

À quoi sert la spécialisation en évolution culturelle de la connaissance?

Andreas Kalaitzakis^a
andreas.kalaitzakis@inria.fr

Jérôme Euzenat^a
jerome.euzenat@inria.fr

^aUniv. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, F-38000 Grenoble

Résumé

Des agents peuvent faire évoluer leurs ontologies en accomplissant conjointement une tâche. Nous considérons un ensemble de tâches dont chaque agent ne considère qu'une partie. Nous supposons que moins un agent considère de tâches, plus la précision de sa meilleure tâche sera élevée. Pour le vérifier, nous simulons différentes populations considérant un nombre de tâches croissant. De manière contre-intuitive, l'hypothèse n'est pas vérifiée. D'une part, lorsque les agents ont une mémoire illimitée, plus un agent considère de tâches, plus il est précis. D'autre part, lorsque les agents ont une mémoire limitée, les objectifs de maximiser la précision de leur meilleures tâches et de s'accorder entre eux sont mutuellement exclusifs. Lorsque les sociétés favorisent la spécialisation, les agents n'améliorent pas leur précision. Cependant, ces agents décideront plus souvent en fonction de leurs meilleures tâches, améliorant ainsi la performance de leur société.

Mots-clés : Évolution culturelle, Spécialisation

Abstract

Agents can evolve their ontologies by jointly performing a task. We consider a set of tasks of which each agent considers only a part. We assume that the fewer tasks an agent considers, the higher the accuracy of its best task. To verify this, we simulate different populations considering an increasing number of tasks. Counter-intuitively, the assumption is not verified. On the one hand, when agents have an unlimited memory, the more tasks an agent considers, the more accurate it is. On the other hand, when agents have limited memory, the goals of maximizing the accuracy of their best tasks and agreeing with each other are mutually exclusive. When societies favor specialization, agents do not improve their accuracy. However, these agents will decide more often based on their best tasks, thus improving the performance of their society.

Keywords: Cultural evolution, specialization

1 Introduction

En communiquant entre eux, des agents font évoluer leurs représentations des connaissances, un phénomène qui se manifeste par un changement de comportement. Ce phénomène est étudié dans le cadre de l'évolution culturelle des connaissances, où les agents acquièrent et font évoluer leurs connaissances, en fonction de leur perception et du retour qu'ils reçoivent des autres agents. Ici, nous modélisons des agents qui représentent leurs connaissances en utilisant des ontologies. Jusqu'à présent, la plupart des travaux sur l'évolution culturelle des connaissances se concentrent sur des agents qui (1) font évoluer leurs connaissances dans des environnements mono-tâche et (2), disposent de ressources illimitées. [15] propose un jeu d'alignement itératif, permettant aux agents de classer progressivement les nombres dans des ensembles identiques. Ainsi, à la fin de l'expérience, les agents partagent la même représentation des connaissances. Dans [2], les agents interagissent sur une seule tâche de décision. Après un nombre fini d'interactions, les agents se mettent d'accord sur toutes les décisions, mais pas nécessairement sur la même base.

Cependant, notre monde n'est pas monolithique et les ressources des agents ne sont pas illimitées. Les agents doivent être capables d'utiliser des ressources limitées afin d'effectuer plusieurs tâches de manière efficace. Pourtant, on sait peu de choses sur la façon dont la réalisation de plusieurs tâches affecte l'évolution culturelle des connaissances. On en sait encore moins sur les agents encouragés à se spécialiser dans des environnements multitâches. Peuvent-ils se spécialiser tout en restant en accord les uns avec les autres? Les sociétés d'agents bénéficient-elles de la spécialisation des agents? Nous abordons ces questions en étendant le cadre introduit dans [2]. Dans ce cadre, les agents effectuent la même tâche consistant à prendre une décision abstraite dans un domaine abstrait. Dans notre cadre, les agents sont affectés à une ou plusieurs tâches et

sont donc capables de prendre des décisions dans plusieurs domaines abstraits. Notre hypothèse principale est que les agents qui entreprennent moins de tâches atteignent une précision plus élevée sur leurs meilleures tâches, que les agents qui entreprennent toutes les tâches. Nous supposons qu'en interagissant sur un ensemble limité de tâches, les agents seront plus précis dans certaines tâches que dans d'autres.

