Modèle de conception de SMA coopératifs par planification réactive

Iadine CHADÈS* François CHARPILLET*
chades@loria.fr charp@loria.fr

*Equipe MAIA
INRIA Lorraine
B.P. 239
54506 Vandœuvre-Lès-Nancy
FRANCE

Résumé :
Nous proposons une nouvelle approche de conception d'agents coopératifs à l'aide de processus décisionnels de Markov (MDPs). Cette méthode de conception tire profit de deux propriétés fondamentales de nos agents : la subjectivité et l'empathie. Tandis que la subjectivité s'intéresse aux problèmes de conception de plan réactif dans des conditions de perceptions incomplètes d'un environnement, l'empathie d'un agent permet de prévoir les incertitudes de comportements de ses compères et de coordonner ses actions. Notre travail sur l'étude théorique de convergence des plans réactifs ainsi calculés a montré que les algorithmes convergeaient dans des conditions favorables vers un équilibre de Nash.

Mots-clés : Systèmes Multi-agents, Processus Décisionnels de Markov, Planification, Apprentissage par renforcement

Abstract:
In this paper, we propose a new method for designing cooperative MAS using Markov decision processes (MDPs). Our approach is based on two major agents' properties : the subjectivity and the empathy. Whereas the subjectivity helps to compute the reactive policy of each agent in a partially observable environnement, the empathy allows an agent to predict the behaviors of the others agents, and thus offers a mean to coordinate their actions. The set of memoryless policies thus computed reveals some interesting properties as it leads to Nash equilibria in some favorable conditions.

Keywords: Multiagent systems, Markov Decision Processes, Planning, Reinforcement Learning

1 Introduction

Dans la littérature, les démarches suivies dans la conception des systèmes multi-agents sont souvent empiriques. Nous distinguons deux méthodes de travail. Le procédé qui consiste à concevoir des agents avec certaines propriétés puis d'étudier les problèmes qu'ils sont capables ou non de résoudre : nous parlerons de méthode ascendante. Dans ce cas, les systèmes multi-agents sont souvent réactifs [Dorigo et Di Caro, 1999]. Nous y opposons la méthode descendante, il s'agit à partir d'un problème donné de concevoir le système multi-agents capable de résoudre la tâche pour laquelle il a été créé. Cette méthode est principalement appliquée dans les cas de systèmes cognitifs [Wooldridge, 2002].

L'objet de notre étude est l'élaboration d'un formalisme concis permettant la conception d'un système multi-agents coopératifs. Notre système devra faire face à une forte incertitude, et par conséquent il sera confronté aux comportements probabilistes des agents. Nous avons choisi de prévoir cette incertitude en **coordonnant** nos agents à travers l'utilisation d'un processus de planification. Les modèles de Markov permettent de gérer la stochasticité des décisions. Certains sont dédiés à une utilisation mono-agent avec un environnement accessible (MDP), ou partiellement observable (POMDP). D'autres sont dédiés à la modélisation de plusieurs agents dans un environnement observable ou non (MMDP, DEC-MDP, DEC-POMDP).

Bien que le DEC-POMDP semble être le modèle le plus respectueux des caractéristiques de notre problème, sa résolution est NEXP-complet pour un nombre d'agents supérieur ou égal à 2 [Bernstein *et al.*, 2000]. Son utilisation ne convient donc pas pour une approche réaliste de la **coordination** des agents dans notre système multi-agents. Nous proposons une heuristique de résolution d'un DEC-POMDP pour concevoir chaque comportement d'agent. Elle repose sur la résolution d'un processus décisionnel de Markov adapté que nous avons appelé MDP subjectif. Ainsi, notre solution approchée à ce problème est fondée sur deux propriétés fondamentales de nos agents : **la subjectivité** et **l'empathie**. La subjectivité prend en compte la localité des perceptions et des actions, tandis que l'empathie, définie comme la faculté de s'identifier à une personne et de ressentir ce qu'elle ressent, est ici utilisée pour coordonner les actions de nos agents par planification réactive.

Dans une première partie, nous définissons le modèle de notre système multi-agents. Puis,

nous montrons comment nous utilisons la propriété de subjectivité de nos agents, afin de concevoir un MDP subjectif mono-agent. Dans la troisième partie, nous proposons d'utiliser la propriété d'empathie dans un cadre multi-agents où l'environnement est complètement observable pour résoudre un problème de type MMDP. Enfin, nous intégrons ces deux propriétés afin de proposer nos algorithmes de conception de plans réactifs, qui, dans des conditions favorables d'utilisation, garantissent une convergence vers des politiques de bonnes qualités (équilibre de Nash).

2 Modèle du système multi-agents coopératifs proposé

Nous proposons une méthode pour concevoir les agents d'un système multi-agents. Il nous faut préciser la définition des systèmes multi-agents que nous avons choisi d'étudier pour notre problème.

