Re-formation décentralisée d'équipes sous incertitude : modèle et propriétés structurelles

Jonathan Cohen, Abdel-Illah Mouaddib

Re-formation décentralisée d'équipes sous incertitude : modèle et propriétés structurelles

Jonathan Cohen Abdel-Illah Mouaddib

Université de Caen Normandie
prénom.nom@unicaen.fr

Résumé

Notre travail concerne la formation et la re-formation dynamique d'équipes dans le contexte de la planification multi-agent sous incertitude. Nous cherchons à concevoir un système décentralisé qui repose sur les observations individuelles de chacun de ses agents pour ajuster sa composition à l'évolution de la situation. Nous présentons notre modèle ainsi que quelques unes de ses propriétés structurelles intéressantes. Nous montrons également comme notre modèle généralise les définitions fréquemment rencontrées en théorie des jeux coopérative au cas stochastique.

Mots Clef

Planification sous incertitude, systèmes multi-agents, processus décisionnels de Markov, formation d'équipes.

Abstract

This paper describes our work concerning the dynamic formation and reformation of teams in the context of multi-agent planning under uncertainty. We seek to develop a fully decentralized system which uses solely the individual observations of each of its agents in order to adjust its own composition in regards to the evolution of the task at hand. We present our model along with some of its interesting structural properties. We also show how our model generalizes the definitions usually encountered within the realm of cooperative game theory to the stochastic case.

Keywords

Planning under uncertainty, multi-agent systems, Markov decision processes, team formation.

1 Introduction

La formation d'équipe, c'est-à-dire, la capacité à déterminer le nombre et le rôle adéquats des agents au sein d'un groupe d'agents, est une caractéristique essentielle du travail d'équipe. C'est un pré-requis à la bonne réalisation de nombreuses tâches impliquant des systèmes multi-agents. Dans le domaine de la planification multi-agent, où le but est de parvenir à coordonner un groupe d'agents dans l'optique de résoudre une tâche le plus efficacement possible, choisir préalablement une équipe efficace est primordial.

Considérons le cas d'une réponse d'urgence lancée après une catastrophe naturelle. Déterminer à l'avance le nombre de pompiers, d'ambulanciers et de policiers à déployer sur une zone sinistrée va conditionner la réussite de ces agents à minimiser les pertes humaines et matérielles. Une fois l'équipe fixée (généralement par un expert humain), alors il revient à un planificateur (centralisé ou décentralisé) de trouver la stratégie optimale que cette équipe va devoir suivre pour gérer la situation de façon adéquate. Cela est d'autant plus vrai dans les environnements incertains et stochastiques, où les communications entre agents sont limitées voire impossibles, et où ces mêmes agents ne peuvent pas percevoir la totalité de leur environnement, mais uniquement réaliser des observations locales et imprécises.

Les *processus de décision de Markov décentralisés partiellement observables* (Dec-POMDP) sont devenus le cadre mathématique standard *de facto* pour ce qui est du calcul de politiques (quasi-)optimales multi-agents sous incertitude. Malgré la complexité intrinsèque à résoudre efficacement un Dec-POMDP ayant un grand nombre d'agents ou un large espace d'états, les récentes recherches dans le domaine ont permis de développer des algorithmes de résolution efficaces et ayant de bonnes capacités de passage à l'échelle [5, 16, 15, 10].

Bien qu'ils constituent un cadre formel pratique lorsqu'il s'agit de trouver des stratégies efficaces pour des agents autonomes, les Dec-POMDP ne permettent pas d'aborder directement le problème de la re-formation d'équipe. Dans l'intégralité des travaux existants, l'ensemble des agents est donné en entrée, et la planification a lieu en conséquence, en supposant que l'équipe va rester fixe durant toute l'exécution de la tâche. Or, dans beaucoup de cas réels, il peut arriver qu'une défaillance survienne chez un agent et l'empêche de rester actif. Ou bien, l'évolution de la tâche fait que la présence de certains agents se révèle être contre-productive, ou qu'il existe simplement une équipe plus performante. Dans cet article, on se concentre sur ce second aspect, où le processus d'entrée et sortie des agents est complètement contrôlé. On ne va ainsi considérer que les cas où l'entrée/sortie des agents est volontaire et calculée, omettant ainsi les éventuelles pannes et causes extérieures qui pourraient modifier la structure de l'équipe.