Nous organisons notre travail en deux expériences. Dans la première expérience [8], des agents disposant d'une mémoire illimitée entreprennent un ensemble limité de tâches, que nous appellerons par la suite le scope. Dans notre travail, la prise de décisions pour chaque tâche repose sur différentes propriétés. Par exemple, décider de prendre un parapluie peut reposer sur la propriété humidité alors que décider de porter un t-shirt peut reposer sur la propriété température. Dans la deuxième expérience [9], les agents entreprennent les mêmes tâches, tout en faisant face à des limitations de mémoire. À cette fin, nous fixons un nombre maximum de classes à maintenir dans l'ontologie d'un agent. Lorsque les agents atteignent ces limites, ils essaient d'oublier des connaissances qui ne sont pas pertinentes pour leur scope et rendent ainsi l'espace à nouveau disponible. Nous évaluons les ontologies des agents par leur contribution à (1) la promotion de l'accord entre les agents et (2) la prise de décisions correctes pour différentes tâches.

Les entités qui constituent l'environnement, ainsi que la notation qui le décrit, sont présentés dans le §3. La section 4 présente les grandes lignes de l'expérience. La section 5 présente un ensemble d'hypothèses ainsi que le protocole utilisé pour les tester. Les résultats des tests sont présentés dans la section 6.

2 État de l'art

Il a été montré que les jeux référentiels [11], facilitent l'établissement de protocoles de communication entre des agents communicants qui tentent d'obtenir une rétribution partagée. [13] soutient qu'un protocole de communication émerge lorsque les agents tentent de minimiser la complexité computationnelle de l'interprétation sémantique. [7] et [3] présentent différents cadres permettant l'émergence d'un langage partagé. [7] étudie un cadre où deux agents développent un langage dans le but de réussir dans un jeu référentiel. [3] propose un cadre permettant la transmission intra-générationnelle, en se concentrant sur la compositionnalité de la langue émergente.

[6] montre que la transmission culturelle implicite conduit à une plus grande compositionnalité de la langue. Notre travail repose également sur les agents qui essaient de se mettre d'accord entre eux. Cependant, les travaux présentés ici se concentrent sur les caractéristiques du protocole de communication émergé. Bien que des caractéristiques similaires soient également présentes dans notre travail, nous ne nous concentrons pas sur le protocole de communication mais plutôt sur l'impact de l'accord des agents sur la réalisation de plusieurs tâches.

Il a été montré que l'apprentissage multitâche améliore considérablement la classification dans différents domaines, par exemple, la robustesse des adversaires [12], la similarité visuelle inter-concept [4], l'apprentissage du phénotype [5]. Les agents ont également été utilisés pour étudier l'impact de l'apprentissage multitâche sur les protocoles de communication émergents. Dans [14], l'apprentissage par renforcement multi-agent coopératif est considéré. La littérature présentée dans ce paragraphe est liée à notre travail car elle considère des agents qui effectuent plusieurs tâches. Cependant, tout changement dans le comportement et les connaissances de ces agents n'est pas le résultat d'une évolution culturelle.

L'évolution culturelle des connaissances a été étudiée dans [2] et [15]. Dans [15], les auteurs cherchent à comprendre comment les concepts sont organisés et comment un comportement collectif peut être établi de manière autonome. Ils montrent qu'il existe un large éventail de conditions sous lesquelles un consensus collectif se produit. Dans [2], une expérience en deux étapes est utilisée où les agents apprennent d'abord un classificateur et interagissent ensuite par paires. Grâce à un mécanisme d'adaptation, il est montré que les agents parviennent à une meilleure connaissance, sans nécessairement aboutir à des ontologies identiques. Nous nous différencions de ceux-ci en introduisant des agents qui font évoluer leurs connaissances par rapport à des tâches multiples.

Enfin, il est à souligner qu'aucun des travaux présentés dans cette section n'impose de limites aux agents.

