1. Cet algorithme converge généralement en moins d'itérations que l'algorithme d'itération de la valeur, cependant le temps de calcul associé est souvent plus long à cause de l'évaluation systématique de la politique à chaque itération. Les algorithmes de PI approchée (API) reposent sur le fait d'approcher la valeur de π sans la calculer exactement pour effectuer les itérations. [Munos, 2003] montre dans quels cas existent des garanties quant à la convergence et à la précision des approches d'API.

Dans le cadre des problèmes Markoviens temporels à horizon infini, on cherche des politiques définies comme des fonctions des variables s et t. La variable t fait partie de l'état mais pour les raisons exposées en section 2 on la met en avant dans la recherche de la politique. Dans le cas le plus simple on a alors un espace d'états et d'action discrets à l'exception du temps, on cherche donc - par une approche d'itération de la politique - à améliorer à la fois la partition de la variable temporelle et la stratégie sur l'espace d'état défini par les intervalles sur le temps et les autres états. De façon plus générale, on appelle ATPI (Approximate Temporal Policy Iteration) la famille des algorithmes d'itération de la politique approchée appliqués au cadre des problèmes de Markov temporels.

5.2 Deux algorithmes pour ATPI

Pour la première étape d'un algorithme ATPI, on doit définir une manière d'évaluer la politique courante sur chaque état. On propose deux approches distinctes pour cette étape.

La première approche ([Rachelson, 2007]) se base sur une représentation de type TMDP du processus décrit. L'évaluation de la politique est alors aisée par les techniques décrites plus haut. L'étape d'optimisation ne concernant qu'un coup à l'avance on n'a plus besoin de mécanisme lourd de réduction du degré des polynômes obtenus.

La seconde approche proposée pour ATPI est inspirée des algorithmes de recherche heuristique comme (l)RTDP ([Barto *et al.*, 1995]). L'idée est de définir une heuristique sur la fonction de valeur de la politique puis d'utiliser la simulation du système (à partir du modèle GSMP par exemple) afin de mettre à jour la valeur de la politique sur les états traversés par plusieurs simulations successives.

5.3 Apprentissage de politiques continues

Plutôt que d'explorer la totalité de l'espace d'état, on peut se concentrer sur l'estimation des Q-valeurs des couples (s, a). Plusieurs méthodes d'apprentissage de politiques sous forme d'arbres sur espaces d'état et d'actions hybrides sont présentées dans [Ernst *et al.*, 2005]. Un de nos axes de recherche actuels concerne l'application de ces méthodes aux problèmes Markoviens temporels. Cette approche utilise uniquement le simulateur GSMP.

5.4 online-ATPI

L'algorithme online-ATPI part sur l'idée du dernier algorithme et la généralise. L'algorithme est présenté en figure 2. Il repose sur l'idée d'instanciation en ligne de la politique optimale et sur le couplage d'un modèle et d'un simulateur pour la recherche de politique.

```
init(V_0, V_1)

/* Boucle principale */
tant que test == 0 faire
    test = 1
    σ = opt_simu(r, P, V_0, s_0, γ)
    pour tous les s ∈ σ.states faire
        test = test × V_1.learn(s, σ.value(s), σ.pp(s), ε_1, ε_2)
    fin
    V_0 ← V_1
fin

/* Simulation et optimisation en ligne */
opt_simu(r, P, V, s_0, γ){{
s ← s_0
tant que simulation pas finie faire
    pour tous les a ∈ A faire
    Q̃(s,a) = r(s,a) + eval(∫_{s'} γ^{t'-t} P(s'|s,a) V(s') ds')
    a = argmax_{a∈A} Q̃(s,a)
    s' = simu(a)
    σ.append({s, a, s', r(s', a, s), P(s'|s, a)})
    s ← s'
fin
σ.wrap()
retourner σ }

/* Evaluation de la politique */
value.learn(s, V, pp, ε_1, ε_2){
si |value.eval(s) − V| < ε_1 alors
    retourner 1
fin
sinon
    si pp < ε_2 alors
        retourner 1
    fin
    sinon
        value.update(V, s, pp)
        retourner 0
    fin
fin
}
```

FIGURE 2 – online-ATPI

Références

[Barto *et al.*, 1995] Andrew G. Barto, Steven J. Bradtke, et Satinder P. Singh. Learning to act using real-time dynamic programming. *Artificial Intelligence*, 72(1-2) :81–138, 1995.

[Bellman, 1957] Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1957.

[Bernstein *et al.*, 2000] Daniel S. Bernstein, Shlomo Zilberstein, et Neil Immerman. The complexity of decentralized control of markov decision processes. Dans *16th Conference on Uncertainty in Artificial Intelligence*, 2000.

[Boyan et Littman, 2001] J. A. Boyan et M. L. Littman. Exact solutions to time dependent MDPs. *Advances in Neural Information Processing Systems*, 13 :1026–1032, 2001.

[Cushing *et al.*, 2007] William Cushing, Subbarao Kambhampati, Mausam, et Daniel S. Weld. When is temporal planning really temporal ? Dans *IJCAI*, 2007.

[Ernst *et al.*, 2005] Damien Ernst, Pierre Geurts, et Louis Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 2005.

[Glynn, 1989] Peter Glynn. A GSMP formalism for discrete event systems. Dans *IEEE*, 1989.

[Mausam *et al.*, 2005] Mausam, Emmanuel Benazera, Ronen Brafman, Nicolas Meuleau, et Eric A. Hansen. Planning with continuous resources in stochastic domains. Dans *International Joint Conference on Artificial Intelligence*, 2005.

[Munos, 2003] Rémi Munos. Error bounds for approximate policy iteration. Dans *Int. Conf. on Machine Learning*, 2003.

[Nilsen, 1998] F. B. Nilsen. GMSim : a tool for compositional GSMP modeling. Dans *Winter Simulation Conference*, 1998.

[Puterman, 1994] Martin L. Puterman. *Markov Decision Processes*. John Wiley & Sons, Inc, 1994.

[Quesnel, 2006] Gauthier Quesnel. *Approche formelle et opérationnelle de la multi-modélisation et de la simulation des systèmes complexes*. PhD thesis, Univ. du Littoral, Calais, 2006.

[Rachelson *et al.*, 2007] E. Rachelson, F. Teichteil, et F. Garcia. XMDP : un modèle de planification temporelle dans l'incertain à actions paramétriques. Dans *Journées Françaises Planification Décision Apprentissage*, 2007.

[Rachelson *et al.*, 2008] Emmanuel Rachelson, Frederick Garcia, et Patrick Fabiani. Extending the bellman equation for MDP to continuous actions and continuous time in the discounted case. Dans *Int. Symp. on AI and Math.*, 2008.

[Rachelson, 2007] Emmanuel Rachelson. Preliminary results for approximate temporal coordination under uncertainty. Dans *ICAPS Doctoral Consortium*, 2007.

[Younes et Simmons, 2004] H .L. S. Younes et R. G. Simmons. Solving generalized semi-markov decision processes using continuous phase-type distributions. Dans *AAAI*, 2004.

[Zeigler *et al.*, 2000] Bernard P. Zeigler, Tag Gon Kim, et Herbert Praehofer. *Theory of Modeling and Simulation*. Academic Press ; 2nd edition, 2000.