Pour en revenir au scénario de la réponse à une situation

de crise, on peut imaginer que, une fois déployés sur place, les agents aient à faire face à des évènements inattendus, ou qui semblaient peu probables avant leur arrivée sur le site, comme l'effondrement soudain d'un bâtiment, ou la propagation plus rapide que prévu d'un feu. Dans ces différents cas, il pourra ainsi être judicieux d'envoyer, ou bien des véhicules de déblaiement, ou bien davantage de pompiers. Selon l'évolution de la situation, l'équipe sera ainsi capable de se re-former pour minimiser à la fois les coûts liés au déploiement d'unités mobiles lourdes ainsi que les pertes liées à la destruction des infrastructures.
La suite de cet article est organisée de la façon suivante. On commence, en partie 2, par rappeler le modèle théorique des Dec-POMDP. Ensuite, en partie 3, nous introduisons notre modèle, le *Team-POMDP*. La partie 4 énumère et analyse plusieurs définitions et propriétés intéressantes, issues de la théorie des jeux coopérative, étendues au cas stochastique et partiellement observable de notre modèle. Nous apportons également une preuve de la complexité théorique de notre modèle. La partie 5 donne quelques pointeurs vers des travaux relatifs aux nôtres et conclue cet article.

2 POMDPs décentralisés

Afin de mieux situer le contexte et comprendre les enjeux scientifiques liés à notre approche, nous commençons par présenter brièvement le modèle sur lequel on se base : le POMDP décentralisé.

2.1 Modèle

Un *processus de décision de Markov décentralisé partiellement observable (Dec-POMDP)* est défini par un tuple

$$\Psi = (\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{O}, p, r, b^0, T)$$

où :

- $\mathcal{N} = \{1, 2, \ldots, n\}$ est l'ensemble fini des agents ;
- $\mathcal{S}$ est l'ensemble fini des états ;
- $\mathcal{A} = \times_{i \in \mathcal{N}} \mathcal{A}_i$ est l'ensemble des actions jointes, et $\mathcal{A}_i$ est l'ensemble des actions individuelles de l'agent i ;
- $\mathcal{O} = \times_{i \in \mathcal{N}} \mathcal{O}_i$ est l'ensemble des observations jointes, et $\mathcal{O}_i$ est l'ensemble des observations individuelles de l'agent i ;
- $p : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \times \mathcal{O} \to [0, 1]$ est la fonction de dynamique, telle que $p(s, a, s', o) = Pr(s', o \mid s, a)$ est la probabilité de transiter dans l'état s' et d'observer o après avoir effectué l'action jointe a dans l'état s ;
- $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ est la fonction de récompense, telle que $r(s, a)$ est la récompense reçue lorsque les agents effectuent l'action jointe a dans l'état s ;
- $b^0 \in \Delta S$ est la distribution de probabilité initiale sur les états ;
- et T est l'horizon temporel de planification.

Les Dec-POMDP sont une variation multi-agent et partiellement observable des processus décisionnels de Markov (MDP). Dans un Dec-POMDP, les agents sont autonomes à l'exécution, ils *ne communiquent pas entre eux* et ne partagent aucune information directement (si ce n'est par le biais de leurs interactions avec leur environnement commun). De plus, les agents ne perçoivent que des *observations bruitées* de leur environnement. Cela signifie que ni le point de vue individuel d'un agent, ni la combinaison des points de vue de tous les agents, ne permettent à coup sûr de déterminer l'état réel du système. Au début de l'exécution (avant d'être «lancés» dans le système pour qu'ils réalisent leur tâche), les agents n'ont qu'une connaissance *a priori* de l'état dans lequel le système peut être, connaissance apportée par le paramètre b^0 du Dec-POMDP (qui est donc une hypothèse de départ, apportée par exemple par un expert du domaine. Dans beaucoup de cas étudiés, b^0 va être une distribution uniforme).
Avant de décrire en quoi consiste la résolution d'un Dec-POMDP, il faut introduire les concepts d'*historiques* et de *politiques*.

2.2 Historiques et politiques

L'*historique d'observations individuel* h_i^t d'un agent i à l'instant $t < T$ est la suite des observations reçues par l'agent i jusqu'à l'instant t :

$$h_i^t = (o_i^1, \ldots, o_i^t),$$

avec, pour tout $0 \leq t' \leq t$, $o_i^{t'} \in \mathcal{O}_i$.
L'*historique d'observations joint* à l'instant t est la collection des historiques d'observations individuels de tous les agents :

$$h^t = (h_1^t, h_2^t, \ldots, h_n^t).$$

On note $\mathcal{H}_i^t = \times_t \mathcal{O}_i$ l'ensemble des historiques d'observations individuels de l'agent i à l'instant t, et $\mathcal{H}_i = \cup_t \mathcal{H}_i^t$ l'ensemble de tous les historiques d'observations individuels de l'agent i.
La définition des historiques permet d'introduire la notion essentielle de *politique*. Une *politique individuelle* π_i pour un agent $i \in \mathcal{N}$ est une fonction

$$\pi_i : \mathcal{H}_i \to \mathcal{A}_i.$$

On note Π_i l'ensemble des politiques individuelles de l'agent i. Une *politique jointe* est le tuple correspondant à l'ensemble des politiques individuelles des agents de $\mathcal{N}$:

$$\pi = (\pi_1, \pi_2, \ldots, \pi_n),$$

avec $\pi_i \in \Pi_i$ pour tout agent $i \in \mathcal{N}$.