Un système multi-agents est un ensemble d'agents en interaction défini par SMA= $\langle \mathcal{A}, \mathcal{E}, \mathcal{I}, \mathcal{G}, \mathcal{R} \rangle$:
– $\mathcal{A}$: l'ensemble fini d'agents possédant des propriétés de perception locale, de décision (réactive ou non), et d'action ;
– $\mathcal{E}$: l'environnement dans lequel évoluent les agents. Il est constitué de tout ce qui n'est pas "l'agent" en action. On y inclut également les lois de l'environnement. A tout moment l'environnement peut être décrit par une configuration s avec $s \in \mathcal{S}$ l'ensemble des configurations possibles du système.
– $\mathcal{I}$: l'ensemble des interactions possibles entre les agents et l'environnement. A définir selon la topologie du problème que l'on cherche à résoudre.
– $\mathcal{G}$: l'ensemble des objectifs/buts que le système doit atteindre.
– $\mathcal{R} : \mathcal{S} \rightarrow \mathbb{R}$: la fonction de récompense globale du système qui identifie la satisfaction du système. Cette fonction est définie telle que l'on puisse la transformer en récompenses individuelles pour chaque agent et de façon à ce qu'elle conserve la propriété coopérative du système.

La fonction de récompense formalise le problème que doit résoudre notre système. Le fait d'utiliser une récompense implique une possibilité de présenter le problème à résoudre sous la forme d'un problème d'optimisation.

La conception de nos agents réactifs repose sur l'élaboration d'un plan ou d'une politique individuelle. Les propriétés de perception de nos agents de l'environnement nous place dans le cadre d'une observabilité limitée. Le POMDP est le formalisme adapté à ce genre d'agent, mais malheureusement peu réaliste en terme d'application dans un univers multi-agents. Contrairement au POMDP, le MDP est lui beaucoup moins coûteux en complexité et plus simple à utiliser [Puterman, 1994]. Bien entendu, son utilisation dans un environnement non markovien ne garantie plus les propriétés de convergence vers une politique optimale. Notre méthode propose de simuler un DEC-POMDP à l'aide de plusieurs MDPs.

3 Subjectivité mono-agent

3.1 Subjectivité et localité

La notion de subjectivité repose sur le respect de la propriété de localité du paradigme multi-agents que nous désirons conserver dans le cadre des modèles décisionnels de Markov. La subjectivité a pour effet de remplacer le repère du système global ou centralisé d'un agent utilisant un MDP, par un repère local ou ego-centré. L'agent est le centre de son monde, et n'en perçoit que des observations incomplètes.

La figure 1 illustre ce principe. En A, nous sommes confrontés à la vue centralisée usuellement utilisée dans les MDPs centralisés. L'agent situé en bas à gauche doit se rendre à son but que l'on a matérialisé par un triangle. B décrit une vue centrée sur l'agent. Dans ce cas, l'agent ne perçoit que partiellement son environnement. Ses capteurs le renseignent avec une relative précision sur son environnement proche et ne lui donne que des indices dans les régions plus éloignées.

C'est ainsi que nous avons traduit la propriété de localité de nos agents ; adaptée au MDP, cette propriété se concrétise par l'utilisation d'un MDP subjectif. Afin de faciliter notre étude, nous développerons les caractéristiques du MDP subjectif dans un cadre mono-agent.

3.2 MDP subjectif

Définition. Un Processus Décisionnel de Markov fini (MDP) est défini par un tuple $\langle S, A, T, R \rangle$:
– S : un ensemble d'états fini ;

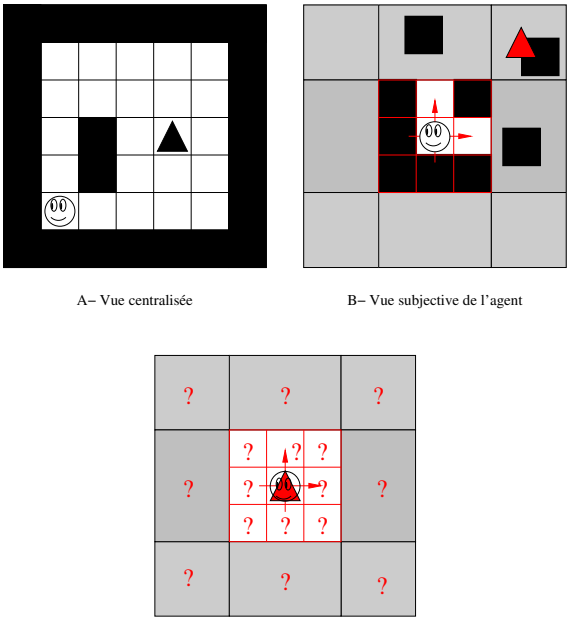


FIG. 1 – Exemple de perception d'un environnement par un agent subjectif.

