Apprentissage de la coordination multiagent

Une méthode basée sur le Q-learning par jeu adaptatif

Olivier Gies — **Brahim Chaib-draa**

Équipe DAMAS
Dépt. Informatique et Génie Logiciel, Université Laval, Québec, Canada G1K 7P4
chaib@damas.ift.ulaval.ca

RÉSUMÉ. *Les algorithmes actuels d'apprentissage multiagent sont pour la plupart limités dans la mesure où ils sont incapables de gérer la multiplicité des équilibres de Nash et de converger vers l'équilibre Pareto-optimal. Pour pallier à cela, nous proposons un algorithme d'apprentissage étendant le Q-learning aux jeux stochastiques non-coopératifs, qui converge (en self-play) vers le Nash Pareto-optimal. Nous présentons des résultats expérimentaux montrant la convergence d'un tel algorithme. Nous étendons ensuite notre approche à un autre aspect essentiel des systèmes complexes qui est la non-stationnarité des agents adverses et qui jusqu'ici a été peu étudié. Finalement, nous abordons la question de la non-stationnarité dans les systèmes multiagents, et présentons des pistes qui nous semblent pertinentes pour améliorer les performances d'adaptation de notre algorithme à des agents non stationnaires.*

ABSTRACT. *Current algorithmes on multiagent learning are for almost limited since they cannot manage the multiplicity of Nash equilibria and thus converge to the Pareto-optimal. To alleviate this, we propose here a learning mechanism extending the Q-learning to non-cooperative stochastique games. This learning mechanism converges to Pareto-optimal equilibria in self-play. We present experimental results showing convergence of such learning mechanism. We then extend our approach to the case of non-stationarity of agents which is another important aspect of multiagent systems. Finally, we tackle the question of non-stationarity in multiagent environments in its generality and we present in this context some research avenues which can lead to improve our preliminary results on adaptation.*

MOTS-CLÉS : *apprentissage multiagent, jeu adaptatif, jeux stochastiques, processus de décision markovien.*

KEYWORDS: *Multiagent Learning, Adaptative Game, Markovien Game, MDP.*

RSTI - RIA – 20/2006. PDM en IA, pages 385 à 412

1. Introduction

Récemment, un intérêt significatif a été porté à l'extension de l'apprentissage par renforcement monoagent (Kaelbling *et al.*, 1996) aux jeux stochastiques[1] (Shapley, 1953). Les jeux stochastiques sont présentés comme un cadre pertinent pour l'apprentissage multiagent par certains chercheurs comme Littman (1994). De tels jeux étendent à la fois le cadre formel des processus décisionnels de Markov, dans lequel est défini l'algorithme d'apprentissage par renforcement du Q-learning (Watkins *et al.*, 1992), et la théorie des jeux (Fudenberg *et al.*, 1994).

L'un des problèmes majeurs dans l'extension de l'apprentissage monoagent aux systèmes multiagent est l'interaction entre agents : les actions individuelles ne peuvent plus être considérées indépendamment des actions des autres agents, car leurs conséquences sont interdépendantes. Un agent seul utilisant l'apprentissage par renforcement peut converger vers une politique optimale face à des agents stationnaires, car la stationnarité des agents adverses peut être incluse dans le modèle de l'environnement, auquel cas le problème revient à un environnement monoagent. Cependant, en présence de non stationnarité induite par d'autres agents (parce qu'ils apprennent ou simplement qu'ils suivent une politique non stationnaire inconnue) l'apprentissage par renforcement monoagent ne permet pas de prendre en compte la présence des autres agents.

Beaucoup d'efforts sont de nos jours portés sur les apports possibles de la théorie des jeux à l'apprentissage multiagent. Parmi les chercheurs les plus actifs, il convient de citer Littman (1994) qui a proposé l'algorithme de minimax-Q learning, dont il a prouvé la convergence pour des jeux purement compétitifs (*i.e.* récompenses opposées). Pour leur part, Claus *et al.* (1998) ont introduit le concept d'agent *joint-action learner* qui apprend la valeur des actions conjointes plutôt que la seule valeur de ses propres actions, et ont prouvé la convergence vers un équilibre de Nash pour les jeux purement coopératifs (récompenses identiques). Les auteurs Hu *et al.* (2003) ont, quant à eux, introduit l'algorithme de NashQ-learning dans le cadre des jeux stochastiques non coopératifs, avec récompenses décorrélées. Dans le même contexte, Greenwald *et al.* (2003) ont proposé une version similaire au NashQ-learning, appelées CE-Q learning (*Correlated Equilibria*) qui apprend en utilisant la valeur des équilibres corrélés (en cas de récompenses corrélées) plutôt que les équilibres de Nash. Littman (2001) a réinterprèté le NashQ-learning, dans l'algorithme *Friend-or-Foe Q-learning*, comme la combinaison d'un algorithme coopératif et d'un algorithme compétitif, et a prouvé la convergence vers un équilibre de Nash pour différentes classes de jeux stochastiques. Tesauro (2004) a, pour sa part, proposé l'algorithme de *Hyper-Q learning*, qui apprend la valeur des stratégies conjointes mixtes plutôt que celles des actions conjointes (*i.e.* stratégies conjointes pures).

À l'exception de deux approches (celles de Wang *et al.* (2003) et de Weinberg *et al.* (2004)) sur lesquelles nous reviendrons plus tard, la plupart des approches relatives

1. Également appelés *jeux de Markov* dans la littérature.

à l'apprentissage multiagent sont limitées, dans la mesure où les algorithmes qu'elles proposent sont incapables de gérer la multiplicité des équilibres de Nash et de converger vers l'équilibre Pareto-optimal si celui-ci existe. De tels algorithmes utilisent en fait une convention pour la sélection de l'équilibre de Nash le plus approprié en cas d'équilibres multiples ainsi que comme méthode d'adaptation lorsqu'il y a lieu de s'adapter à une politique non stationnaire. Pour pallier cela, nous proposons un algorithme qui s'inspire du jeu adaptatif, proposé par Young (1998) et que nous désignons dans la suite par le terme *Q-learning par jeu adaptatif*. Il s'agit d'une variante du jeu fictif (Brown, 1951) dans laquelle les agents ont une mémoire limitée et utilisent une règle de décision bruitée. Young (1998) a prouvé que cette règle de décision peut permettre aux joueurs de se coordonner sur l'équilibre de Nash Pareto-optimal.

2. Concepts préalables

2.1. *Apprentissage monoagent*

En environnement inconnu, l'algorithme de programmation dynamique adaptative (PDA) alterne entre calcul du modèle (récompenses et transitions) et évaluation d'une politique donnée pour le modèle appris. Les algorithmes de la classe "différences temporelles" TD(λ) (*Temporal Difference*), sont une variante de la PDA calculant les fonctions de valeurs de leur politique donnée *indépendamment du modèle*. Plus précisément, étant donnée la politique fixée π, l'apprentissage utilisant les différences temporelles (appelé par la suite TD-learning) parcourt le processus de décision markovien (PDM, ou MDP en anglais) en mettant à jour les fonctions de valeurs des différents états. Pour chaque transition observée entre les états s et s' suite à l'action a, les fonctions de valeurs U sont mises à jour selon la règle suivante :

$$U^{\pi}(s) \leftarrow U^{\pi}(s) + \alpha(R(s,\pi(s)) + \gamma U^{\pi}\left(s'\right) - U^{\pi}(s)) \quad [1]$$

où α est le coefficient d'apprentissage et γ est le coefficient d'actualisation. Notons que le terme pondéré par α correspond à l'égalité dans l'équation générale suivante :

$$U^{\pi}\left(s\right) = R(s,\pi(s)) + \gamma \sum_{s' \in S} T\left(s, \pi\left(s\right), s'\right) U^{\pi}\left(s'\right) \quad [2]$$

équation dans laquelle on a enlevé le terme de probabilité de transition entre les états s et s' selon la politique π, noté $T(s,\pi(s),s')$. Cette simplification permet de s'abstraire du modèle lors de l'apprentissage en considérant chaque transition comme une approximation locale du modèle de transition. Si le modèle de transition n'est pas déterministe, les probabilités sont implicitement prises en compte dans la proportion de transitions observées pour un grand nombre de visite de l'état s. Une telle approche est appelée *apprentissage par renforcement sans modèle*.

L'apprentissage par Q valeurs appelé plus communément "Q-learning" est une méthode de résolution dynamique qui dérive du TD-learning. Elle consiste à apprendre la valeur des actions selon les états, ce qui permet de calculer les utilités *et* la politique optimale dynamiquement. En Q-learning précisément, l'agent possède une fonction $Q : S \times A \mapsto \Re$ qui attribue à chaque couple *état-action* (s,a) une *Q-valeur* $Q(s,a)$,

correspondant à la récompense espérée obtenue en effectuant l'action a dans l'état s et en suivant une politique optimale à partir de l'état suivant :

$$Q(s,a) = R(s,a) + \gamma \sum_{s' \in S} T\left(s,a,s'\right) \max_{a \in A} Q(a,s') \quad [3]$$

Le Q-learning consiste à évaluer cette *Q-fonction* dynamiquement par l'expérience de l'agent dans l'environnement. Étant donnés l'état précédent s, l'action a effectuée en s et l'état courant s', l'agent utilise la règle de mise à jour suivante, qui dérive de la règle de mise à jour du TD learning (équation 1) en remplaçant les utilités pour une politique donnée par les Q-valeurs des paires *(état, action)* :

$$Q(s,a) \leftarrow Q(s,a) + \alpha(s) \left(R\left(s,a\right) + \gamma \max_{a \in A} Q\left(s',a\right) - Q\left(s,a\right) \right) \quad [4]$$

où $\alpha(s) = 1/n(s)$ est un taux d'apprentissage inversement proportionnel au nombre de fois que l'état s a été visité.

La convergence des Q-valeurs vers les Q-valeurs optimales, et par conséquent vers la politique optimale, a été démontrée par Watkins *et al.* (1992) sous l'hypothèse que les couples état-action soient visités une infinité de fois et également sous d'autres conditions restrictives en particulier sur les paramètres d'apprentissage.

Le Q-learning est un algorithme d'apprentissage monoagent, qui peut être utilisé en environnement multiagent, mais sans prendre en compte explicitement la présence des autres agents. La théorie des jeux est un cadre formel qui permet de prendre en compte explicitement la présence des autres agents.