2.3 Résolution

La *valeur* $V(\pi \mid b_0)$ d'une politique jointe π est la récompense accumulée que l'on peut espérer obtenir en suivant cette politique en supposant une distribution initiale b_0 sur les états :

$$V(\pi \mid b_0) \stackrel{def}{=} \mathbb{E}\left[\sum_{t=0}^{T} r(s^t, a^t)\right],$$

où l'espérance $\mathbb{E}$ est sur les états visités et les observations faites par les agents, $a^t \sim \pi$ est l'action jointe prescrite par la politique jointe à l'instant t, et $r(s^t, a^t)$ est la récompense reçue à l'instant t.
Résoudre un Dec-POMDP, c'est trouver une politique jointe π^*, appelée *politique optimale*, telle que sa valeur soit maximale. Les résultats de complexité liés à la résolution d'un Dec-POMDP ne sont pas optimistes : il a été prouvé que trouver une politique jointe pour un Dec-POMDP (avec plus de 2 agents) ayant une valeur d'au moins K est un problème NEXP-complet [5, 18]. Cela signifie qu'il n'existe pas d'algorithme polynomial pouvant trouver une telle politique, et probablement qu'il n'existe pas non plus d'algorithme exponentiel pouvant la trouver. Cette complexité est due à deux facteurs : l'observabilité partielle et le nombre d'agents : dû à la nature décentralisée du problème, un agent ne peut pas maintenir, durant l'exécution, un état de croyance sur l'état du système. Il doit donc se souvenir de l'ensemble de ses observations. La complexité de la résolution augmente ainsi doublement-exponentiellement avec le nombre d'agents et l'horizon de planification.
Les récentes recherches dans le domaine se sont attachées à trouver des politiques quasi-optimales [13, 20, 11] ou bien à exploiter les propriétés structurelles de certaines sous-classes de Dec-POMDP, telles que l'indépendance des agents ou l'observabilité collective [4, 14]. Les approches optimales ont quant à elles proposé des approches efficaces telles que la compression d'historiques et de politiques [6, 3, 17, 10]. Toutefois, les approches permettant un passage à l'échelle efficace (en terme de nombre d'agents, d'actions, d'observations, d'états et d'horizon de planification) sont encore rares et sujettes à d'actives recherches [22, 10].
Maintenant que nous avons présenté les Dec-POMDP et défini ce que sont les historiques, politiques et valeurs de politiques, nous pouvons introduire notre modèle : le *POMDP d'équipes* (Team-POMDP).

3 POMDP d'équipes

Comme nous l'avons vu, l'une de propriétés intrinsèques d'un Dec-POMDP est l'ensemble des agents $\mathcal{N}$. La planification a lieu dans un système clos : l'équipe dans le système n'a pas vocation à être modifiée durant la totalité du processus de décision. L'ensemble des agents $\mathcal{N}$ reste fixe. Nous présentons quelques approches de travail d'équipe *ad hoc* et de (re-)formation d'équipes dans la section 5. Toutefois, à notre connaissance, la planification dans les systèmes multi-agents dits *ouverts* n'a jamais été étudiée auparavant et constitue donc un domaine de recherche entièrement inexploré.

3.1 Modèle

Afin de faire de la formation et re-formation dans les systèmes multi-agents décentralisés, stochastiques et partiellement observables, nous proposons le modèle de POMDP d'équipe.

Définition 1 (POMDP d'équipe). *Un* processus décisionnel de Markov d'équipes partiellement observable *(Team-POMDP) est défini par le tuple*

$$\Phi = (\mathcal{N}, \mathcal{C}, \mathcal{S}, \{\mathcal{A}_C\}, \{\mathcal{O}_C\}, \{p_C\}, \{r_C\}, b^0, T)$$

où

- $\mathcal{N} = \{1, 2, \ldots, n\}$ *est l'ensemble fini des agents ;*
- $\mathcal{C}$ *est l'ensemble des équipes possibles basées sur la population* $\mathcal{N}$*. Dans le cas le plus général, on prendra* $\mathcal{C} \equiv 2^{\mathcal{N}}$*, l'ensemble des parties de* $\mathcal{N}$ *;*
- $\mathcal{S}$ *est l'ensemble fini des états ;*
- $\mathcal{A}_C = \times_{i \in C} \mathcal{A}_i$ *est l'ensemble des actions jointes de l'équipe* $C, \forall C \in \mathcal{C}$*, et* $\mathcal{A}_i$ *est l'ensemble des actions individuelles de l'agent* $i \in \mathcal{N}$ *;*
- $\mathcal{O}_C = \times_{i \in C} \mathcal{O}_i$ *est l'ensemble des observations jointes de l'équipe* $C \in \mathcal{C}$*, et* $\mathcal{O}_i$ *est l'ensemble des observations individuelles de l'agent* $i \in \mathcal{N}$ *;*
- $p_C : \mathcal{S} \times \mathcal{A}_C \times \mathcal{S} \times \mathcal{O}_C \to [0, 1]$ *est la fonction de dynamique de l'équipe* $C \in \mathcal{C}$*, telle que* $p_C(s, a_C, s', o_C) = Pr(s', o_C \mid s, a_C)$ *est la probabilité de transiter dans l'état* s' *et d'observer* o_C *après que l'équipe* C *ait effectué l'action jointe* a_C *dans l'état* s *;*
- $r_C : \mathcal{S} \times \mathcal{A}_C \times \mathcal{S} \to \mathbb{R}$ *est la fonction de récompense de l'équipe* $C \in \mathcal{C}$*, telle que* $r_C(s, a_C, s')$ *est la récompense reçue lorsque l'équipe* C *effectue l'action jointe* a_C *dans l'état* s *et transite dans l'état* s' *;*
- $b^0 \in \Delta S$ *est la distribution de probabilité initiale sur les états ;*
- *et* T *est l'horizon temporel de planification.*