3 Cadre expérimental

Nous étendons [2] en introduisant des agents qui entreprennent plusieurs tâches. Ici nous fournissons des définitions préliminaires concernant les

agents, les différentes tâches, les objets, ainsi que leur environnement.

3.1 Environnement

Les agents évoluent dans un environnement peuplé d'objets décrits par un ensemble \mathcal{P} de propriétés booléennes. Les objets sont donc décrits par la présence ou l'absence d'une propriété $p \in \mathcal{P}$, dénotée par p et $\neg p$ respectivement. Il n'existe donc que $2^{|\mathcal{P}|}$ types d'objets, qui sont rassemblés dans un ensemble \mathcal{I} .

3.2 Tâches

Le terme tâche fait référence à un travail effectué par un agent. Ici, nous nous concentrons sur un ensemble de tâches de décision : prendre une décision abstraite dans un domaine abstrait, c'est-à-dire choisir parmi $d_1, d_2 \dots d_n$. Il peut y avoir différentes tâches $t \in \mathcal{T}$ associées à un ensemble différent de décisions possibles \mathcal{D}_t . Dans ce contexte, chaque objet o peut être considéré par rapport à toute tâche $t \in \mathcal{T}$. Une fonction $h^*(o, t) \rightarrow \mathcal{D}_t$ fournit la décision correcte, inconnue des agents, pour un objet o par rapport à une tâche t .

3.3 Agents

Les agents sont des entités autonomes, coexistantes, capables de percevoir et de distinguer des objets en fonction de leurs propriétés. Dans ce contexte, une société d'agents multitâches \mathcal{A} , réalise de multiples tâches. À cette fin, les agents construisent et font évoluer des connaissances sous la forme d'ontologies, privées à chaque agent, exprimées en \mathcal{ALC} [1]. Chaque agent α utilise ses connaissances pour calculer une fonction $h^\alpha(o, t) \rightarrow \mathcal{D}_t$ qui, étant donné un objet o et une tâche t , fournit une décision $h^\alpha(o, t)$. La figure 1 montre un exemple de connaissance multi-tâche construite par un agent α . La partie inférieure représente l'ontologie privée \mathcal{O}^α de l'agent α , lui permettant de classifier les objets de l'environnement. La partie supérieure présente deux ontologies de décision, chacune contenant les décisions valides pour les tâches t_1 et t_2 . Étant donné qu'un agent α apprend au plus une décision pour un objet o et une tâche t , chaque feuille de \mathcal{O}^α ne peut être alignée plus d'une fois avec la même ontologie de décision.

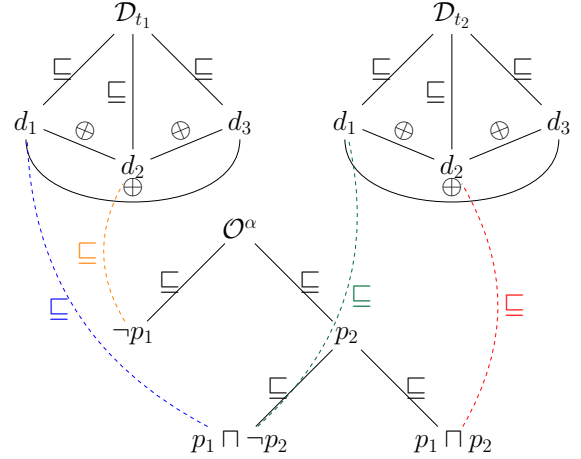


FIGURE (1) Exemple d'une ontologie multitâche pour un agent α . Chaque couleur représente une décision différente. Le symbole \oplus indique que deux décisions sont disjointes.

4 Aperçu de l'expérience

L'expérience est initialisée par une phase d'apprentissage, à la fin de laquelle chaque agent a appris une ontologie privée. Une fois leurs ontologies apprises, les agents passent par un nombre fixe d'interactions. En fonction du résultat d'une interaction, un agent peut adapter son ontologie. La figure 2 illustre le déroulement de l'expérience du point de vue de l'agent α . De plus amples détails sur la façon dont les agents apprennent, définissent leur scope, interagissent, oublient et adaptent leurs ontologies sont présentés dans les sous-sections 4.1, 4.2, 4.3, 4.4 et 4.5 respectivement.