2.2. *Théorie des jeux*

Dans un jeu où le mouvement simultané d'un ensemble de joueurs est constitué d'un seul coup, chaque joueur i choisit simultanément une stratégie[2] $m^i \in M^i$. Le vecteur de stratégies des joueurs est appelé profil de stratégies et il est noté par $m \in M \equiv \times_{i=1}^{N} M^i$, avec N le nombre d'agents. Chaque joueur reçoit alors une utilité (appelée aussi payement ou récompense et pouvant, dans certains cas, rejoindre la fonction valeur d'un PDM). La combinaison de l'ensemble des joueurs, de l'espace des stratégies et les fonctions "utilités" est appelée forme normale ou stratégique d'un jeu. Les stratégies peuvent être "pures" ou "mixtes". Dans le cas des stratégies mixtes, chaque joueur i utilise les stratégies pures de manière aléatoire avec des probabilités notées $\sigma^i \in \Sigma^i \equiv \Delta(M^i)$, et où l'espace de distribution de probabilités est noté par $\Delta(.)$. Les profils de stratégies mixtes sont alors notées $\sigma \in \Sigma = \times_{i=1}^{N} \Sigma^i$.

Partant de là, chaque joueur i a comme utilité espérée : $u^i(\sigma) = \sum_m u^i(m) \Pi_{j=1}^{N} \sigma^j(m^j)$ Là aussi, chaque joueur va tenter de maximiser sa propre

2. En théorie des jeux, une stratégie s^i, au sens de la manIJuvre, est une règle qui indique à l'agent i quelle action il convient de choisir à chaque instant t du jeu. C'est une notion spécifique à la théorie des jeux, et qu'il convient de ne pas confondre avec la stratégie au sens des processus de Markov.

utilité espérée. Bien entendu, cela va dépendre de comment il anticipe les jeux des autres agents. La question soulevée par l'apprentissage via les jeux tente de répondre à cela en formant justement de telles *anticipations*. Supposons pour le moment que i croit que la distribution de probabilités de ses opposants est σ^{-i}. Dans ce cas, i doit jouer la meilleure réponse, c'est à dire une stratégie $\hat{\sigma}^i$ telle que :

$$u^i(\hat{\sigma}^i, \sigma^{-i}) \geq u^i(\sigma^i, \sigma^{-i}) \quad \forall \sigma^i$$

L'ensemble des meilleures réponses (BR pour best responses) à σ^{-i} est noté $BR^i(\sigma^{-i})$, avec bien entendu, $\hat{\sigma}^i \in BR^i(\sigma^{-i})$. La notion de solution en théorie des jeux repose sur le concept d'*équilibre stratégique*, qui correspond à une stratégie conjointe, optimale selon un certain critère. On parle de *Pareto*-optimalité (ou Pareto-efficacité ou efficacité au sens de Pareto) pour une action conjointe si en jouant une autre action conjointe réduit l'utilité d'au moins l'un des joueurs. Une telle action conjointe est appelée un équilibre Pareto-optimal et elle réfère à une propriété "englobant" l'ensemble des intervenants. Formellement, un profil stratégique $\overline{\sigma} = \left(\overline{m}^1, \ldots, \overline{m}^N\right)$ est Pareto-optimal si :

$$\forall \sigma \neq \overline{\sigma},\ \exists j \text{ tel que } u^j(\sigma) < u^j(\overline{\sigma})$$

Une notion d'équilibre plus largement utilisée est celle d'*équilibre de Nash*. Un équilibre de Nash est une action conjointe telle que dévier *individuellement* de son action pour chaque joueur i réduit son utilité propre. En d'autres termes, un équilibre de Nash est une stratégie conjointe où la stratégie de chaque agent est une meilleure réponse au profil stratégique adverse. Formellement, un profil stratégique $\hat{\sigma}$ est un équilibre de Nash si :

$$\hat{\sigma}^i \in BR^i(\hat{\sigma}^{-i}) \quad \forall i$$

Un équilibre de Nash peut être Pareto-optimal, mais les deux notions, à savoir "équilibre de Nash" et "Pareto-optimal", sont différentes. Illustrons cette différence sur le dilemme du prisonnier comme représenté sur le tableau 1 : l'action conjointe $(Nier, Nier)$ est Pareto-optimale car pour chacune de toutes les autres actions conjointes, au moins l'un des deux joueurs a une utilité inférieure à celle de $(Nier, Nier)$. Par contre ce n'est pas un équilibre de Nash, car si l'un des deux joueurs décide de changer individuellement sa stratégie pour jouer l'action $Avouer$, il augmentera sa propre utilité au détriment de celle de l'autre. A l'inverse, l'action conjointe $(Avouer, Avouer)$ est un équilibre de Nash qui n'est pas Pareto-optimal.

Bien que les équilibres Pareto-optimaux semblent meilleurs pour tous les joueurs, les équilibres de Nash sont plus fréquents et plus faciles à déterminer, tout en proposant une notion d'équilibre de meilleure réponse mutuelle. Particulièrement, il existe *toujours* un équilibre de Nash en stratégies mixtes. Le jeu fictif est une méthode de coordination en théorie des jeux, dont la convergence vers un équilibre de Nash (en stratégies pures ou mixtes) a été prouvée pour plusieurs classes de jeux. Nous présentons ce processus dans la section suivante.

		Prisonnier 2	
		Nier	*Avouer*
Prisonnier 1	*Nier*	$(-1,-1)$	$(-10,0)$
	Avouer	$(0,-10)$	$(-5,-5)$

Tableau 1. *Dilemme du prisonnier : les utilités négatives représentent les années de prison encourues*

2.3. *Jeu fictif*

Le jeu fictif[3] est un processus d'apprentissage basé sur la théorie des jeux qui a été établie par Brown (1951). En jeu fictif, les joueurs entretiennent des *croyances empiriques* individuelles sur les stratégies suivies par les autres joueurs. Pour fixer les idées, considérons seulement 2 joueurs ayant des espaces de stratégies finies M^1 et M^2 et les utilités u^1 et u^2. Le modèle de jeux fictifs suppose que les joueurs choisissent leurs actions à chaque période pour maximiser leur utilité espérée, étant donnée leur évaluation des distributions des actions d'autrui durant cette période. Cette évaluation prend la forme suivante. On suppose que i a une fonction de poids initiale qui serait $c_0^i : M^{-i} \rightarrow \Re_+$. Ce poids est mis à jour en ajoutant 1 au poids de chacune des stratégies de son opposant lorsque cette stratégie est jouée :

$$c_t^i\left(m^{-i}\right) = c_{t-1}^i\left(m^{-i}\right) + \begin{cases} 1 & \text{si} \quad m_{t-1}^{-i} = m^{-i} \\ 0 & \text{sinon} \end{cases}$$

Dans ces conditions la probabilité que le joueur i assigne au joueur $-i$ jouant m^{-i} à la date t est donnée par :

$$\gamma_t^i(m^{-i}) = \frac{c_t^i\left(m^{-i}\right)}{\sum_{\tilde{m}^i \in S^i} c_t^i(\tilde{m}^{-i})}$$

On peut maintenant associer une règle par les moyens de $\Gamma(.)$ telle que $\Gamma_t^i(\gamma_t^i)$ fait partie des meilleures réponses de i, soit : $\Gamma_t^i(\gamma_t^i) \in BR^i(\gamma_t^i)$. Dès lors $\Gamma(.)$ indique la meilleure réponse de i quand $-i$ joue m^{-i} avec la probabilité $\gamma_t^i(m^{-i})$. Cependant il n'y a pas qu'une unique règle puisqu'il peut y avoir plusieurs meilleures réponses. Étant données ses croyances sur le profil stratégique adverse γ_t^i, chaque joueur i choisit alors sa réponse aléatoirement dans l'ensemble des meilleures réponses, soit : $m_{t+1}^i = random(BR^i(\gamma_t^i))$

La convergence vers un équilibre de Nash des stratégies des joueurs et des croyances empiriques sur ces stratégies a été prouvée pour plusieurs classes de jeux. Nous référons à Hofbauer *et al.* (2002) pour un aperçu de l'essentiel de ces travaux, et une étude générale de la convergence du jeu fictif stochastique introduit par Fudenberg *et al.* (1998).

3. Ce concept est essentiellement traité en littérature anglophone sous le terme *fictitious play*.

L'inconvénient principal du jeu fictif est sa forte dépendance aux croyances initiales. Prenons par exemple le jeu dit de coopération économique présenté dans le tableau 2. Il modélise l'interaction entre les choix d'un constructeur de supports numériques et d'un constructeur de lecteurs numériques à travers l'intérêt qu'ont les deux joueurs à se coordonner sur le même type de support, avec un avantage arbitraire (meilleure technologie) pour le support DVD, ainsi que les pertes encourues en cas de non coordination. Les actions conjointes (CD, CD) et (DVD, DVD) sont toutes deux des équilibres de Nash en stratégies pures. En supposant que les joueurs ont toujours opté pour l'action CD, leurs croyances empiriques sur l'action adverse est une probabilité de 1 pour l'action CD et de 0 pour l'action DVD. Ils vont donc continuer à jouer l'action conjointe (CD, CD) alors que l'équilibre (DVD, DVD) leur procurerait une utilité supérieure (équilibre Pareto-optimal).

		Entreprise 2	
		CD	DVD
Entreprise 1	CD	$(5, 5)$	$(-1, -3)$
	DVD	$(-4, -1)$	$(9, 9)$

Tableau 2. *Coopération économique modélisant les gains et pertes de deux entreprises fabriquant respectivement des supports numériques et des lecteurs de supports numériques*

Cette forte dépendance aux croyances initiales se traduit également dans l'apparition de cycles stratégiques dégénérés. Si l'on prend l'exemple du jeu présenté dans le tableau 3, et que les joueurs ont leurs compteurs initialisés à $c_0^1\left(A^2\right) = c_0^2\left(A^1\right) = 1$ et $c_0^1\left(B^2\right) = c_0^2\left(B^1\right) = 1.5$. Au premier tour de jeu, chaque joueur va supposer que son adversaire va jouer l'action B. Ils vont donc tous deux jouer leur meilleure réponse à B, à savoir A. Ils mettent à jour leurs compteurs après ce tour de jeu à $c_{t=1}^1\left(A^2\right) = c_{t=1}^2\left(A^1\right) = 2$. Cette fois-ci, chaque joueur croit que le joueur adverse va jouer son action A, action de plus grande probabilité, et joue donc sa meilleure réponse à A, c'est-à-dire B, et ainsi de suite. On voit donc que les croyances initiales ont dans ce cas mené à un *cycle* entre les actions conjointes (A, A) et (B, B), dans lequel les deux joueurs sont perdants.