– A : un ensemble d'actions fini ;
– $T : S \times A \times S \rightarrow [0,1]$: une distribution de probabilité appelée "fonction de transition d'états". A tout instant, la probabilité de passer de l'état s à l'état s' en faisant l'action a est $T(s,a,s')$. Nous supposons que cette probabilité est stationnaire (elle ne varie pas au cours du temps) ;
– $R : S \rightarrow \mathbb{R}$: une fonction de récompense. Elle définit les objectifs.

Sous l'hypothèse que nous recherchons des plans réactifs pour nos agents, nous proposons d'utiliser des MDP subjectifs. Nous définissons un MDP subjectif de la façon suivante[1] :

– S_i devient l'ensemble O_i des perceptions de l'agent ;
– A_i reste l'ensemble des actions ;
– $T_i : O_i \times A_i \times O_i \rightarrow [0,1]$ devient la distribution de probabilité approchée entre les perceptions. Elle reflète l'incertitude des observations de l'agent mais aussi l'incertitude de son comportement.
– Enfin, R est défini sur l'ensemble des perceptions. $R_i : O_i \rightarrow \mathbb{R}$.

[1]Notons que nous appelons MDP subjectif un POMDP dans lequel nous travaillons directement sur les observations avec une vue locale, sans utiliser d'états probables, ni d'historique.

Perception et état agrégé. [Dean *et al.*, 1997] ont travaillé sur les meilleures façons de construire des états agrégés. Les perceptions des MDPs subjectifs sont des états s agrégés. Nous ne traitons pas ce problème dans cet article, mais nous tenons compte des résultats obtenus qui permettent un découpage de l'environnement intéressant.

Fonction de récompense individuelle. La fonction de récompense $R_i : O_i \rightarrow \mathbb{R}$ identifie le but local de l'agent. Reprenons notre exemple, dans ce problème, l'agent doit se rendre sur la case signalée par un triangle quelle que soit sa position initiale dans l'environnement. La figure 1 illustre un exemple des configurations buts possibles. Les informations contenues dans les cases adjacentes n'ont pas d'importance dans ce cas simple.

Comment calculer T ?. Dans le cas d'une observabilité complète, la connaissance de T ne pose aucun problème, il suffit de prendre en compte l'incertitude des actions occasionnées par l'agent lui-même. Dans le cas d'un MDP subjectif, comment calculer T ? Si l'on travaille sur les observations en ayant connaissance des états sous-jacents, c'est-à-dire en connaissant la fonction $O \rightarrow S$, on peut déterminer avec précision la fonction T. Il faut pour cela dénombrer le nombre des états s_i correspondant à une observation o_i et estimer la probabilité de transition que l'on peut associer à une action a_i. Dans le cas contraire, on fait l'hypothèse de ne pas connaître cette fonction, T sera alors calculée à partir du peu d'information disponible dans l'environnement[2].

3.3 Effets de la subjectivité

Comme dans tout problème, les capacités des agents à percevoir et utiliser l'information disponible dans l'environnement sont essentielles. Dans notre cas, nous cherchons à concevoir des agents réactifs qui réagissent à leur perception en suivant un plan simple calculé au préalable, soit par les agents eux-mêmes, soit par un système centralisé, et qui fait correspondre une action à chaque perception. Nos agents ne possèdent pas de mémoire du passé, il nous est donc impossible d'utiliser des techniques faisant appel aux fenêtres d'historique [McCallum, 1995]. Par conséquent, les politiques de nos agents

[2]Notre étude suppose que T peut être estimée ou apprise suffisamment correctement pour pouvoir planifier le comportement de l'agent.

seront très mauvaises dans des environnements de type labyrinthe où une connaissance totale de l'environnement est primordiale pour éviter des performances de l'agent catastrophiques. Toutefois, dans des environnements de grande taille, où les obstacles sont rares et de faible envergure, calculer un MDP subjectif, à l'aide des algorithmes dédiés aux MDPs (*Value Iteration, Policy Iteration,* etc. [Littman, 1996]), apporte les satisfactions liées à la résolution d'un MDP de petite taille avec une complexité en temps de résolution constante quelle que soit la taille de l'environnement. Dans sa thèse, McCallum a mis en valeurs les effets de l'observabilité partielle qui sont de deux ordres [McCallum, 1995] : aider le système à ne pas prendre en compte des détails inutiles, et de ce fait simplifier la résolution du problème, et cacher d'importants détails qui, s'ils sont ignorés, dégradent la qualité des solutions trouvées. C'est dans des conditions favorables que nous proposons d'utiliser nos MDPs subjectifs.

Sous certaines conditions restrictives, calculer un MDP en utilisant des états agrégés ou des observations incomplètes de l'environnement conduisent à l'élaboration d'une politique intéressante en terme de performance. Il s'agit à présent de considérer non plus l'agent réactif, mais bien le système multi-agents dans sa globalité. Pour cela, nous nous intéressons à la deuxième propriété fondamentale de nos agents : l'empathie.