Remarquons que, dans la définition du tuple, on écrit $\{\mathcal{A}_C\}$ pour désigner le fait que les ensembles d'actions jointes sont définies pour chaque équipe $C \in \mathcal{C}$ (et de façon similaire pour les espaces d'observations jointes, et les fonctions de dynamique et de récompense).
Le modèle de Team-POMDP possède quelques caractéristiques communes avec le modèle des Dec-POMDP. Les principales différences viennent de la nature multiple des différents éléments du tuple : plusieurs fonctions de dynamique, de récompenses, et plusieurs ensembles d'actions et d'observations, un pour chaque équipe possible. Le modèle est une généralisation du modèle standard de Dec-POMDP : un Team-POMDP se ramène à un Dec-POMDP lorsque $|\mathcal{C}| = 1$, c'est-à-dire lorsqu'une seule équipe est autorisée/possible. Comme nous le verrons dans la section suivante, cette observation permet d'affirmer que les jeux d'équipes stochastiques ont une complexité au moins égale à celles des Dec-POMDP.
Un Team-POMDP se déroule principalement de la même façon qu'un Dec-POMDP. L'état initial du système à $t = 0$ est $s^0 \sim b^0$. À chaque étape de décision t, une (et une seule) équipe d'agents $C^t \in \mathcal{C}$ est dite *opérationnelle* dans le système. Cette équipe joue une action jointe a_{C^t} et fait transiter

le système d'un état s^t à un état s^{t+1}, puis l'équipe reçoit l'observation jointe $o^{t+1}_{C^{t+1}}$. À l'étape de décision $t+1$, une nouvelle équipe $C^{t+1} \in \mathcal{C}$ (potentiellement la même qu'à l'instant t), devient l'équipe opérationnelle. [1] C'est elle qui va effectuer une action jointe et recevoir une observation jointe. Ce processus se déroule ainsi jusqu'à ce que l'horizon de planification $t = T$ soit atteint. Dans ce travail, on ne considère que des horizons finis.
Le modèle de Team-POMDP vient avec une difficulté supplémentaire par rapport aux Dec-POMDP. Dans ces derniers, les agents reçoivent des observations à chaque instant t. En maintenant leurs historiques d'observations, ils peuvent utiliser leurs politiques individuelles pour décider quelle action choisir. Or, dans un Team-POMDP, les agents qui ne sont pas dans l'équipe opérationnelle ne perçoivent pas d'observations. C'est la raison pour laquelle nous introduisons les notions d'historiques partiels et de politiques d'équipes.

3.2 Historiques partiels et politiques

L'*historique d'observation partiel individuel* $\dot{h}^t_i$ d'un agent i à l'instant $t < T$ est la suite partielle des observations reçues par l'agent jusqu'à l'instant t :

$$\dot{h}^t_i = (\dot{o}^1_i, \dots, \dot{o}^t_i)$$

où

$$\dot{o}^{t'}_i = \begin{cases} o^{t'}_i \text{ si } i \in C^{t-1} \\ null \text{ sinon} \end{cases},$$

pour tout $t' \in [1, t]$ et $o^{t'}_i \in \mathcal{O}_i$. La notation $null$ désigne une observation manquante.
L'*historique d'observations partiel joint* à l'instant t est la collection des historiques d'observations partiels individuels des agents présents dans l'équipe opérationnelle $C^t \in \mathcal{C}$:

$$\dot{h}^t = (\dot{h}^t_i \mid i \in C^t).$$

On note $\dot{\mathcal{H}}_i$ l'ensemble des historiques d'observations partiels individuels de l'agent i. Remarquons qu'un historique d'observations partiel se ramène à un historique d'observations individuel classique, complètement spécifié, lorsque toutes les observations sont présentes. Par souci de concision, nous allons simplement parler d'historique individuel $\dot{h}^t_i$ pour désigner l'historique d'observations partiel individuel d'un agent i (et de façon similaire pour les historiques joints).
Une *politique d'équipe individuelle* $\dot{\pi}^i$ pour un agent $i \in \mathcal{N}$ est une fonction

$$\dot{\pi}_i : \dot{\mathcal{H}}_i \to \mathcal{A}_i \cup \{0, 1\}$$

où $\{0, 1\}$ désigne un espace Booléen où 0 correspond à *Ne pas rejoindre l'équipe* et 1 à *Rejoindre l'équipe*. On note $\dot{\Pi}_i$ l'ensemble des politiques d'équipe individuelles de l'agent i.