		Joueur 2	
		A	B
Joueur 1	A	$(0, 0)$	$(1, 1)$
	B	$(1, 1)$	$(0, 0)$

Tableau 3. *Les joueurs ont tout intérêt à faire un choix asymétrique* (A, B) *ou* (B, A)

2.4. *Jeu adaptatif*

Pour surmonter les défauts du jeu fictif, Young (1998) a introduit une variante du jeu fictif appelée *jeu adaptatif*. L'idée principale du jeu adaptatif est, d'une part,

d'introduire une exploration stochastique des actions et, d'autre part, de supprimer l'influence des croyances initiales en utilisant une mémoire limitée. De plus, pour limiter l'influence du bruit introduit par l'exploration des joueurs adverses, seul un échantillon de la mémoire est utilisé pour construire les croyances empiriques sur les stratégies.

Formellement, chaque joueur i garde en mémoire un historique $H_t = \{\sigma_{t-p}, \ldots, \sigma_t\}$ des p dernières stratégies au temps t. Pour chaque joueur adverse j, le joueur i tire aléatoirement et sans remise un échantillon $\hat{H}_t^j = \left\{\sigma_{k_1}^j, \ldots, \sigma_{k_l}^j\right\}$ de l stratégies passées du joueur j prises dans l'historique H_t. La croyance du joueur i sur la stratégie future du joueur j est alors calculée sur l'échantillon $\hat{H}_t^j$ de taille $\left|\hat{H}_t^j\right| = l$ de la manière suivante :

$$\eta_t^i(\hat{H}_t^j) = \frac{n_{\hat{H}_t^j}\left(\sigma_{k_j}^j\right)}{l} \qquad [5]$$

où $n_{\hat{H}_t^j}\left(\sigma_{k_j}^j\right)$ est le nombre de fois qu'apparaît la stratégie $\sigma_{k_j}^j$ dans l'échantillon $\hat{H}_t^j$. Avec $1-\epsilon$, l'agent i joue alors une meilleure réponse à la distribution statistique qu'il a échantillonnée η_t^i et ; avec une probabilité de ϵ, il choisit au hasard son action. Notons que cette méthode d'exploration se retrouve largement au-delà du cadre de la théorie des jeux sous le terme ϵ-*glouton*. Le coefficient d'exploration ϵ peut être fixé (exploration stationnaire) ou variable (par exemple, décroissant si l'on recherche une convergence). On voit aisément qu'avec une telle distribution stochastique, les joueurs 1 et 2 du tableau 3 ne peuvent plus engendrer des cycles comme il le faisaient dans les jeux fictifs.

Young (1998) a démontré que le jeu adaptatif permet de sélectionner l'équilibre de Nash Pareto opimal s'il existe. Comme on peut le constater, la notion de jeu adaptatif est fort intéressante dans un cadre multiagent où pour pleinement l'appliquer tenant compte du stochastique, il faudrait étendre les processus markoviens à la théorie des jeux : les jeux markoviens ou jeux stochastiques constituent une telle extension.

2.5. *Jeux stochastiques*

Les jeux stochastiques sont un cadre formel permettant de modéliser un environnement multiagent non coopératif. Le fait d'être non coopératif signifie que les agents ne poursuivent pas a priori de but commun, mais des objectifs individuels. Ils peuvent cependant être amenés à se coordonner, voire à coopérer, pour atteindre leurs buts individuels. Le cadre formel des jeux stochastiques forment un modèle qui étend les PDM à un cadre multiagent. Il peut aussi être considéré comme une extension à plusieurs états des *jeux en forme normale* de la théorie des jeux. Notons particulièrement que chaque état d'un jeu stochastique peut être vu comme un jeu en forme normale, comme représenté sur la figure 1. Dans la suite, on emploie donc indifféremment les termes *agent* et *joueur*.

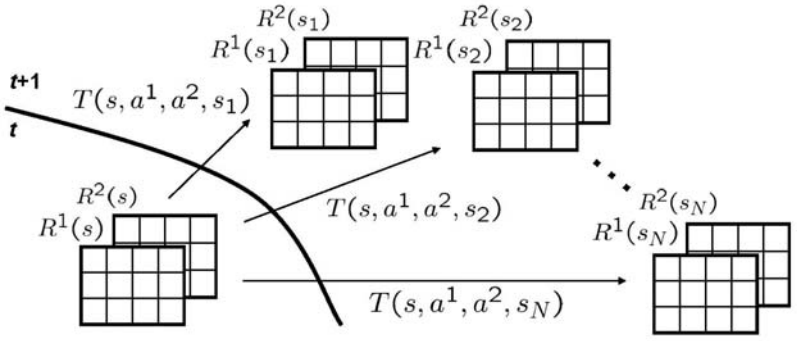

Figure 1. *Jeu stochastique à deux joueurs : transitions possibles entre le tour t et le tour $t+1$ lorsque les joueurs jouent les actions a_1 et a_2 ; chaque état peut être vu comme un jeu en forme normale*

Formellement, un jeu stochastique est un tuple $\langle N, S, \{A^1, \ldots, A^N\}, \{R^1, \ldots, R^N\}, T\rangle$ où N est le nombre d'agents modélisés, S l'ensemble fini d'états du jeu, $A^i = \left\{a^i_1, \ldots, a^i_{|A^i|}\right\}$ l'ensemble d'actions de l'agent i, $R^i : S \times A^1 \times \ldots \times A^N \mapsto \Re$ est la fonction de récompense de l'agent i et $T : S \times A^1 \times \ldots \times A^N \times S \mapsto \Re$ est le modèle de transition entre états, dépendant de l'action conjointe des agents.

A chaque tour de jeu, étant donné l'état courant s, les agents choisissent les actions $a^1, \ldots, a^N$. Ils obtiennent alors les récompenses $\{R^i\left(s, a^1, \ldots, a^N\right)\}_{i\in[1,n]}$ et le système passe dans l'état s' en suivant le modèle de transition T, qui vérifie $\sum_{s'\in S} T\left(s, a^1, \ldots, a^N, s'\right) = 1$. Une politique $\pi^i : S \mapsto [0,1]^{|A^i|}$ pour l'agent i définit une stratégie locale en chaque état au sens de la théorie des jeux. Autrement dit, $\pi^i(s)$ est un vecteur dont les éléments sont des masses de probabilité sur les actions du joueur i, spécifiques au jeu en forme normale défini par l'état s.

Le terme d'utilité espérée d'un joueur en théorie des jeux désigne l'espérance de "récompense" sur les *stratégies des joueurs adverses*, alors que la fonction de valeur en PDM est l'espérance *temporelle* de la récompense. Nous emploierons donc le concept d'utilité U^i en jeu stochastique comme l'espérance temporelle des utilités espérées u^i_s de l'agent i définies pour chaque état s de manière similaire aux utilités espérées des jeux en forme normale. Les utilités U^i des états pour chaque joueur i, associées à la politique conjointe $\check{\pi}(s) \equiv \times_{i=1}^N \pi^i$, sont donc définies comme l'uti-

lité espérée par l'agent i à partir de l'état s si tous les agents suivent cette politique conjointe :

$$U^i_{\pi^1,\ldots,\pi^N}(s) = E\left[\sum_{t=0}^{\infty} \gamma^t u^i_{s_t}\left(\pi^1(s_t),\ldots,\pi^N(s_t)\right) | s_0 = s\right]$$
$$= u^i_s\left(\pi^1(s),\ldots,\pi^N(s)\right) + \gamma \sum_{s'\in S} T\left(s,\pi^1(s),\ldots,\pi^N(s),s'\right) U^i_{\pi^1,\ldots,\pi^N}(s') \quad [6]$$

En jeu stochastique, une politique conjointe $\pi^1_\circ,\ldots,\pi^N_\circ$ est un équilibre de Nash si les stratégies qu'elle définit en chaque état forment un équilibre de Nash (EqNash) pour cet état au sens de la théorie des jeux. Formellement :

$$\pi^1_\circ,\ldots,\pi^N_\circ \text{ est un EqNash}$$
$$\Updownarrow$$
$$\forall s \in S, \pi^1_\circ(s),\ldots,\pi^N_\circ(s) \text{ est un EqNash pour l'état } s$$

De la même manière, on définit pour l'agent i une *politique de meilleure réponse* π^i_{br} aux politiques adverses $\left\{\pi^j\right\}_{j\neq i}$ comme une politique définissant en chaque état une stratégie de meilleure réponse aux stratégies définies par les politiques adverses pour cet état :

$$\pi^i_{br} \text{ politique de meilleure réponse à } \left\{\pi^j\right\}_{j\neq i}$$
$$\Updownarrow$$
$$\forall s \in S,\ \pi^i_{br}(s) \in BR^i(\pi^j(s))$$

Soulignons qu'à l'instar des jeux en forme normale, un équilibre de Nash en jeux stochastiques est une politique conjointe où la politique de chaque agent est une politique de meilleure réponse aux politiques adverses. Le Q-learning est une méthode d'apprentissage *dynamique* qui permet d'apprendre simultanément les utilités des états ainsi que la politique optimale de l'agent en PDM. Le jeu adaptatif, quant à lui, permet à plusieurs joueurs de se coordonner sur un *équilibre optimal* dans un jeu en forme normale, *i.e.* un équilibre de meilleure réponse mutuelle. En utilisant la notion de solution d'équilibre de Nash présente en théorie des jeux, ainsi que le concept de meilleure réponse, nous étendons donc le Q-learning en utilisant le jeu adaptatif pour prendre en compte la présence des autres agents, et ce dans le cadre formel multiagent formé par les jeux stochastiques. Notons qu'un autre avantage du jeu adaptatif est de permettre une adaptation rapide des croyances de l'agent sur les stratégies adverses, même si ces dernières sont non stationnaires. Notre algorithme présente donc des capacités potentielles d'adaptation à des agents non stationnaires, comme nous le verrons plus tard.