4 Empathie des agents

Dans le domaine de la psychologie, l'empathie se définit comme l'habileté à percevoir, à identifier et à comprendre les sentiments ou les émotions d'une autre personne tout en maintenant une distance affective par rapport à cette dernière. C'est cette propriété qui va nous permettre de coordonner nos agents.

Dans cette section, nous laisserons de coté, temporairement, les contraintes de perception locale que nous imposons à nos agents, afin de nous concentrer sur le problème de la coordination d'actions. Ainsi, nous considérerons le monde comme étant parfaitement observable.

4.1 Notion d'empathie

Considérons le modèle centralisé de MDP Multi-agents (MMDP)[Boutilier, 1999] pour n agents :

- S : ensemble fini des états du monde. Dans cet ensemble apparaît la position de chaque agent, et des objets du monde, c'est-à-dire toutes les informations observables.
- $A = A_1 \times A_2 \times ... \times A_n$ est l'ensemble fini des actions jointes des agents. Elle se définit à partir des ensembles d'actions finis A_i de chaque agent i.
- T et R sont les habituelles matrices de transitions et fonction de récompenses sachant S et A.

La politique calculée π est une politique globale ou jointe. Elle fait correspondre à chaque état global du système une action jointe. A chaque action jointe correspond les décisions simultanées des agents :

$$\pi : S \rightarrow A = A_1 \times A_2 ... \times A_n \qquad (1)$$

Dans notre contexte, calculer une politique jointe de manière centralisée va à l'encontre des principes et des problèmes que nous cherchons à résoudre : la localité et le calcul décentralisé des politiques demeurent les propriétés essentielles de nos agents, il n'est donc pas satisfaisant de faire comme si chaque agent avait une parfaite connaissance du système et du comportement de chacun. De plus, la complexité des méthodes de résolution d'un MMDP est à ce jour trop importante pour rendre réalisable le calcul d'une politique jointe pour un grand nombre d'agents dans un environnement de grande taille.

L'idée que nous voulons mettre en oeuvre afin que nos agents coordonnent leurs actions, repose sur leur capacité à prévoir le comportement de leurs compères et d'adapter ainsi leur propre comportement. Nous proposons, connaissant les politiques de certains agents, de concevoir des algorithmes capables de déterminer les politiques optimales individuelles des autres agents. Formalisons notre approche.

4.2 Formalisme de l'empathie

Si nous reprenons la politique jointe (équation (1)), nous pouvons l'écrire d'une manière équivalente en faisant apparaître le comportement individuel des agents, c'est-à-dire les politiques individuelles de chaque agent ($\pi_i : S \rightarrow A_i$) :

$$\forall s \in S, \pi(s) = (\pi_1(s), ..., \pi_n(s))$$

Supposons qu'un certain nombre $n - m$ parmi les n agents aient une politique déjà définie. On

note :

$$\forall s \in S, \pi(s) = (\pi_1(s), ..., \pi_m(s), ..., \pi_n(s))$$
$$\text{avec } (\pi_i)_{m<i} \text{ connu}$$

La probabilité d'atteindre l'état s' à partir de l'état s dépend uniquement de l'incertitude de comportement des m premiers agents, et donc de leurs actions.

On peut alors définir un nouveau MMDP $M' =< S, A', T', R, \gamma >$ qui concerne seulement les m premiers agents, et auquel on a ajouté la connaissance du comportement des $n-m$ autres agents dans la fonction de transition T'. Autrement dit, nous intégrons la politique individuelle de chaque agent afin de calculer l'action optimale jointe des autres agents. C'est ce principe que nous exprimons à travers la propriété d'empathie de nos agents.

Le nouveau MMDP $M' =< S, A', T', R, \gamma >$ est défini formellement par :

- S l'ensemble fini d'états,
- $A' = A_1 \times .. \times A_m$, l'ensemble fini d'actions jointes des m premiers agents,
- la nouvelle fonction de transition probabiliste calculée à partir de T et des $n - m$ politiques connues :

$$\forall (s, s') \in S^2, \forall (a_1, .., a_m) \in A',$$
$$T'(s, (a_1, .., a_m), s') =$$
$$T(s, (a_1, .., a_m, \pi_{m+1}(s), .., \pi_n(s)), s')$$

- et enfin R, la fonction de récompense qui ne change pas.

Clairement, M' est plus facile à résoudre que M puisqu'il possède un ensemble d'actions plus petit. Dans [Chadès *et al.*, 2002], nous avons montré que si la politique optimale jointe $\pi^* = (\pi_1^*, .., \pi_m^*, .., \pi_n^*)$ est optimale pour M, $\pi' = (\pi_1^*, .., \pi_m^*)$ est une politique optimale jointe pour le nouveau MMDP M' ainsi constitué.