1. On ne suppose pas de *temps de transition* d'une équipe opérationnelle à l'autre. Par exemple, on néglige le temps de trajet pour sortir de l'équipe ou pour la rejoindre. On suppose que le changement d'équipe opérationnelle se fait entre deux étapes de décisions $t_1, t_2 < T$.

Une *politique d'équipe jointe* $\dot{\pi}$ est le tuple correspondant à l'ensemble des politiques d'équipe individuelles des agents de $\mathcal{N}$:

$$\dot{\pi} = (\dot{\pi}_1, \dot{\pi}_2, \dots, \dot{\pi}_n),$$

avec $\dot{\pi}_i \in \dot{\Pi}_i$ pour tout agent $i \in \mathcal{N}$.
Comme pour les Dec-POMDP, résoudre un Team-POMDP Φ consiste à trouver une politique d'équipe jointe optimale, notée π^*_Φ, qui va permettre de maximiser la somme espérée des récompenses reçue :

$$\pi^*_\Phi = \underset{\pi_\Phi}{\operatorname{argmax}}\, \mathbb{E}\left[\sum_{t=0}^{T-1} r_{C^t}(s^t, a^t_{C^t})\right].$$

où l'espérance $\mathbb{E}$ est sur les états et les observations faites par les agents, $a^t_{C^t} \sim \pi^*_\Phi$ est l'action jointe prescrite par la politique d'équipe jointe optimale à l'instant t, et $r_{C^t}(s^t, a^t_{C^t})$ est la récompense reçue à l'instant t.
Dans ce travail, on ne cherche pas de méthode pour résoudre de façon (quasi-)optimale un tel problème. Plutôt, on présente dans la section suivante quelques propriétés intéressantes de notre modèle. La recherche de méthodes et d'algorithmes de résolution seront le sujet de prochains travaux.

4 Propriétés et complexité

Nous commençons dans une première partie par présenter quelques propriétés structurelles de notre modèle. Ces définitions et propriétés sont des concepts habituellement rencontrés dans le domaine de la théorie des jeux coopérative, où les agents doivent former des coalitions afin de maximiser leurs gains [19]. Dans une seconde partie, nous donnons quelques résultats de complexité liés à la résolution des Team-POMDP.

4.1 Propriétés structurelles

On commence par introduire l'opérateur de restriction d'un Team-POMDP à un Dec-POMDP, opérateur qui nous servira plus tard à établir notre principal résultat de complexité.

Définition 2 (Restriction). *Soit Φ un Team-POMDP. Le Dec-POMDP $\Phi|_C$ est la* restriction de Φ *aux agents de l'équipe $C \in \mathcal{C}$, et est défini par :*

$$\Phi|_C = (C, \mathcal{S}, \mathcal{A}_C, \mathcal{O}_C, p_C, r_C, b^0, T).$$

Cette définition permet simplement de formaliser la réduction d'un Team-POMDP à un Dec-POMDP lorsqu'on ne considère qu'une seule équipe opérationnelle possible (c'est-à-dire, pour tout $t < T$, $C^t = C$). En somme, il est possible de voir un Team-POMDP comme une collection de Dec-POMDP qui, lors de l'exécution, alternent les uns avec les autres au gré des décisions transmises par la politique d'équipe jointe.
Nous définissons à présent les propriétés de *monotonie*, de *suradditivité* et de *convexité* des jeux d'équipes stochastiques [9].

Définition 3 (**Monotonie**). *Un Team-POMDP* Φ *est* faiblement monotone *si ses fonctions de récompense* $r_C \in \mathcal{R}$ *sont faiblement monotones, c'est-à-dire, pour toutes équipes* $C, C' \in \mathcal{C}$ *telles que* $C \subseteq C'$*, pour tous états* $s, s' \in \mathcal{S}$ *et pour toutes actions jointes* $a_C \in \mathcal{A}_C, a_{C'} \in \mathcal{A}_{C'}$ *telles que* $a_C \subseteq a_{C'}$*, on a :*