3. Apprentissage multiagent par la méthode Q-learning par jeu adaptatif

3.1. *Description de l'algorithme*

Nous proposons un algorithme qui étend le Q-learning monoagent aux jeux stochastiques, en utilisant le *jeu adaptatif* proposé par Young (1998). Dans cette section, nous décrivons l'algorithme du Q-learning par jeu adaptatif ainsi que les modes d'exploration utilisées dans nos expériences. En Q-learning par jeu adaptatif, l'agent i apprend les stratégies adverses *en chaque état s* par jeu adaptatif. Plus précisément, il mémorise $H_t(s)$ un historique des actions conjointes en chaque état, et calcule les probabilités empiriques des actions de l'agent j en chaque état sur un échantillon $\hat{H}_t^j(s)$ des actions de j prises dans l'historique $H_t(s)$. Il sélectionne ensuite la meilleure réponse à sa croyance sur la stratégie conjointe adverse $\breve{\pi}^{-i}(s)$ dans l'état s. Il convient de préciser que le modèle de jeu stochastique étend les jeux en forme normale à plusieurs états et par conséquent l'utilité d'une action conjointe ne dépend plus du seul état courant, mais également des utilités des états futurs. Or le Q-learning permet de tenir compte de l'utilité espérée, au sens de l'espérance temporelle, d'une action conjointe dans un état donné. En Q-learning monoagent, l'utilité d'un état est calculée comme étant la Q-valeur *maximale* de cet état sur toutes les actions possibles. Comme expliqué précédemment, le critère d'optimalité en jeux stochastiques ne peut pas être une maximisation des Q-valeurs sur les actions individuelles des agents, car leurs conséquences dépendent de l'action conjointe. L'extension du Q-learning au cadre multiagent (avec N agents) prend en compte l'influence des actions des autres joueurs par la définition de Q-valeurs état-action conjointe $Q^i\left(s, a^1, \ldots a^N\right)$ pour chaque agent i. Dans ce contexte, on définit formellement une Q-valeur comme suit :

$$Q^i\left(s, a^1, \ldots a^N\right) = R^i\left(s, a^1, \ldots, a^N\right) + \gamma \sum_{s' \in S} T\left(s, a^1, \ldots a^N, s'\right) \Lambda_{\pi^1, \ldots, \pi^N}\left[U^i_{\pi^1, \ldots, \pi^N}\left(s'\right)\right] \quad [7]$$

Dans cette équation 7, $\Lambda_{\pi^1, \ldots, \pi^N}$ est un opérateur générique qui définit la politique conjointe $\pi^1, \ldots, \pi^N$ qui est supposée suivie par les agents à partir de l'état suivant. Cet opérateur définit le critère d'optimalité recherché. En Q-learning monoagent, $\Lambda_\pi = \max_{a \in A}$. Hu *et al.* (2003) ont par la suite introduit l'opérateur $\Lambda_\pi \equiv NashQ$, qui définit la politique conjointe $\pi^1_\circ, \ldots, \pi^N_\circ$ comme un équilibre de Nash pour le jeu stochastique considéré. De manière similaire, Weinberg *et al.* (2004) ont proposé un opérateur $\Lambda(.)$ qui définit une politique conjointe constituée des *politiques stationnaires limites* des autres agents et de la politique de meilleure réponse pour l'agent i.

En ce qui nous concerne, nous adoptons la formulation des Q-valeurs suivante :

$$Q^i\left(s, a^1, \ldots a^N\right) = R^i\left(s, a^1, \ldots, a^N\right) + \gamma \sum_{s' \in S} T\left(s, a^1, \ldots a^N, s'\right) U^i_{\pi^1_\star, \ldots, \pi^i_{br}, \ldots, \pi^N_\star}\left(s'\right) \quad [8]$$

où $\pi^j_\star$ est la politique *non stationnaire réellement suivie* par l'agent j, et π^i_{br} une politique de meilleure réponse du joueur i aux politiques adverses $\left\{\pi^j_\star\right\}_{j\neq i}$. Nous faisons ici l'hypothèse qu'une "bonne approximation" de $\pi^j_\star(s')$ est $\hat{\pi}^j(s')$ qui représente la croyance empirique de l'agent i sur la stratégie de l'agent j dans l'état s', calculée selon la règle (5) du jeu adaptatif. Bien entendu, ceci n'est valable que si $\lim_{x\to\infty}|\pi^j_\star(s') - \hat{\pi}^j(s')| \to 0$, ce qui a été montré par ailleurs (Young, 1998).

Dans ce cas, la formule de mise à jour des Q-valeurs devient alors :

$$\begin{array}{rl} Q^i_{t+1}\left(s,a^1,\ldots a^N\right) \leftarrow & (1-\alpha)Q^i_t\left(s,a^1,\ldots a^N\right) + \alpha[R^i\left(s,a^1,\ldots,a^N\right) \\ & +\gamma u^i_{s'}\left(\hat{\pi}^1(s'),\ldots,\pi^i_{br}(s'),\ldots,\hat{\pi}^N(s')\right)] \end{array} \quad [9]$$

où $\pi^i_{br}(s')$ est une action de meilleure réponse dans l'état s' au profil stratégique adverse reflété par $\hat{\pi}^j(s')_{j\neq i}$. Conformément à ce qu'on a dit auparavant, l'agent i choisit son action dans l'état s' en suivant la règle de décision du jeu adaptatif, *i.e.* $\pi^i_{br}(s') = (1-\epsilon)BR^i(\hat{\pi}^j(s')) + \epsilon.random(A^i)$. Nous reviendrons plus en détail sur ces aspects dans la section 3.3 plus bas.

3.2. *Les étapes de l'algorithme*

Comme notre jeu adaptatif fait appel à des jeux stochastiques, les agents ont des actions conjointes et donc par rapport aux jeux adaptatifs de la section 2.4, il convient de remplacer les stratégies $\sigma^i_{t-p},\ldots,\sigma^i_t$ par $a^i_{t-p},\ldots,a^i_t$. Dans ce cas, l'historique des p dernières stratégies $H_t = \{\sigma^i_{t-p},\ldots,\sigma^i_t\}$ devient $H_t = \{a^i_{t-p},\ldots,a^i_t\}$. Dès lors, et étant donnés la taille mémoire p de l'agent, la taille d'échantillonnage l dans cette mémoire, et le facteur d'exploration stochastique ϵ, le déroulement de l'algorithme est le suivant :

– les Q-valeurs sont initialisées à 0 pour toutes les paires *(état, actions conjointes)* ; les probabilités des actions des agents adverses en chaque état s sont initialisées à l'équiprobabilité ;

– à chaque tour de jeu t, l'agent observe l'état courant s_t ;

– il observe alors l'action conjointe $a^1_t,\ldots,a^N_t$ et met à jour ses historiques concernant les actions passées de son adversaire j pour l'état s_t, $H^i(j,s_t) \leftarrow H^i(j,s_t) \cup \left\{a^j_t\right\}$, en éliminant les actions les plus anciennes pour conserver la taille de l'historique $\left|H^i(j,s_t)\right| = p$;

– il met ensuite à jour les probabilités empiriques des actions adverses dans l'état s_t, en sélectionnant un échantillon $\hat{H}^j \subset H^i(j,s_t)$ de taille $\left|\hat{H}^j\right| = l$, puis en assignant la proportion d'une action du joueur j dans $\hat{H}^j$ à sa probabilité empirique de la jouer, $\eta^i_t(\hat{H}^j_t) = \eta_{\hat{H}^i_t}(a^j_{k_j})/l$, où $\eta_{\hat{H}^i_t}(a^j_{k_j})$ est le nombre d'occurrences de l'action $a^j_{k_j}$ dans l'échantillon $\hat{H}^j$, et ce conformément à l'équation 5 ;

– il sélectionne ensuite une action aléatoire avec une probabilité ϵ ; et avec une probabilité $1 - \epsilon$, une action faisant partie de son ensemble de meilleure réponse $BR_t^i\left(\hat{\pi}^j\left(s'\right)\right)$ comme nous l'avons expliqué à la fin de la section précédente ;

– enfin il met à jour la Q-valeur $Q\left(s_t, a_t^1, \ldots, a_t^N\right)$ selon l'équation 9.

3.3. *Quelques considérations sur l'exploration-exploitation*

Le processus de jeu adaptatif tel que proposé ici introduit une exploration stochastique de l'espace d'actions avec un facteur ϵ, appelée ϵ-*glouton*. Le principe de cette exploration est de sélectionner l'action de meilleure réponse avec une probabilité $1-\epsilon$, et de sélectionner une action aléatoirement avec une probabilité ϵ. Dans notre étude, nous utilisons ce type d'exploration avec deux définitions distinctes de ϵ : exploration stationnaire et exploration GLIE.

L'exploration stationnaire consiste à attribuer une valeur fixée à ϵ : quel que soit le stade d'apprentissage de l'agent, celui-ci choisit une action aléatoire avec une probabilité ϵ *fixée*, et sa meilleure réponse avec la probabilité $1 - \epsilon$. Intuitivement, cette approche présente un problème conceptuel immédiat lorsque l'on recherche une convergence de la politique apprise. Nous utilisons cependant cette méthode pour l'apprentissage offline[4] d'une politique optimale, ainsi que face à un agent non stationnaire.

GLIE est l'acronyme de *Greedy in the Limit with Infinite Exploration*. Cette approche consiste à faire tendre ϵ vers 0 à l'infini. L'idée est de permettre une exploration intensive (gloutonne) au début de l'apprentissage, et d'utiliser exclusivement les données apprises à l'infini. Une manière intuitive de définir ϵ est de lui assigner l'inverse du nombre de tours de jeu depuis le début : $\epsilon = 1/t$. Cette définition est plutôt simpliste, et peut présenter un inconvénient lorsque les espaces d'état et d'actions sont vastes. On utilise dans notre étude une version qui permet d'explorer les actions possibles *en chaque état* : $\epsilon = 1/n(s)$, où $n(s)$ est le nombre de fois que l'état s a été visité.

La section suivante présente le cadre expérimental dans lequel nous avons évalué notre algorithme, comparativement à différents algorithmes d'apprentissage, et utilisant les différents modes d'exploration qu'on vient de voir.

4. Expérimentations

Les performances d'apprentissage du Q-learning par jeu adaptatif sont évaluées dans deux grilles de jeu où se déplacent deux agents, représentées sur les figures 2 et 4. Les jeux se déroulent en temps discret. A chaque pas (tour de jeu), les agents choisissent leurs actions simultanément, et se déplacent dans la grille. Les conséquences

4. Dans ce type d'apprentissage, l'agent apprend une "solution" avant de l'appliquer dans le monde réel. Une fois la "solution" apprise, il pourrait selon des circonstances, l'appliquer à son environnement dans sa totalité, et ce, sans avoir recours aux rétroactions via les capteurs.

des actions sont déterministes : un agent qui effectue l'action $haut$ se déplacera dans la case supérieure au tour suivant avec une probabilité 1 (sauf en cas de collision dans le jeu de coordination). Dans ce même contexte, on définit un *épisode* comme une phase de jeu entre l'état initial et le premier état où un agent atteint son but. Le premier jeu est un jeu de coordination inspiré des travaux de Hu *et al.* (2003). Le second jeu est un jeu d'adaptation à un agent non stationnaire, du type *proie-prédateur*, inspiré des travaux de Weinberg *et al.* (2004).