Théorème 1 ([Chadès *et al.*, 2002]) :
Si tous les agents i avec $m < i \leq n$ suivent leur politique individuelle optimale au sens de M, alors la résolution de M' donnera les m' politiques individuelles optimales restantes.
Formellement, notons $(\pi_1^*, .., \pi_n^*)$ la politique optimale jointe de M. Si on définit M' à partir des politiques $(\pi_{m+1}, .., \pi_n) = (\pi_{m+1}^*, .., \pi_n^*)$, alors $(\pi_1^*, .., \pi_m^*)$ est une politique optimale jointe de M'. □

Ce théorème nous permet de concevoir un algorithme de co-évolution itératif alternatif et un algorithme de co-évolution itératif simultané.

4.3 Algorithme itératif de co-évolution alternatif

Le principe de co-évolution a été décrit en biologie comme un phénomène naturel qui illustre un cycle d'évolution : les espèces se transforment et par cette évolution elles transforment leur environnement, qui, lui-même, à son tour, modifie les espèces et ainsi de suite. Nous l'évoquons sous deux approches, l'évolution alternative d'individus, et l'évolution simultanée d'un groupe d'individus.

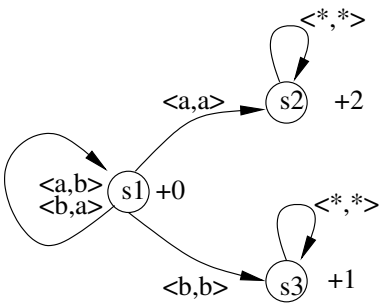
En s'inspirant de ce principe, le premier algorithme proposé traite l'évolution alternative des agents (figure 1). Nous faisons évoluer deux sous-populations en alternance, les changements de politiques du premier groupe permettent de faire évoluer les politiques d'un autre groupe. Le théorème (1) montre que la solution optimale jointe est un point fixe de cet algorithme, ce qui reste vrai pour un groupe composé d'un seul agent.

Algorithme 1 Co-évolution alternative

Entrée: *Un ensemble de politiques individuelles* $(\pi_1^0, ..., \pi_n^0) = \Pi^0$
1: $t \leftarrow 0$;
2: ***Répéter***
3: $m \leftarrow random(n)$;
4: $\mathcal{A}^t \leftarrow (\pi_1^t, ..., \pi_{m-1}^t, \pi_{m+1}^t, \ldots, \pi_n^t)$;
5: $b^t \leftarrow (\pi_m)$;
6: $b^{t+1} \leftarrow ResoudreMMDP(\mathcal{A}^t, b^t)$;
7: $t \leftarrow t + 1$;
8: ***Jusqu'à*** *Convergence vers un équilibre de Nash*
Sortie: $(\pi_1^t, ..., \pi_n^t) = \Pi^t$

On commence avec un ensemble de politiques individuelles arbitraires données en paramètre d'entrée $\Pi^0 = (\pi_1^0, ..., \pi_n^0)$. Puis, à chaque pas de temps t, on fixe les politiques d'un groupe d'agents $\mathcal{A}^t$ (ligne 4). L'agent b^t (ligne 5), trouve la politique optimale complémentaire en incorporant dans son modèle du monde les politiques fixes du premier groupe (ligne 6). La politique π_m ainsi calculée remplace la précédente. Enfin, l'algorithme s'arrête lorsqu'il converge vers un point fixe (ligne 8).

La figure 2 montre qu'il peut exister des points fixes sous-optimaux. Si l'on applique notre al-



Situation A

FIG. 2 – Exemple de convergence possible vers une politique sous-optimale.

gorithme à ce MMDP à deux agents, la politique trouvée peut être de deux types :

1. Si la politique fixée du premier agent $\pi_{\mathcal{A}}$ est égale à :

$$\langle \mathbf{s_1} \rightarrow \mathbf{b}; s_2 \rightarrow a; s_3 \rightarrow a \rangle$$

La politique optimale π_b calculée par l'agent b sera

$$\langle \mathbf{s_1} \rightarrow \mathbf{b}; s_2 \rightarrow *; s_3 \rightarrow * \rangle$$

L'algorithme convergera vers une politique qui est un point fixe sous-optimal de la forme $\pi_{\mathcal{A}\cup b}$ égale à :

$$\langle \mathbf{s_1} \rightarrow < \mathbf{b}, \mathbf{b} >; s_2 \rightarrow < *, * >; s_3 \rightarrow < *, * > \rangle$$

2. Dans le cas favorable, si la politique fixée du premier agent $\pi_{\mathcal{A}}$ est égale à :

$$\langle s_1 \rightarrow a; s_2 \rightarrow a; s_3 \rightarrow a \rangle$$

La politique optimale calculée par l'agent b π_b sera :

$$\langle \mathbf{s_1} \rightarrow \mathbf{a}; s_2 \rightarrow *; s_3 \rightarrow * \rangle$$

