Apprentissage par renforcement et jeux stochastiques à information incomplète

Raghav Aras, Alain Dutech

Apprentissage par Renforcement et Jeux Stochastiques à Information Incomplète

Raghav Aras et Alain Dutech
Equipe MAIA / LORIA (Nancy)
{aras,dutech}@loria.fr

4 avril 2005

Introduction

Le but de notre travail est de permettre à des agents d'*apprendre à coopérer*. Chaque agent étant autonome et, forcément, différent des autres, c'est une tâche particulièrement difficile, surtout si les but des deux agents ne sont pas exactement les mêmes. Notre souci est de travailler avec des agents les plus simples possibles, c'est-à-dire plutôt réactifs (par opposition à cognitif).

Le formalisme des jeux stochastiques ([Myerson, 1991]) offre un cadre intéressant pour modéliser ces problèmes d'apprentissage par renforcement dans les systèmes multi-agents. Dans ce formalisme, la fonction de récompense de chaque agent peut être différenciée et chaque agent apprend *indépendamment* une stratégie ou une politique. S'appuyant sur des résultats classiques de la théorie des jeux, les méthodes actuelles s'attachent surtout à permettre aux agents de trouver des équilibres de Nash (comme par exemple *Nash Q-Learning* de [Hu and Wellman, 2003] ou *Friend or Foe Q-Learning* de [Littman, 2001]). Notre optique est différente, et ce, sur deux points importants :

- **coopération** : trouver un équilibre de Nash n'est pas la garantie de la meilleure collaboration possible entre les agents. Nous voulons donc proposer des algorithmes pour que les agents convergent vers des équilibres qui ne soient pas forcément des équilibres de Nash.
- **information incomplète** : nous nous intéressons à des agents qui n'observent ni les actions ni les récompenses des autres agents, nous démarquant en cela des algorithmes usuels du domaine qui supposent une connaissance totale du modèle.

Nous proposons alors de doter les agents de capacités limitées de communication pour mettre en place une notion similaire aux "contrats" de la théorie des jeux. Si les agents s'accordent sur cette notion de contrat, notre algorithme leur permet de converger vers des équilibres qui induisent des comportements "plus coopératifs" que le simple équilibre de Nash.

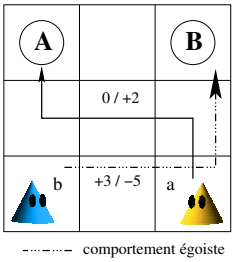

FIG. 1 – L'agent a (resp. b) veut rejoindre la position A (resp. B). Les agents bougent simultanément, et reçoivent -1 s'ils sont sur la même case sauf sur les deux cases spéciales x/y qui rapportent x au premier qui y va, mais qui coûtent y si les agent se croisent.

Exemple

L'exemple de la Figure 1 est représentatif du genre de problème que nous voulons résoudre. Les agents, qui bougent d'une case par tour dans une des 4 directions cardinales, doivent se rendre sur leur case but. Se croiser est pour eux problématique : dans la majorité des cas, ils sont tous les deux pénalisés de 1, sauf dans deux cases spéciales (précisées sur la figure). Si les deux agents adoptent un comportement *égoïste*, ils y perdent tous les deux, mais dès que l'un d'eux n'est pas égoïste (comportement *coopératif*, l'autre sera tenté de le devenir.

En simplifiant un peu, ce problème peut s'exprimer sous une forme un peu plus concise en utilisant un jeu sous forme normale, comme le montre la Table 1. Sous cette forme, on voit que le problème possède deux équilibres de Nash (ego ; coop) mais, que d'un point de vue plus global, il serait plus

	ego	coop
ego	-5 ; -5	+3 ; 0
coop	0 ; +3	+2 ; +2

TAB. 1 – Une représentation normale simplifiée du problème exposé à la Figure 1.

intéressant que les deux agents soient coopératifs et convergent vers un équilibre qui *n'est pas* un équilibre de Nash.