Ces deux environnements peuvent être modélisés par un jeu stochastique $G = \langle S, \{A^i\}_{i\in\{1,2\}}, \{R^i\}_{i\in\{1,2\}}, T\rangle$ où les états sont les positions conjointes des agents, $S = \{(0,1),(0,2),\ldots\}$ et le modèle de transition T est déterministe, *i.e.* $\forall\left(s, a^1, a^2\right)$ $\exists!\boldsymbol{s'}$tel que $T\left(s, a^1, a^2, s'\right) = 1$.

Les récompenses et les actions sont légèrement différentes selon le jeu, ces deux notions sont précisées dans leurs sections respectives.

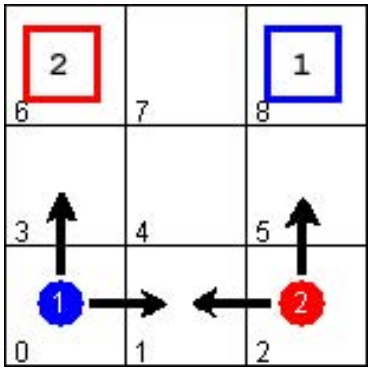

Figure 2. *Jeu de coordination : les agents doivent se coordonner pour atteindre la case dans le coin supérieur opposé à leur position initiale sans se gêner*

4.1. *Jeu de coordination*

Dans le jeu de coordination (figure 2), les deux agents sont initialement positionnés dans les coins inférieurs de la grille. Ils doivent atteindre leurs cases destinations, situées dans les coins supérieurs opposés à leurs positions initiales. S'ils tentent d'aller sur la même case au même tour de jeu, ils entrent en collision et sont replacés sur la case d'où ils viennent. Pour atteindre leur destination en un temps minimal, les agents doivent donc coordonner leurs actions pour ne pas se gêner.

Un agent reçoit une récompense : (a) de $+80$ s'il atteint sa destination en ne suivant aucune case déjà visitée par l'autre agent ; (b) de $80 + 10$ fois le nombre de cases déjà visitées par l'autre agent ; (c) de -1 en cas de collision. Pour toute autre situation, il reçoit une récompense nulle. Les deux agents disposent des actions $\{haut, bas, gauche, droite\}$, dépendant de la case dans laquelle ils se trouvent. Par

exemple, $A^1((0,*)) = \{haut, droite\}$, autrement dit l'agent 1 ne peut choisir que l'action $haut$ ou l'action $droite$ lorsqu'il est sur la case 0 (À noter que $(0,*)$ représente ici tous les états où l'agent 1 est sur la case 0, quelque soit la position de l'agent 2).

Dans le jeu stochastique représentant cet environnement, un équilibre de Nash est une paire de "trajectoires optimales" où chaque agent atteint son but en un *minimum* de pas, sans détour ni collision. En effet, dans cette situation chaque trajectoire est une meilleure réponse à la trajectoire adverse. La figure 3 représente 5 des 10 équilibres de Nash du jeu de coordination, les 5 autres équilibres étant obtenus par symétrie. Dans cette configuration les utilités pour les joueurs sont de haut en bas et de gauche à droite (le premier gain étant celui de l'agent 1) : (90,80), (90,80), (100,80), (80,90) et (80,90), le jeu le plus à droite de la figure 3 et son symétrique sont des configurations où le 2^e joueur suit les pas du 1^{er} joueur dans un plus grand nombre de cases (soit ici 2 cases) et par conséquent ces deux jeux sont considérés comme des équilibres de Nash Pareto-optimaux.

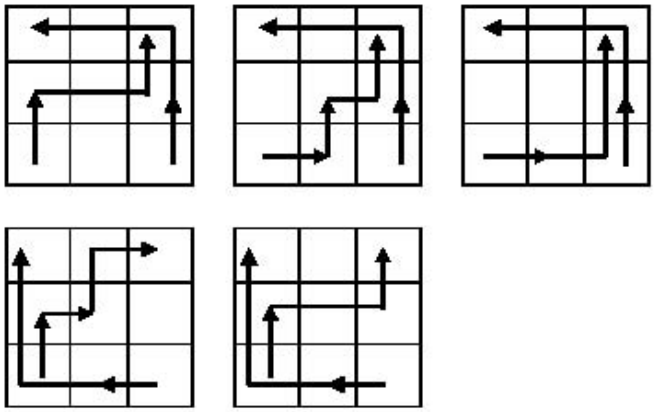

Figure 3. *Équilibres de Nash pour le jeu de coordination*

Nous avons étudié dans ce jeu la convergence empirique vers un équilibre de Nash pour différents algorithmes d'apprentissage (Q-learning classique, Q-learning par jeu fictif, Q-learning par jeu adaptatif) et différents modes d'exploration (exploitation pure, GLIE, exploration stochastique stationnaire).

Dans un premier temps, les agents passent par une phase d'apprentissage *off-line* de 5 000 épisodes. Au début de cette phase, les Q-valeurs des agents sont initialisées à 0, et les croyances sur les stratégies adverses sont initialisées à l'équiprobabilité. Au cours de l'apprentissage, les agents mettent à jour leurs Q-valeurs et leurs croyances, et sont replacés aléatoirement sur la grille après chaque épisode en conservant les données apprises. La durée moyenne d'un épisode durant l'apprentissage est de 4 pas (environ 20 000 pas par apprentissage). Il y a 424 couples $\left(s, a^1, a^2\right)$, donc après 5 000 épisodes, chaque couple *(état, actions conjointes)* a été visité $20\,000/424 \approx 47$

fois. On alors $\alpha\left(s, a^1, a^2\right) \approx 1/47 \approx 0,02$. Les Q-valeurs et les probabilités ne sont donc quasiment plus modifiées. A la fin de l'apprentissage, on observe les politiques apprises par les agents partant de leurs positions initiales. Cette simulation est réitérée 50 fois, après quoi on évalue le pourcentage de simulations ayant mené à un équilibre de Nash.

4.2. *Jeu de poursuite*

Dans le jeu de poursuite, les deux agents débutent le jeu aux positions initiales indiquées sur la figure 4. Le prédateur (agent 1) doit attraper la proie (agent 2) qui évolue dans l'environnement en suivant une politique non stationnaire[5]. Le prédateur reçoit une récompense de +100 lorsqu'il se retrouve à un tour donné sur la même case que la proie. Les deux agents peuvent effectuer les actions $\{immobile, haut, bas, gauche, droite\}$ dans tous les états et la grille de jeu est toroïdale. La politique de la proie est indépendante de l'état s, et elle est définie comme suit :

- $p_{immobile} = 0,2$
- $p_{gauche} = \max(0, \frac{\cos(\theta(t))}{C})$
- $p_{bas} = \max(0, \frac{\sin(\theta(t))}{C})$
- $p_{droite} = \max(0, \frac{cos(\theta(t))}{C})$
- $p_{haut} = \max(0, \frac{sin(\theta(t))}{C})$

où t est le nombre de tours de jeu depuis le début de l'expérience, $\theta(t) = t{*}0,0005 \mod 2\pi$(radians), et $C = 0,2 + \cos(\theta(t)) + \sin(\theta(t))$ normalise les probabilités. Autrement dit, la proie reste sur place avec une probabilité 0,2, et effectue ses autres actions avec une masse de probabilités "tournante" dans le sens trigonométrique. La politique de la proie est donc périodique de période $2\pi/0,0005 \approx 12\ 566$ tours de jeu.

Comme précédemment, nous étudions dans ce jeu la qualité d'adaptation à un environnement non stationnaire pour différents algorithmes d'apprentissage (Q-learning classique, Q-learning par jeu fictif, Q-learning par jeu adaptatif) avec différentes politiques d'exploration (exploitation pure, GLIE, exploration stochastique stationnaire).

Le comportement des algorithmes d'apprentissage face à la non stationnarité de la proie est évalué dynamiquement. Les agents sont placés sur leur positions initiales et le jeu est lancé. Un épisode se termine dès que le prédateur et la proie se retrouvent sur la même case. A chaque nouvel épisode, les agents sont replacés sur leurs positions initiales. On observe l'évolution dynamique de la durée moyenne d'un épisode, qui représente la vitesse à laquelle le prédateur attrape la proie, et par conséquent la

5. Rappelons qu'un processus est dit stationnaire, s'il est gouverné par des lois qui ne changent pas au cours du temps.

qualité d'adaptation du prédateur à la non stationnarité de la politique de la proie. Les expériences sont menées sur 4 000 épisodes. La moyenne de pas par épisode dépend de l'algorithme d'apprentissage utilisé, et varie ainsi de 10 à plus de 20 pas par épisode. Ainsi, l'expérience compte de 40 000 à plus de 80 000 pas, ce qui permet au moins 3 cycles dans la politique de la proie.

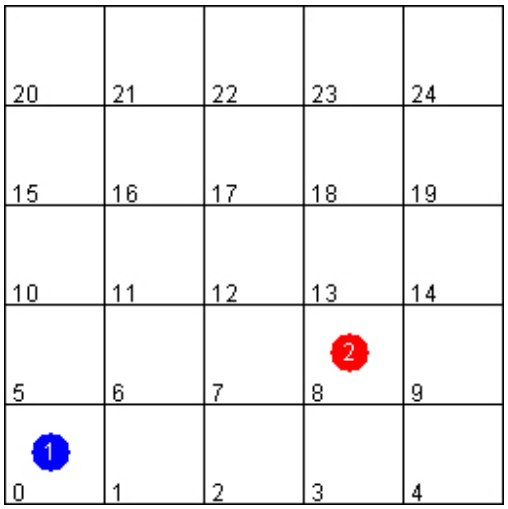

Figure 4. *Jeu de poursuite, du type prédateur-proie. Le prédateur (agent 1) apprend à attraper la proie (agent 2) qui suit une politique non stationnaire*

La durée moyenne d'un épisode est calculée à chaque instant par rapport au nombre total d'épisodes qui ont eu lieu jusqu'à l'épisode courant. Nous analysons la stabilité de cette métrique sur une période de 4 000 épisodes. Pour des agents a priori inadaptés à un environnement non stationnaire, nous attendons donc que cette mesure présente des fluctuations liées à la période du cycle de la stratégie de la proie. Nous étudions par ailleurs la capacité du Q-learning par jeu adaptatif à surmonter cette non stationnarité.