L'algorithme convergera vers une politique qui est un point fixe optimal de la forme $\pi_{\mathcal{A}\cup b}$ égale à :

$$\langle \mathbf{s_1} \rightarrow < \mathbf{a}, \mathbf{a} >; s_2 \rightarrow < *, * >; s_3 \rightarrow < *, * > \rangle$$

Dans ces deux cas, le système a atteint un point d'équilibre dans lequel aucun agent ne peut améliorer la politique du système sans diminuer sa fonction de valeur individuelle V_i. Dans cet exemple, les points fixes obtenus par convergence sont des équilibres de Nash, dont certains sont optimaux. La solution optimale ne peut être trouvée à coup sûr, que par une résolution globale du système.

Dans [Chadès, 2003], nous avons montré que si l'algorithme itératif de co-évolution alternative converge, la politique obtenue est un équilibre de Nash.

Théorème 2 ([Chadès, 2003]) :
L'algorithme itératif de co-évolution alternative converge vers un équilibre de Nash qui peut être la politique optimale. □

Nous proposons à présent d'étendre cet algorithme à une évolution simultanée de groupes d'agents.

4.4 Algorithme itératif de co-évolution simultané

L'idée de ce nouvel algorithme est de faire évoluer successivement deux groupes d'agents afin de favoriser l'exploration de l'espace des politiques, et d'améliorer les valeurs des équilibres de Nash rencontrés.

Algorithme 2 Co-évolution simultanée

Entrée: *Un ensemble de politiques individuelles* $(\pi_1^0, ..., \pi_n^0) = \Pi^0$
1: $t \leftarrow 0$;
2: ***Répéter***
3: $m \leftarrow random(n)$;
4: $\mathcal{A}^t \leftarrow random(\{\pi_1^t, ..., \pi_n^t\}; m)$;
5: $\mathcal{B}^t \leftarrow \Pi^t \setminus \mathcal{A}^t$;
6: $\mathcal{B}^{t+1} \leftarrow ResoudreMMDP(\mathcal{A}^t, \mathcal{B}^t)$;
7: $t \leftarrow t + 1$;
8: ***Jusqu'à*** *Convergence vers un équilibre de Nash*
Sortie: $(\pi_1^t, ..., \pi_n^t) = \Pi^t$

On commence avec un ensemble de politiques individuelles arbitraires données en paramètre d'entrée $\Pi^0 = (\pi_1^0, ..., \pi_n^0)$. Puis, à chaque pas de temps t, on choisit deux groupes d'agents. Les agents du premier groupe ont des politiques fixes $\mathcal{A}^t$. Les agents du deuxième groupe, $\mathcal{B}^t$, trouvent la politique optimale complémentaire en incorporant dans leur modèle du monde les politiques fixes du premier groupe. L'ensemble des nouvelles politiques $(\pi_1^t, ..., \pi_n^t)$ ainsi calculé remplace l'ensemble précédent. Enfin, l'algorithme s'arrête lorsqu'il converge vers un point fixe.

Dans [Chadès, 2003], nous avons montré que si l'algorithme itératif de co-évolution simultanée converge, la politique obtenue est un équilibre de Nash. Remarquons que l'algorithme simultané permet de passer d'un point fixe à un autre en améliorant la fonction de valeur globale, tandis que l'algorithme alternatif ne saura pas sortir de ce point fixe. Par conséquent, l'algorithme simultané constitue un bon compromis entre une résolution globale capable de trouver la politique optimale mais cependant impossible à résoudre en terme de complexité. L'algorithme alternatif, quant à lui, diminue la complexité d'une itération mais il est plus souvent sujet aux équilibres de Nash de faibles valeurs.

5 Subjectivité et empathie

Nous avons montré dans les deux sections précédentes, comment nous avons exprimé les propriétés de subjectivité et d'empathie de manière indépendante. Nous proposons à présent de nous intéresser à notre problème dans son ensemble, et nous présentons notre algorithme de conception de plans pour des agents à exécution réactive.

La manière dont peuvent être conçus les plans de nos agents peut se dérouler soit :

De manière décentralisée et fait appel à des agents cognitifs coopérant, capable d'interagir et de communiquer leur plan intermédiaire afin de parvenir à une éventuelle convergence ;

De manière centralisée, connaissant les données d'un problème multi-agents à résoudre, une phase de conception des agents précède leur utilisation.

Dans les deux cas, nous ne pouvons adapter qu'un seul algorithme, l'algorithme alternatif. En effet, bien qu'il soit possible d'estimer de manière approchée la fonction de transition probabiliste pour un groupe d'agents, il n'existe pas, à notre connaissance, de méthode pour résoudre des MMDP subjectifs où plus généralement des Multi-agents POMDPs.