$$r_C(s, a_C, s') \leq r_{C'}(s, a_{C'}, s'). \quad (1)$$

Φ *est* (fortement) monotone *si l'inéquation est stricte.*

Si un Team-POMDP est monotone, cela signifie que les agents n'ont pas d'impact négatif à faire partie de l'équipe opérationnelle. Par exemple, si l'introduction d'agents dans l'équipe opérationnelle comporte un coût positif modélisé d'une certain façon dans les fonctions de récompense du modèle, alors le Team-POMDP sera non-monotone (sauf si la perte liée à l'ajout d'un agent est totalement contrebalancée par le gain que cet agent va apporter à l'équipe).
Un exemple de situation monotone est le scénario de réponse d'urgence décrit en introduction. Imaginons un large feu de forêt qui se répand aux abord d'une zone habitée. L'intuition est que plus le nombre de pompiers mobilisés est grand, meilleure sera la réponse. En effet, comme les pompiers n'interfèrent pas en mal les uns avec les autres et ont un coût de déploiement relativement faible comparé à la valeur qu'ils ajoutent à leur équipe, alors, peu importe le nombre de pompiers déployés, il ne peut être que bénéfique à la mission d'ajouter un autre pompier.

Définition 4 (**Suradditivité**). *Un Team-POMDP* Φ *est* faiblement suradditif *si ses fonctions de récompense* $r_C \in \mathcal{R}$ *sont faiblement suradditives, c'est-à-dire, pour toutes équipes disjointes* $C, C' \in \mathcal{C}$ *telles que* $C \cap C' = \varnothing, C \cup C' \in \mathcal{C}$*, pour tous états* $s, s' \in \mathcal{S}$ *et pour toutes actions jointes* $a_C \in \mathcal{A}_C, a_{C'} \in \mathcal{A}_{C'}$*, on a :*

$$r_C(s, a_C, s') + r_{C'}(s, a_{C'}, s') \leq r_{C \cup C'}(s, (a_C, a_{C'}), s'). \quad (2)$$

Φ *est* (fortement) suradditif *si l'inéquation est stricte.*

Si un Team-POMDP est suradditif, alors il est monotone.
Cette définition spécifie que deux équipes disjointes peuvent être (faiblement) meilleures en se regroupant en une seule équipe. On peut dire d'une certaine manière que le tout est supérieur à la somme des parties. De la définition de suradditivité ci-dessus, on peut dériver la définition suivante de convexité.
Une illustration immédiate montrant l'omniprésence de la notion de suradditivité en situation réelle est notre exemple de catastrophe naturelle. On pourrait envoyer une équipe de pompiers pour empêcher le feu de ravager la zone habitée ; ou bien, on pourrait envoyer une brigade de policiers ériger une périmètre de sécurité et procéder à l'évacuation des civils. Ces deux équipes, si envoyées séparément (l'une après l'autre) pourraient efficacement réussir la mission qui leur incombe, mais elles auraient tout intérêt à coopérer et agir en parallèle pour obtenir de meilleurs résultats.

Définition 5 (**Convexité**). *Un Team-POMDP* Φ *est* convexe *si pour toutes équipes* $C, C' \in \mathcal{C}$ *telles que* $C \subseteq C'$*, pour tous états* $s, s' \in \mathcal{S}$*, pour toutes actions jointes* $a_C \in \mathcal{A}_C, a_{C'} \in \mathcal{A}_{C'}$ *telles que* $a_C \subseteq a_{C'}$*, et pour tout agent* $i \in \mathcal{N}, i \notin C'$*, jouant l'action individuelle* $a_i \in \mathcal{A}_i$ *et tel que* $C \cup \{i\} \in \mathcal{C}$*, on a :*

$$r_{C \cup \{i\}}(s, (a_C, a_i), s') - r_C(s, a_C, s') \leq r_{C' \cup \{i\}}(s, (a'_C, a_i), s') - r_{C'}(s, a'_C, s'). \quad (3)$$

Si un Team-POMDP est convexe, alors il est suradditif.
La propriété de convexité d'un Team-POMDP spécifie que la volonté d'un agent quelconque à rejoindre une équipe augmente proportionnellement avec la taille de cette équipe – Lloyd Shapley parle d'*effet boule de neige* [21].
La propriété de convexité est très forte et rarement présente en pratique. Dans les faits, la plupart des systèmes multi-agents ne sont pas convexes mais plutôt *sous-modulaires*. La sous-modularité est liée à la propriété des rendements décroissants : intuitivement, il s'agit de l'idée selon laquelle ajouter de plus en plus d'agents à l'équipe opérationnelle sera de moins en moins profitable, sans pour autant que cela ne devienne préjudiciable. Notre exemple de feu de forêt est sous-modulaire : à partir d'un certain point, ajouter de nouveaux pompiers n'apportera plus aucune valeur si l'équipe opérationnelle est déjà assez grande.

4.2 Complexité

Nous en venons maintenant aux résultats théoriques de notre travail.

Lemme 1. *Soit* Φ *un Team-POMDP. Si* Φ *est suradditif, alors*

$$\pi^*_\Phi = \pi^*_{\Phi|_\mathcal{N}}.$$

Ce résultat indique que pour résoudre de façon optimale un Team-POMDP suradditif Φ, il est suffisant de trouver la politique optimale du Dec-POMDP défini par la restriction de Φ à la grande équipe $\mathcal{N}$ de tous les agents. C'est un résultat qui suit de la définition de suradditivité, où la grande équipe $\mathcal{N}$ de tous les agents sera toujours au moins aussi performante que n'importe quelle autre équipe.