Les hypothèses de convergence du Q-learning classique supposent que l'environnement est stationnaire, particulièrement dans la définition du coefficient d'apprentissage $\alpha(s) = 1/n(s)$. Cette condition est nécessaire pour que les Q-valeurs convergent à l'infini vers les Q-valeurs optimales. Dans notre expérience, le modèle de transition et la fonction de récompense du prédateur sont stationnaires[6]. Cependant, les Q-valeurs sont calculées en chaque instant en utilisant les croyances de l'agent sur les

6. Le fait que le prédateur doit attraper une proie non stationnaire peut laisser penser le contraire, mais il n'en est rien. Pour s'en convaincre, il suffit de réaliser que les états finaux sont fixés (ce sont les états $(k,k)_{k\in[0,24]}$), *i.e.* la proie et le prédateur sont sur la même case. La fonction de récompense donne ainsi toujours une récompense pour les couples *(état, action conjointe)* qui aboutissent à un état final, et rien sinon.

politiques adverses non stationnaires, ce qui implique qu'il n'existe pas de Q-valeurs optimales à l'infini. Pour adapter le Q-learning classique et les différents algorithmes de Q-learning multiagent, nous fixons donc la valeur du coefficient d'apprentissage $\alpha = 0,5$. Il s'agit d'une simplification arbitraire que nous abordons de manière plus approfondie dans la section 6.

Apprentissage	Exploration	EqNash
Jeu fictif pur	sans	10 %
Jeu fictif pur	GLIE	66 %
Jeu fictif pur	ϵ fixé (0,15)	98 %
Jeu adaptatif $(p = 32, l = 16)$	ϵ fixé (0,15)	100 %

Tableau 4. *Pourcentage de convergence vers un EN en self-play*

Apprentissage	Nash Optimal
Jeu fictif pur (ϵ fixé à 0,5)	10 %
Jeu adaptatif $(p = 32, l = 16)$ et ϵ fixé (0,15)	100 %

Tableau 5. *Pourcentage de convergence vers un EqNash Pareto-optimal*

5. Résultats

5.1. *Convergence dans le jeu de coordination*

Le tableau 4 montre la performance en *self-play*[7] du Q-learning par jeu adaptatif par rapport au Q-learning par jeu fictif utilisant différents modes d'exploration. On voit que seule l'exploration stochastique stationnaire permet au Q-learning par jeu fictif pur de converger avec une très forte probabilité (98%), sans pour autant la garantir. Les simulations pour le Q-learning par jeu adaptatif en *self-play* convergent vers un équilibre de Nash dans 100 % des expériences.

Le tableau 5 montre, quant à lui, la convergence vers l'équilibre de Nash Pareto-optimal. Comme on le voit, le fictif à exploration stochastique bien qu'il converge vers le Nash dans 98 % des cas, il ne trouve le Nash Pareto-optimal que dans 10 % des cas. Ceci serait sans aucun doute plus faible s'il n'y avait qu'un seul équilibre Pareto-optimal parmi les dix. L'adaptatif est assuré, quant à lui, de converger vers l'un des deux équilibres Pareto-optimal. Il faudra toutefois, pousser l'expérimentation plus

7. Les deux agents utilisent le même algorithme d'apprentissage

loin et voir si une telle convergence se maintient même dans le cas d'un seul équilibre Pareto-optimal.

La figure 5 montre l'influence comparée du paramètre ϵ pour le Q-learning par jeu fictif (avec exploration stationnaire) et le Q-learning adaptatif avec une mémoire de 32 tours, et une taille d'échantillonnage de 16 actions. On constate que le Q-learning converge dans plus de 98 % des cas pour des valeurs de ϵ prises dans l'intervalle $[0,15 - 0,55]$. Une valeur d'ϵ trop petite s'approche d'une exploitation pure des données apprises, ce qui explique la dégradation des performances pour $\epsilon < 0,15$. La convergence pour des valeurs de ϵ allant jusqu'à 0,55 s'explique par la structure spécifique du jeu. Le nombre d'actions disponibles pour un agent varie de 2 à 4 selon la position dans la grille. Ainsi, dans le pire des cas (4 actions disponibles pour la position 4), choisir l'action réalisant la meilleure réponse avec une probabilité $1 - \epsilon = 0,45$ et une action aléatoire avec une probabilité $\epsilon = 0,55$ revient à choisir l'action de meilleure réponse avec une probabilité de $0,45 + 0,55/4 \approx 0,59$. Pour 3 actions possibles, elle monte à $0,45 + 0,55/3 \approx 0,63$, et pour 2 actions possibles, à $0,45 + 0,55/2 \approx 0,73$. La meilleure réponse est donc choisie avec une probabilité moyenne sur toutes les positions de $0,59 + 4 * 0,73 + 3 * 0,63/1 + 4 + 3 \approx 0,66$ malgré un paramètre d'exploration stochastique $\epsilon = 0,55$. Les probabilités entretenues sur les actions adverses gardent donc une bonne représentativité même si ϵ est élevé. Notons ici que lorsque le nombre d'actions augmente, la probabilité de choisir l'action de meilleure réponse tend vers ϵ.

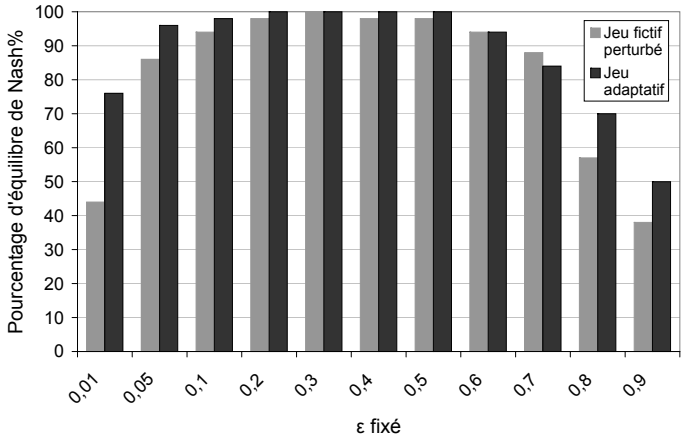

Figure 5. *Influence du paramètre ϵ sur la convergence en self-play en Q-learning par jeu fictif et Q-learning par jeu adaptatif*

La figure 6 indique l'influence de la taille mémoire p et de la taille d'échantillonnage l pour une valeur fixée de ϵ. On observe que la performance de coordination est de 100 % à partir de $(p, l) = (32, 16)$. Pour un échantillonnage limité $(l < 16)$, les échecs de convergence s'expliquent par la nature très approximative des croyances sur

les actions adverses. A l'inverse, une grande taille mémoire permet d'avoir une statistique plus fine de ces actions. Notons que le Q-learning par jeu fictif correspond au Q-learning par jeu adaptatif avec mémoire et taille d'échantillonnage infinies. L'avantage fondamental de la mémoire limitée du jeu adaptatif, *quelle que soit sa taille*, est de supprimer l'influence des premières actions dans les croyances sur les stratégies adverses. En effet, celles-ci sont effectuées lors des toutes premières étapes d'apprentissage alors que les agents ont principalement une attitude d'exploration de l'environnement, et ne sont donc pas représentatives des politiques courantes des agents.

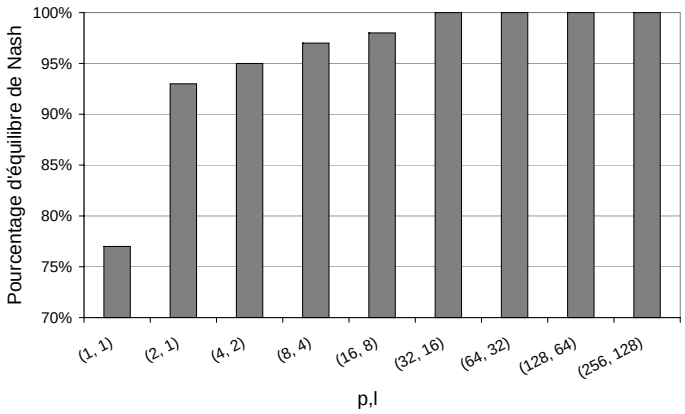

Figure 6. *Influence de de la taille mémoire et de la taille d'échantillonnage sur la convergence du Q-learning par jeu adaptatif en self-play*

Le tableau 6 indique les performances du Q-learning classique en *self-play*, comparées aux performances obtenues en faisant jouer un agent apprenant par Q-learning classique contre un agent Q-learning adaptatif. Sans surprise, le fait que l'agent utilisant le Q-learning par jeu adaptatif en s'appuyant sur un modèle explicite de la stratégie de l'agent adverse améliore les performances de coordination. On peut toutefois s'étonner de la bonne performance en *self-play* du Q-learning classique. Une piste expliquant ces résultats est la nature des états du modèle de jeu stochastique. En effet, bien que le Q-learning simple tienne à jour des Q-valeurs *(état, action)* sans tenir compte de l'action adverse, les états du jeu stochastique définis précédemment incluent la position de l'agent adverse, ce qui procure au Q-learning classique une modélisation implicite de l'autre agent. Notons à ce sujet que le problème essentiel de coordination du Q-learning classique dans notre expérience est la coordination sur un cycle dégénéré entre les états $(3, 5)$ et $(6, 8)$, chaque agent cherchant à éviter la collision dans les cases 4 et 7. Le Q-learning par jeu adaptatif permet précisément d'éviter cette situation.