5.1 Système décentralisé : agents cognitifs à exécution réactive

Chaque agent cognitif possède un MDP-subjectif, chaque agent a son propre but qui peut être dépendant de ses compères.

Algorithme décentralisé de conception alternative. L'approche consiste à faire calculer alternativement par chaque agent sa propre politique en tenant compte de celles des autres agents. Ainsi, conformément à l'algorithme itératif de co-évolution alternative, tous les agents commencent avec une politique arbitraire (algorithme (3)).

Algorithme 3 Conception de politique décentralisée (co-évolution alternatif)

Entrée: *Une politique individuelle* π_i^0
1: $t \leftarrow 0$;
2: ***Répéter***
3: ***Si*** *ACTIF* ***Alors***
4: *Recevoir*($\{\Pi^t\}$) ;
5: $\pi_i^t \leftarrow$ *ResoudreMDPsubjectif*(*Calcul* $T_i(\Pi^t)$);
6: ***Sinon***
7: *Envoyer*(π_i^t) ;
8: ***Fin Si***
9: $t \leftarrow t + 1$;
10: ***Jusqu'à*** *Convergence vers un point fixe ou* $t = tMax$
Sortie: π_i^t

A chaque pas de temps, un agent (choisi au hasard) demande aux autres agents leur politique. Il les intègre à son modèle du monde dans sa fonction de transition T_i et il calcule la politique optimale correspondant à sa connaissance. Les agents apprennent ainsi leur politique individuelle pendant un certain nombre d'itérations (*tMax*), car nous ne disposons plus de la preuve de convergence vers l'optimal de la fonction *ResoudreMDPsubjectif*.

Algorithme d'estimation de T_i. Dans le cas d'un MDP subjectif, les états peuvent être des états agrégés, intégrer les politiques des autres dans le modèle de l'agent n'est pas aussi simple que pour un MDP. Pour chaque perception o, l'agent doit calculer $T_i(o, ., .)$ la distribution de probabilité de la fonction de transition. L'algorithme (4) décrit la manière dont l'agent i estime T_i.

- Ligne 2 : A partir de o_i, calculer les distributions d_s^i sur les états pour tous les agents j en utilisant son modèle du monde. L'agent i doit explorer tous les états possibles que comprend l'observation o_i.
- Ligne 4 : A partir des d_s^i, l'agent i obtient les d_o^j les distributions sur les perceptions de tous les agents en utilisant le modèle du monde.
- Ligne 7 : En utilisant les $\{\pi_j\}$ communiqués, l'agent i estime la prochaine distribution de

Algorithme 4 Procédure de calcul de $T_i(o,.,.)$

Entrée: *Un ensemble de politiques individuelles* $(\pi_1, ..., \pi_n) = \Pi$

1: ***Pour tout*** $o_i \in \mathcal{O}_i$ ***Faire***
2: // *Exploration des possibles*
$d^i_s \leftarrow OtoS(o_i, Monde)$
3: ***Pour tout*** *Agent* $j \neq i$ ***Faire***
4: // *Empathie phase 1*
$d^j_o \leftarrow StoO(d^i_s, Monde)$
5: ***Fin Pour***
6: ***Pour tout*** $a_i \in \mathcal{A}_i; j \neq i$ ***Faire***
7: // *Empathie phase 2*
$d'^i_s \leftarrow Estimation(d^j_o, a_i, \{\pi_j\})$
8: ***Fin Pour***
9: $d'^i_o \leftarrow StoO(d'^i_s, Monde)$
10: ***Fin Pour***

Sortie: $d'^i_o = T_i(o,.,.)$

probabilité sur les états des agents d'^i_s.

– Ligne 9 : Enfin, l'agent i convertit d'^i_s en $d'^i_o = T_i(o,.,.)$.

Autrement dit, l'agent i doit estimer à partir de ses perceptions o_i où les autres agents se situent, ce qu'ils perçoivent, ce qu'ils font, et quelles conséquences ont ces actions sur le monde et ainsi sur sa perception o. Par dénombrement des possibles, l'agent i estime $T_i(.,.,.)$.

5.2 Système centralisé : conception d'agents réactifs

Dans le cas d'un système centralisé la démarche est quasi-similaire, il s'agit de calculer le plan des agents avant de les concevoir. Pour cela, un algorithme centralisé prend en charge l'évolution des politiques individuelles de chaque agent. Le bénéfice de cette centralisation s'exprime à travers la définition globale de la fonction de récompense des MDPs subjectifs. Il devient alors possible, en théorie, d'utiliser l'algorithme de co-évolution simultané afin de faire évoluer non plus un seul agent, mais un groupe d'agents.