Théorème 1. *Soit* Φ *un Team-POMDP et* $K \in \mathbb{Z}$ *un nombre entier relatif. Trouver une politique d'équipe jointe pour* Φ *générant un gain d'au moins* K *est un problème NEXP-complet.*

Démonstration. Nous nous servons du fait que le problème de résoudre de façon optimale un Dec-POMDP (avec 2 agents ou plus) est également un problème NEXP-complet [5]. Comme nous l'avons vu, si un Team-POMDP est suradditif, alors il suffit de résoudre le Dec-POMDP issu de la restriction du Team-POMDP à la grande équipe $\mathcal{N}$. Comme un Team-POMDP n'est pas nécessairement suradditif, et comme la restriction est un opérateur simplifiant (puisqu'il permet d'omettre une partie des agents, donc des actions et des observations), alors il suit que résoudre un Team-POMDP est au moins aussi dur que résoudre un Dec-POMDP. □

5 Discussion

Notre résultat de complexité donné en théorème 1 révèle l'intuition que l'on peut avoir lorsqu'on regarde au modèle de Team-POMDP : il semble toujours possible de ramener notre modèle au modèle plus classique des Dec-POMDP. L'idée est de doter les agents d'actions et d'observations fictives nop et nob, qui consistent respectivement à *Ne rien faire* et *Ne rien observer*, et de considérer que la grande équipe $\mathcal{N}$ est l'équipe opérationnelle à chaque étape de décision. De plus, l'espace d'états $\mathcal{S}$ devra également considérer l'état de l'équipe. Une telle approche a déjà été étudiée dans un travail précédent [8]. Bien qu'il soit possible d'effectuer cette transformation, le passage d'un Team-POMDP à un Dec-POMDP voile les propriétés structurelles intéressantes de notre modèle, telles que les définitions décrites dans ce travail. Nous pensons qu'il est possible de se servir de certaines de ces propriétés structurelles pour parvenir à trouver des politiques jointes (quasi-)optimales de façon efficace. De plus, la transformation d'un Team-POMDP vers un Dec-POMDP induit des espaces d'états, d'actions jointes et d'observations jointes bien plus grands, ce qui limite l'applicabilité des méthodes de résolution de Dec-POMDP existantes.

Des travaux similaires aux nôtres ont déjà abordé le problème de (re-)formation d"équipes. Récemment, un article sur les *agents non-dévoués* (*non-dedicated agents*) dans les équipes opérant dans des environnements incertains a abordé une facette du travail décrit ici [1]. Les agents non-dévoués sont des agents susceptibles de quitter l'équipe opérationnelle à un moment donné. Dans [1], l'accent est mis sur le départ d'un agent, et sur la façon dont l'équipe opérationnelle va réagir à un tel départ. L'article propose plusieurs heuristiques, telles que le fait de simplement ignorer le départ de l'agent, ou bien de calculer à nouveau, en réaction au départ, une nouvelle politique aux agents restants. Les différences fondamentales avec notre travail vient du fait que pour eux, un agent i a une probabilité Δ_i^t de quitter l'équipe à un instant t, tandis que nous abordons le problème de la formation et re-formation optimale d'équipes pendant l'exécution, où l'entrée/sortie des agents est calculée et déterministe (relativement aux observations stochastiques faites par les agents). De plus, ils ont une vision centralisée et totalement observable du problème.

Également, on peut citer les méthodes de *travail d'équipe ad hoc* [24, 23, 2]. Dans ce contexte particulier, un agent, appelé l'*agent ad hoc*, rejoint un ensemble d'agents équipiers, auparavant inconnus. L'agent ad hoc n'a pas de coordination préalable et il n'y a pas de communication supposée possible entre lui et les autres agents. Peut-être que ceux-ci ont été développés par une tierce-partie. Contrairement à ce que l'on présente avec notre modèle, la plupart des recherches réalisées dans le domaine du travail d'équipe ad hoc se concentre sur le contrôle d'un seul et unique agent, l'agent ad hoc. De plus, celui-ci ne peut pas quitter son équipe et doit faire face à des coéquipiers potentiellement égoïstes, ou tout du moins qui ne vont pas nécessairement avoir à justifier leurs actions auprès de lui. Des extensions récentes aux cas multi-agents existent [7], mais la nature inhérente du travail d'équipe ad hoc rend de toute façon son étude hors de la portée de nos recherches.

Finalement, il pourrait être intéressant d'étudier les jeux à population incertaine [12] ou encore la planification stochastique décentralisée avec anonymat dans les interactions [25]. Il s'agit de modèles de jeux théoriques où les récompenses reçues par les équipes dépendent du nombre (et du type) des agents, et pas seulement sur leurs stratégies individuelles.