Apprentissage	*self-play*	vs. jeu adaptatif $p = 32$ $l = 16$ $\epsilon = 0,15$
Q-learning	16 %	28 %
Q-learning avec GLIE	50 %	91 %
Q-learning avec ϵ fixé $(0, 25)$	98 %	100 %

Tableau 6. *Pourcentage de convergence vers un EN pour le Q-learning classique en self-play et contre un agent apprenant par Q-learning par jeu adaptatif*

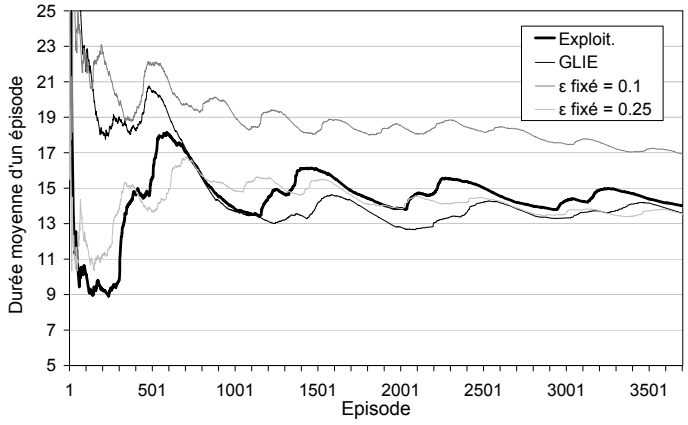

Figure 7. *Durée moyenne d'un épisode lorsque le prédateur apprend par Q-learning classique, avec différents mode d'exploration*

5.2. *Adaptation dans le jeu de poursuite*

Le graphique 7 présente les résultats obtenus pour un prédateur apprenant par Q-learning classique, avec différentes méthodes d'exploration. Indépendamment de la méthode d'exploration, on constate que le Q-learning ne parvient pas à s'adapter à la non stationnarité de la proie. La nombre de pas moyen d'un épisode décroît globalement, mais sa valeur reste fluctuante sans se stabiliser. Le Q-learning reposant sur un principe de rétro-propagation de la récompense entre les états, les fluctuations s'expliquent par l'ancienneté des Q-valeurs utilisées. Le prédateur n'explore pas l'environnement assez rapidement pour qu'assez de Q-valeurs soient à jour relativement à la stratégie courante de la proie. L'exploration est d'ailleurs si lente que

les Q-valeurs redeviennent cohérentes à chaque cycle de la stratégie de la proie. Les résultats montrent également que l'exploration GLIE améliore sensiblement les performances du Q-learning classique, tandis que les améliorations par exploration stationnaire semblent dépendre significativement du facteur d'exploration stochastique ϵ.

Soulignons que le manque de performance du prédateur par Q-learning classique est en partie dû à l'environnement spécifique du jeu de poursuite. Intuitivement, on peut en effet supposer que les performances sont d'autant meilleures que la rapidité de changement de l'environnement est faible vis-à-vis du temps critique nécessaire à l'apprentissage des Q-valeurs. Cette question est soulevée dans la section 6.

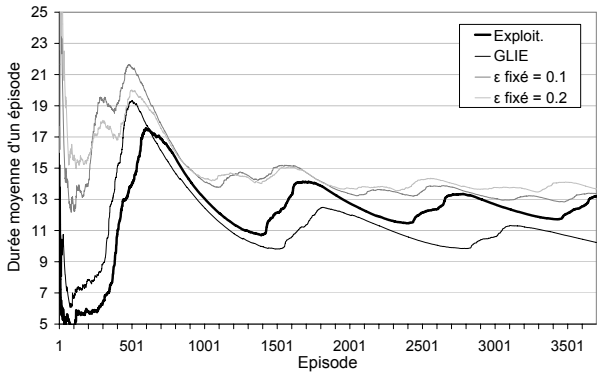

Figure 8. *Durée moyenne d'un épisode lorsque le prédateur apprend par Q-learning par jeu fictif (B̃est-Response Q-learning), avec différents mode d'exploration. (Exploit. signifie exploitation pure sans exploration*

Le jeu de poursuite est inspiré des travaux de Weinberg *et al.* (2004) et y illustre la performance comparée entre Q-learning classique et Q-learning par jeu fictif (NSCP-Q learning) face à un agent *non stationnaire avec limite*. La figure 8 montre les résultats obtenus en Q-learning par jeu fictif dans notre variante non stationnaire *sans limite*[8] du jeu de poursuite, pour différents mode d'exploration. Bien que cette méthode d'apprentissage modélise explicitement la stratégie adverse, la nature non stationnaire de la stratégie de la proie rend les croyances du prédateur inadaptées aux probabilités d'action réelles de la proie. On observe ainsi des fluctuations pseudo-périodiques similaires à celles obtenues en Q-learning classique, qui correspondent aux cycles de la stratégie de la proie. Notons d'une part que, comme pour le Q-learning classique, l'exploration GLIE améliore les performances du Q-learning par jeu fictif, tandis que l'exploration stochastique stationnaire semble les dégrader, le critère de qualité étant

8. Un agent non stationnaire avec limite est un agent dont la politique est non stationnaire à tout instant, mais converge à l'infini vers une politique stationnaire. À l'inverse, un agent non stationnaire sans limite est un agent dont la politique est non stationnaire et ne converge *pas* vers une politique stationnaire à l'infini. Il s'agit donc du cas le plus général.

ici la durée moyenne la plus courte possible. D'autre part, les performances quantitatives du Q-learning par jeu fictif sont toujours meilleures que celles du Q-learning classique : le nombre de pas moyen par épisode tend à fluctuer autour de 13 dans le pire des cas en Q-learning par jeu fictif, tandis que dans le meilleur des cas pour le Q-learning classique, ces fluctuations ne passent pas en-dessous de 13 pas en moyenne par épisode.

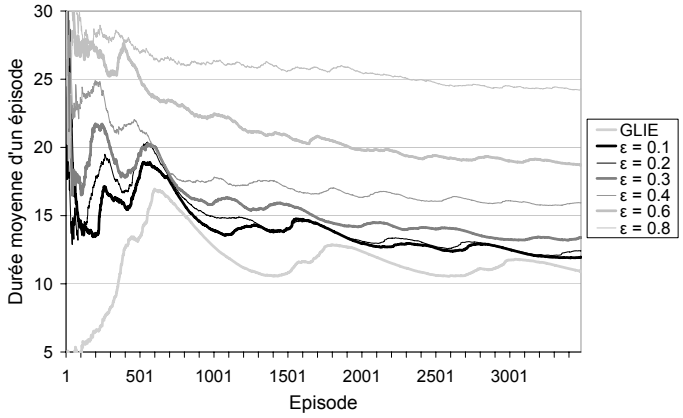

Figure 9. *Influence du mode d'exploration sur l'adaptation du prédateur en Q-learning par jeu adaptatif (mémoire m=16, échantillonnage s=6)*

Le graphique 9 illustre les performances du Q-learning par jeu adaptatif avec exploration GLIE et exploration stationnaire. L'exploration GLIE permet au prédateur d'atteindre la proie plus rapidement en moyenne, mais on constate que l'exploration stationnaire atténue mieux les fluctuations de la durée moyenne d'un épisode. Ces résultats s'expliquent par la nature inadaptée de l'exploration GLIE à la non stationnarité de la proie. En effet, ce mode d'exploration est un compromis optimal entre exploration et exploitation lorsque l'on recherche la convergence optimale des Q-valeurs (et donc de la politique) de l'agent *en environnement stationnaire*, ce qui n'est pas le cas dans le jeu de poursuite. L'agent explorant par GLIE finit donc par exploiter purement les Q-valeurs apprises ce qui donne des résultats similaires au Q-learning par jeu fictif. A l'inverse, l'exploration stationnaire est indépendante du tour de jeu, et permet donc à l'agent d'explorer en tout temps. On peut interpréter ce mode d'exploration comme l'hypothèse qu'à tout instant, les Q-valeurs apprises peuvent être inadaptées à l'environnement courant. Soulignons qu'en accord avec l'intuition, plus l'exploration est fréquente ($\epsilon \rightarrow 1$), plus les performances sont dégradées. Le fait que la durée moyenne d'un épisode soit supérieure à celle obtenue avec GLIE peut s'interpréter de plusieurs façons. D'une part, l'exploration stationnaire correspond concrètement à une règle de décision bruitée, ce qui introduit des actions non optimales qui dégradent les performances. D'autre part, on peut s'interroger sur les capacités propres du Q-learning en environnement non stationnaire. Nous abordons cette question à la section suivante.

L'influence de la mémoire et de l'échantillonnage sont décrits sur le graphique 10. Ces paramètres ne semblent pas influencer significativement la qualité d'adaptation, bien que l'on observe une amélioration pour $(m, s) = (16, 6)$ par rapport aux autres paramètres. Cette légère amélioration peut être interprétée comme un bon compromis entre représentativité de la statistique calculée sur la politique de la proie, et la taille mémoire suffisamment courte pour ne pas prendre en compte des actions périmées. Le graphique 11 représente l'influence de la taille d'échantillonnage s pour une taille mémoire fixée à $m = 16$. Il semble que la durée moyenne d'un épisode converge vers une valeur d'autant plus petite que la taille d'échantillonnage est petite, bien que ces résultats ne permettent pas de juger significativement de l'influence de ce paramètre.

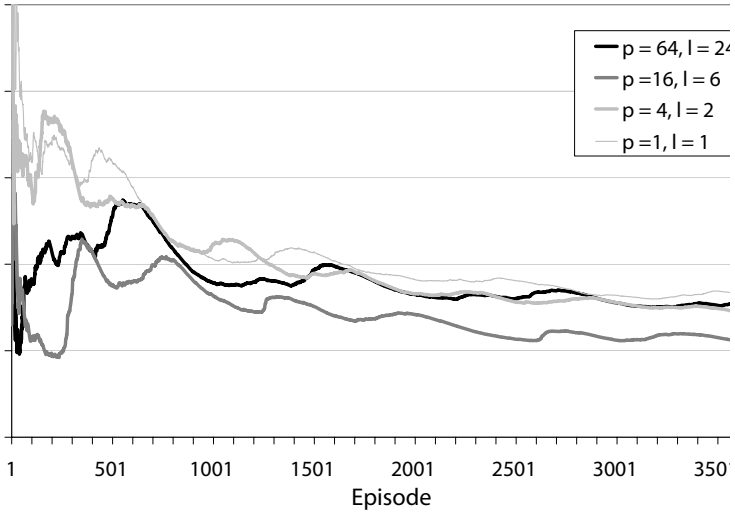

Figure 10. *Influence de la taille mémoire p et de la taille d'échantillonnage l sur l'adaptation du prédateur en Q-learning par jeu adaptatif (exploration stationnaire : $\epsilon = 0,1$)*

Globalement, la proximité des résultats obtenus en faisant varier les paramètres de mémoire laisse penser qu'ils n'ont que peu d'influence sur la qualité d'adaptation du Q-learning par jeu adaptatif. En réalité, nous avons l'intuition que les améliorations qui pourraient éventuellement être obtenues en jouant sur ces paramètres sont occultées par le caractère a priori inadapté du Q-learning à un environnement non stationnaire. En effet, il est important de souligner que dans toutes les approches dérivant du Q-learning, l'apprentissage et l'utilisation de la politique apprise se font simultanément par le biais des Q-valeurs. Ainsi, même si les croyances empiriques sur les stratégies adverses correspondent parfaitement à leur stratégies réelles, les Q-valeurs utilisées ont été calculées à un tour de jeu où celles-ci étaient différentes. Nous approfondissons cette question dans la section suivante.