Algorithme centralisé de conception alternative. Tous les agents commencent avec une politique arbitraire. A chaque pas de temps, la politique d'un agent est améliorée. La fonction de transition T_i est calculée de manière similaire. Puis la résolution du MDP subjectif nous donne la politique approchée correspondant aux informations disponibles. Les agents sont ensuite conçus. Notons qu'il n'y a plus convergence vers un équilibre de Nash, puisque nous travaillons sur l'espace des observations, et que la résolution du MDP subjectif ne converge pas vers une politique optimale au sens des états.

6 Conclusion

Ce travail constitue une approche originale de conception formelle de Systèmes Multi-Agents à l'aide de processus décisionnels de Markov.

Dans cet article, nous avons étudié indépendamment les deux propriétés de subjectivité et d'empathie de notre approche de conception descendante de systèmes multi-agents. Et nous avons proposé différents algorithmes de conception impliquant ces deux principes.

La localité est une propriété essentielle des agents qui constituent les systèmes multi-agents. Concevoir un système multi-agents nécessite donc de prendre en compte cette caractéristique qui nous entraîne dans la prise en compte des perceptions partielles. En dépit de leur adéquation théorique, nous avons pris le parti de ne pas utiliser le formalisme des POMDPs au profit de la simplicité, car leur complexité de résolution dans un contexte multi-agents les rendent inexploitables. Nous avons proposé de contourner la difficulté en choisissant le formalisme des MDPs subjectifs qui dans des conditions d'utilisation favorables (ambiguïté minimale des perceptions, connaissance de l'évolution de l'environnement), nous permettent d'envisager le calcul de politiques de qualité appréciable. De plus, travailler sur l'espace des états subjectifs, nous permet de réduire la complexité de résolution d'un MDP, motivés par notre contexte d'étude multi-agents, nous perdons toutefois les propriétés de convergence vers la politique optimale.

L'empathie est la propriété qui nous permet de prévoir le comportement des agents dans le calcul de politiques jointes ou individuelles. Elle est l'élément coordinateur de leur planification d'actions. Inspirés du comportement biologique, nous avons proposé deux algorithmes de co-évolution, l'un alternatif, l'autre simultané, dans un contexte complètement observable et coopératif. Nous avons montré que tous deux convergeaient vers un équilibre de Nash. Ce résultat théorique constitue une base de référence dans notre approche, et nous laisse espérer que dans des conditions favorables d'observabilité partielle, ces algorithmes livreront des politiques de qualité intéressante.

Enfin, nous avons lié ces deux propriétés en proposant plusieurs méthodes de conception descendantes de SMA coopératifs. Selon les exigences du problème à résoudre, nous sommes en mesure de concevoir, de manière centralisée ou décentralisée, des systèmes multi-agents aux comportements réactifs à l'exécution, connaissant le problème d'optimalité que nos agents coopérants seront amenés à résoudre. De plus, remarquons que le principe de la méthode proposée définit une heuristique de résolution du formalisme DEC-POMDP et permet ainsi de contourner sa complexité.

Références

[Bernstein *et al.*, 2000] D. S Bernstein, S. Zilberstein, et N. Immerman. The complexity of decentralized control of markov decision processes. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, July 2000.

[Boutilier, 1999] C. Boutilier. Sequential optimality and coordination in multiagent systems. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, 1999.

[Chadès *et al.*, 2002] I. Chadès, B. Scherrer, et F. Charpillet. A heuristic approach for solving decentralized-pomdp : Assessment on the pursuit problem. In *Proceedings of the 2002 ACM Symposium on Applied Computing*, 2002.

[Chadès, 2003] I. Chadès. *Planification Distribuée dans les Systèmes Multi-agents à l'aide de Processus Décisionnels de Markov.* Thèse de Doctorat, Université Henri Poincaré, Nancy, France, janvier 2003.

[Dean *et al.*, 1997] T. Dean, R. Givan, et S. Leach. Model reduction techniques for computing approximately optimal solutions for Markov Decision Processes. In *Uncertainty in Artificial Intelligence*, pages 124–131, 1997.

[Dorigo et Di Caro, 1999] M. Dorigo et G. Di Caro. Ant colony optimization : A new meta-heuristic. In *Proceedings of the Congress on Evolutionary Computation*, éditeurs Peter J. Angeline, Zbyszek Michalewicz, Marc Schoenauer, Xin Yao, et Ali Zalzala, volume 2, pages 1470–1477, Mayflower Hotel, Washington D.C., USA, 6-9 1999. IEEE Press.

[Littman, 1996] M. L. Littman. *Algorithms for Sequential Decision Making.* Thèse de Doctorat, Department of Computer Science, Brown University, 1996.

[McCallum, 1995] A. K. McCallum. *Reinforcement Learning with Selective Perception and Hidden State*. Thèse de Doctorat, Department of Computer Science, University of Rochester, December 1995.

[Puterman, 1994] L. Puterman, M. *Markov Decision Processes*. J. Wiley & Sons, 1994.

[Wooldridge, 2002] Michael Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons, February 2002.