Références

[1] P. Agrawal and P. Varakantham. Proactive and reactive coordination of non-dedicated agent teams operating in uncertain environments. In *IJCAI*, 2017.

[2] S. V. Albrecht and S. Ramamoorthy. A Game-Theoretic Model and Best-Response Learning Method for Ad Hoc Coordination in Multiagent Systems. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1155–1156. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[3] R. Aras and A. Dutech. An investigation into mathematical programming for finite horizon decentralized POMDPs. *Journal of Artificial Intelligence Research*, 37 :329–396, 2010.

[4] R. Becker, S. Zilberstein, V. R. Lesser, and C. V. Goldman. Solving transition independent decentralized Markov decision processes. *Journal of Artificial Intelligence Research*, 22 :423–455, 2004.

[5] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of operations research*, 27(4) :819–840, 2002.

[6] A. Boularias and B. Chaib-draa. Exact dynamic programming for decentralized POMDPs with lossless policy compression. In *Proceedings of the Eighteenth International Conference on Automated Planning and Scheduling*, pages 20–27, 2008.

[7] M. Chandrasekaran, P. Doshi, Y. Zeng, and Y. Chen. Can bounded and self-interested agents be teammates ? application to planning in ad hoc teams. *Autonomous Agents and Multi-Agent Systems*, 31(4) :821–860, Jul 2017.

[8] J. Cohen, J. S. Dibangoye, and A.-I. Mouaddib. Open Decentralized POMDPs. In *Proceedings of the 2017 International Conference on Tools for Artificial Intelligence*, 2017.

[9] I. Curiel. *Cooperative Game Theory and Applications : Cooperative Games Arising from Combinatorial Optimization Problems*. Theory and Decision Library C. Springer US, 2013.

[10] J. S. Dibangoye, C. Amato, O. Buffet, and F. Charpillet. Optimally solving Dec-POMDPs as continuous-

state MDPs. *Journal of Artificial Intelligence Research*, 55 :443–497, 2016.

[11] A. Kumar and S. Zilberstein. Point-based backup for decentralized POMDPs : complexity and new algorithms. In *Proceedings of the Ninth International Conference on Autonomous Agents and Multiagent Systems*, pages 1315–1322, 2010.

[12] R. B. Myerson. Population uncertainty and Poisson games. *International Journal of Game Theory*, 27(3) :375–392, 1998.

[13] R. Nair, M. Tambe, M. Yokoo, D. V. Pynadath, and S. Marsella. Taming decentralized POMDPs : Towards efficient policy computation for multiagent settings. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 705–711, 2003.

[14] R. Nair, P. Varakantham, M. Tambe, and M. Yokoo. Networked distributed POMDPs : A synthesis of distributed constraint optimization and POMDPs. In *Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 133–139, 2005.

[15] F. A. Oliehoek and C. Amato. Dec-POMDPs as Non-Observable MDPs. *IAS technical reports*, 2014.

[16] F. A. Oliehoek, M. T. Spaan, and N. A. Vlassis. Optimal and approximate Q-value functions for Decentralized POMDPs. In *Journal of Artificial Intelligence Research*, volume 32, pages 289–353, 2008.

[17] F. A. Oliehoek, M. T. J. Spaan, C. Amato, and S. Whiteson. Incremental clustering and expansion for faster optimal planning in Dec-POMDPs. *Journal of Artificial Intelligence Research*, 46 :449–509, 2013.

[18] C. Papadimitriou and J. Tsitsiklis. The complexity of Markov decision processses. *Mathematics of Operations Research*, 12(3) :441–450, 1987.

[19] D. Ray and R. Vohra. Coalition formation. In *Handbook of game theory with economic applications*, volume 4, pages 239–326. Elsevier, 2015.

[20] S. Seuken and S. Zilberstein. Formal models and algorithms for decentralized decision making under uncertainty. *Journal of Autonomous Agents and Multi-Agent Systems*, 17(2) :190–250, 2008.

[21] L. S. Shapley. Cores of convex games. *International Journal of Game Theory*, 1(1) :11–26, Dec 1971.

[22] M. T. Spaan, F. A. Oliehoek, and C. Amato. Scaling up optimal heuristic search in Dec-POMDPs via incremental expansion. In *Sixth Annual Workshop on Multiagent Sequential Decision Making in Uncertain Domains (MSDM-2011)*, page 63. Citeseer, 2011.

[23] P. Stone, G. A. Kaminka, S. Kraus, J. R. Rosenschein, and N. Agmon. Teaching and leading an ad hoc teammate : Collaboration without pre-coordination. *Artificial Intelligence*, 203 :35–65, October 2013.

[24] P. Stone, G. A. Kaminka, S. Kraus, and J. S. Rosenschein. Ad hoc autonomous agent teams : Collaboration without pre-coordination. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence*, July 2010.

[25] P. Varakantham, Y. Adulyasak, and P. Jaillet. Decentralized stochastic planning with anonymity in interactions. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.