Algorithme

En matière de communication, nous supposons que les agents sont capables d'envoyer un signal *on/off* aux autres agents. Le coeur de notre algorithme repose sur le fait que les agents acceptent une sorte de contrat sur le traitement qu'il feront des communications très simples qui seront échangées. En quelque sorte, ce contrat est une règle de modification interne de la récompense en fonction des messages reçus ou envoyés. L'Algorithme 1 indique comment s'applique ce contrat Γ de manière générale.

Algorithme 1 Compromise Q-Learning

1: L'agent est dans l'état s
2: L'agent choisi action a et envoie message m_s
3: Nouvel état s', message reçu m_r
4: L'agent reçoit une récompense r
5: Met à jour $Q(s,a) \leftarrow (1-\alpha) + \alpha(r + \gamma \max_A Q(s',.))$
6: Modifie $Q(s,a) \leftarrow \Gamma(Q(), m_s, m_r)$

Pour l'instant, nous appliquons des règles différentes en fonction des type d'équilibres présents dans le jeu. Par exemple, quand il y a un équilibre de Nash "opposé"[1], nous proposons un contrat détaillé (Algorithme 2 qui permet de converger vers des comportements coopératifs, ce que nous avons vérifié par la pratique.

Algorithme 2 Contrat Γ pour équilibre "opposé"

1: Soit $\hat{Q}(s) = max_S Q(.,a)$
2: **si** Tous les agents ont envoyé un signal **alors**
3: **si** $Q(s,a) < \hat{Q}(s)$ **alors**
4: $Q(s,a) \leftarrow Q(s,a) - \hat{Q}(s)$
5: **fin si**
6: **fin si**

Nous avons testé plusieurs contrats Γ sur plusieurs types de problèmes (équilibre de Nash, équilibre corrélé, équilibre opposé). De plus, pour chaque contrat, nous avons comparé notre algorithmes avec des algorithmes classiques comme le *Nash Q-Learning*([Hu and Wellman, 2003]), le *Friend or Foe Q-Learning*([Littman, 2001]) ou le *Correlated Q-Learning*([Greenwald and Hall, 2003]).

[1] Adversarial Nash Equilibrium

Conclusion

Notre but est de permettre à des agents non-cognitifs d'apprendre à se coordonner en n'ayant que peu d'informations sur les autres agents. Nous proposons une méthode qui repose sur un protocole de communication limité mais commun et qui permet aux agents de rechercher un "meilleur compromis". Nous avons testé et comparé notre méthode d'apprentissage "meilleur-compromis" à plusieurs autres algorithmes d'apprentissage multi-agents, et ce sur plusieurs types de problèmes différents.

Ce travail ouvre plusieurs perspectives, notamment en ce qui concerne la définition des "contrats" entres les agents. Selon le type de jeu, les contrats sont plus ou moins efficaces et, pour l'instant, les contrats sont définis *a priori* par le concepteur. Notre méthode repose en fait sur la notion de confiance, qui est bien au centre du du problème. En théorie des jeux, l'équilibre de Nash est un équilibre *en pire cas*, c'est-à-dire quand on ne peut pas faire confiance à l'autre joueur. Notre algorithme impose *de fait* cette confiance en faisant l'hypothèse que les deux agents mettent en oeuvre le même traitement de la communication, le même "contrat". Comment les agents peuvent ensuite développer cette confiance, ou apprendre ce genre de contrat est encore un problème ouvert, et difficile...

Références

[Greenwald and Hall, 2003] Greenwald, A. and Hall, K. (2003). Correlated Q-learning. In *Proc. of the 20th Int. Conf. on Machine Learning (ICML)*.

[Hu and Wellman, 2003] Hu, J. and Wellman, M. (2003). Nash Q-learning for general-sum stochastic games. *Journal of Machine Learning Research*.

[Littman, 2001] Littman, M. (2001). Friend-or-foe Q-learning in general-sum games. In *Proc. of the 18th Int. Conf. on Machine Learning (ICML)*.

[Myerson, 1991] Myerson, R. (1991). *Game Theory : Anslysis of Conflict*. Harvard University Press.