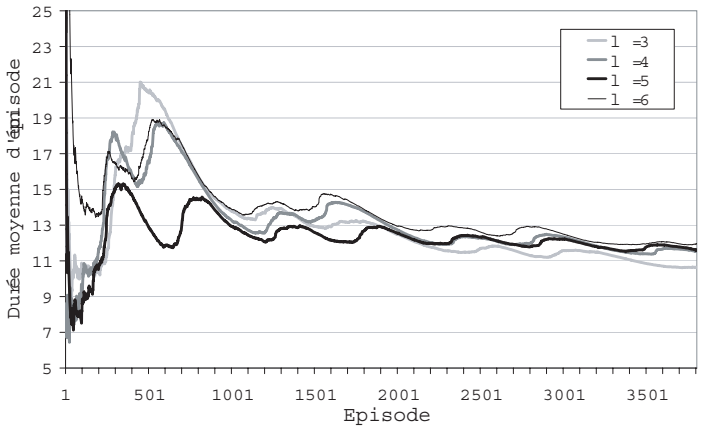

Figure 11. *Influence de la taille d'échantillonnage l sur l'adaptation du prédateur en Q-learning par jeu adaptatif (taille mémoire p =16, exploration stationnaire : ϵ = 0,1)*

6. Discussion

Bien que nous n'ayons pas utilisé de métrique explicite pour évaluer la qualité des croyances du Q-learning par jeu adaptatif sur la stratégie adverse en chaque état, nous postulons que les baisses de fluctuations dans le jeu de poursuite, associées à une durée moyenne d'épisode globalement équivalente au NSCP-Q learning (Weinberg *et al.*, 2004), résultent d'une meilleure qualité des croyances de l'agent. Toutefois, les améliorations apportées par le Q-learning par jeu fictif ne sont pas significatives, et nous attribuons notamment ces résultats mitigés à la règle de mise à jour utilisée. En effet, dans toutes les approches dérivant du Q-learning, l'apprentissage et l'utilisation de la politique apprise se font simultanément par le biais des Q-valeurs. Si l'environnement est stationnaire, un agent seul en Q-learning simple peut donc apprendre les actions optimales. En Q-learning par jeu adaptatif, face un agent non stationnaire, le problème vient du fait que la mise à jour des Q-valeurs utilise les croyances sur les stratégies adverses en chaque état (voir l'équation 9). Ainsi, même si les croyances sur les politiques adverses sont parfaitement exactes à chaque instant, les Q-valeurs utilisées pour déterminer l'action optimale ont été calculées à un tour où ces politiques étaient différentes.

Plus généralement, la problématique soulevée ici est le lien entre la rapidité d'apprentissage de l'agent et la rapidité de changement des politiques non stationnaires des autres agents. Soulignons d'ailleurs que cette problématique s'étend à l'éventuelle non stationnarité de tout le jeu stochastique, incluant les fonctions de récompense et le modèle de transition. Adapter le Q-learning à un environnement non stationnaire implique donc plusieurs modifications indispensables. Tout d'abord, le coefficient d'apprentissage α ne peut plus tendre vers 0, étant donné que les Q-valeurs sont susceptibles de

changer “éternellement”. Nous avons fixé sa valeur à $0,5$ dans nos expériences dans le jeu de poursuite, mais on pourrait imaginer un coefficient dynamique qui évolue en fonction de la qualité d'adaptation de l'agent apprenant. Cela suppose que l'agent bénéficie d'une mesure de sa propre qualité d'adaptation (dans le jeu de poursuite, par exemple, l'agent pourrait se rendre compte qu'il met de plus en plus de temps à atteindre la proie). Cependant, une telle modification correspond à un ajout de connaissance chez l'agent, ce qui est contraire aux efforts de recherche sur la création d'agents minimaux, autonomes et adaptatifs. Une piste tout à fait intéressante développée par Banerjee *et al.* (2003) consiste à optimiser le coefficient α tout en fixant une fenêtre temporelle des w Q-valeurs précédentes d'un couple *(état, action)* pour calculer les nouvelles Q-valeurs. On peut alors jouer sur la *réactivité* de l'agent aux changements dans l'environnement avec les paramètres α et w. Notons toutefois que cette approche est une variante du Q-learning monoagent dans laquelle l'agent apprenant n'entretient aucun modèle des autres agents.

Un autre effort de recherche pourrait être porté sur la rapidité de convergence des algorithmes de Q-learning multiagent. L'idée est ici de considérer la non stationnarité comme une succession de périodes stationnaires d'une durée minimale L donnée (approche similaire à celle proposée par Weinberg *et al.* (2004)). La qualité d'un algorithme adaptatif correspondrait alors à la qualité de la politique apprise après L tours de jeu. Soulignons toutefois que pour appliquer ces résultats à un cadre effectivement non stationnaire, un mécanisme de détection du changement de période est nécessaire à l'agent. Si la non stationnarité est continue, la problématique se rapproche de celle de la *moving target function* en systèmes multiagent, pour laquelle Vidal *et al.* (2003) proposent un cadre d'étude formel appelé CLRI (*Change, Learn, Retention and Impact*). Notons cependant que le cadre proposé n'est pas markovien, dans le sens où les états utilisés pour évaluer l'adaptation des agents sont tirés aléatoirement dans l'espace d'états, sans tenir compte du modèle de transition. Bien que cet aspect rende a priori caduque l'étude des extensions du Q-learning dans ce cadre, celui-ci apporte une lumière intéressante sur les critères d'évaluation pertinents pour l'apprentissage multiagent en environnement non stationnaire.

7. Conclusion

Nous avons présenté dans cet article un algorithme d'apprentissage multiagent appelé Q-learning par jeu adaptatif. Les résultats expérimentaux engendrés par un tel algorithme montrent à l'évidence qu'il converge vers un équilibre de Nash lorsque les deux joueurs utilisent le même algorithme d'apprentissage. Nous pouvons postuler dès maintenant, bien que nous visons à le faire dans le cadre de nos futurs travaux, que la convergence peut être prouvée formellement, notamment sur la base des travaux de Wang *et al.* (2003) portant sur les jeux stochastiques coopératifs.

Nous avons montré expérimentalement que notre algorithme converge vers l'équilibre de Nash optimal si celui-ci existe et dès lors, il n'a nullement besoin d'être guidé par le concepteur pour y parvenir.

Nous avons ensuite, étudié le comportement de notre algorithme face à un agent non stationnaire. Nos résultats montrent des améliorations attribuées à la qualité des croyances sur les politiques adverses. Toutefois, les améliorations ne sont pas suffisamment significatives pour juger de l'influence des paramètres d'apprentissage. Nous imputons ce manque au fait que les Q-valeurs utilisées pour la règle de décision ont été mises à jour avec des croyances périmées de l'agent sur les stratégies adverses. Ceci nous a alors amené à aborder plus généralement la non stationnarité que nous considérons comme une problématique majeure de l'apprentissage dans les systèmes multiagents.

Remerciements

Nous tenons à remercier tout particulièrement Junling Hu, pour l'aide qu'elle nous a prodiguée, ainsi que les deux évaluateurs pour leurs suggestions fort constructives. Cette recherche est supportée par le conseil de recherche en sciences naturelles et en génie du Canada (CRSNG).

8. Bibliographie

Banerjee B., Peng J., « Countering Deception in Multiagent Reinforcement Learning », *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS-03) Workshop on Trust, Privacy, Deception and Fraud in Agent Societies*, Melbourne, Australia, July, 2003.

Brown G. W., « Iterative Solution of Games by Fictitious Play », *in* T. C. Koopmans (ed.), *Activity Analysis of Production and Allocation*, Wiley, New York, chapter XXIV, 1951.

Claus C., Boutilier C., « The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems », *AAAI/IAAI*, p. 746-752, 1998.

Fudenberg D., Levine D. K., *The Theory of Learning in Games*, The MIT Press, Cambridge, Massachussets, 1998.

Fudenberg D., Tirole J., *Game Theory*, MIT Press, 1994.

Greenwald A., Hall K., « Correlated-Q Learning », *Proceedings of the Twentieth International Conference on Machine Learning*, p. 242-249, 2003.

Hofbauer J., Sandholm W. H., « On the Global Convergence of Stochastic Fictitious Play », *Econometrica*, vol. 70, nř 6, p. 2265-2294, November, 2002. available at http ://ideas.repec.org/a/ecm/emetrp/v70y2002i6p2265-2294.html.

Hu J., Wellman M. P., « Nash Q-learning for general-sum stochastic games », *J. Mach. Learn. Res.*, vol. 4, p. 1039-1069, 2003.

Kaelbling L. P., Littman M. L., Moore A. P., « Reinforcement Learning : A Survey », *Journal of Artificial Intelligence Research*, vol. 4, p. 237-285, 1996.

Littman M. L., « Markov Games as a Framework for Multi-Agent Reinforcement Learning », *Proceedings of the 11th International Conference on Machine Learning (ML-94)*, Morgan Kaufmann, New Brunswick, NJ, p. 157-163, 1994.

Littman M. L., « Friend-or-foe : Q-learning in general-sum stochastic games », *Proceedings of the Eighteenth International Conference on Machine Learning*, p. 322-328, 2001.

Shapley L. S., « Stochastic Games », *Proceedings of the National Academy of Science*, vol. 39, p. 327-332, 1953.

Tesauro G., « Extending Q-Learning to General Adaptive Multi-Agent Systems », *in* S. Thrun, L. Saul, B. Schölkopf (eds), *Advances in Neural Information Processing Systems 16*, MIT Press, Cambridge, MA, 2004.

Vidal J. M., Durfee E. H., « Predicting the Expected Behavior of Agents that Learn About Agents : The CLRI Framework », *Autonomous Agents and Multi-Agent Systems*, vol. 6, nř 1, p. 77-107, 2003.

Wang X., Sandholm T., « Reinforcement Learning to Play an Optimal Nash Equilibrium in Team Markov Games », *in* S. Becker, S. Thrun, K. Obermayer (eds), *Advances in Neural Information Processing Systems 15*, MIT Press, Cambridge, MA, p. 1571-1578, 2003.

Watkins C. J., Dayan P., « Q-learning », *Machine Learning*, vol. 8, nř 3/4, p. 279-292, 1992.

Weinberg M., Rosenschein J. S., « Best-Response Multiagent Learning in Non-Stationary Environments », *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-04)*, Columbia University, New York City, July, 2004.

Young H. P., *Individual Strategy and Social Structure : An Evolutionary Theory of Instituions*, Princeton University Press, Princeton, New Jersey, 1998.