4.1 Apprentissage initial

Nous abordons l'apprentissage multitâche comme un problème d'induction d'une ontologie capable de fournir une décision pour toute tâche $t \in \mathcal{T}$. L'induction d'une telle ontologie peut être vue dans la figure 3. Nous observons qu'il est possible pour un agent d'être capable de classifier un objet mais d'être incapable de prendre une décision concernant une ou plusieurs tâches. Par exemple, étant donné l'ontologie de la figure 1 et un objet décrit par $p_1 \sqcap p_2$, il est impossible de prendre une décision par rapport à la tâche t_1 .

4.2 Définition du scope

Nous implémentons ici des agents qui accomplissent différents sous-ensembles de tâches, en donnant la priorité à certaines tâches par rap-

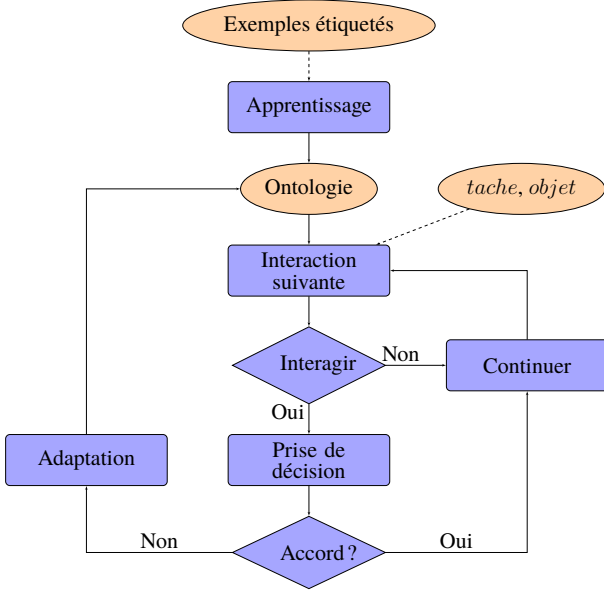


FIGURE (2) Schéma de l'expérience. Les éléments en couleur bisque constituent des entrées pour différentes activités (e.g., les exemples étiquetés). Les éléments en violet constituent soit des activités (rectangles), soit des décisions (diamants).

port à d'autres. À cette fin, nous introduisons la fonction $prior^\alpha(t) \rightarrow \mathcal{N}$, qui renvoie un entier unique attribué à la tâche t par un agent α . Cet entier va de 1 à $|\mathcal{T}|$ et représente la priorité de la tâche t parmi toutes les tâches existantes. Munis de leurs classements de priorité, les agents définissent l'ensemble des tâches qu'ils entreprennent. Afin de décider si une tâche doit être incluse dans leur scope, les agents comparent la priorité de chaque tâche t avec le rang de priorité maximum accepté m . Si $prior^\alpha(t) \leq m$, alors la tâche est dans le scope de l'agent. Si $prior^\alpha(t) > m$, alors la tâche est en dehors du scope de l'agent.

4.3 Interaction

À chaque interaction, deux agents α et β sélectionnés aléatoirement se voient présenter un objet o et une tâche t . Les agents vont alors fournir des décisions basées sur leurs fonctions respectives $h^\alpha(o, t)$ et $h^\beta(o, t)$. Si un des agents n'est pas en mesure de fournir une décision, alors cet agent en choisit une au hasard. Les agents révèlent alors leurs décisions les uns aux autres. Si $h^\alpha(o, t) = h^\beta(o, t)$, les agents sont d'accord et leur interaction se termine avec succès. Dans le cas contraire, on considère qu'il s'agit d'un échec et l'un des deux agents adaptera ses connaissances.

4.4 Limitations et oubli des connaissances

Nos agents sont limités en termes de nombre maximum de classes présentes dans leurs ontologies. Lorsque cette limitation est atteinte, ils essaient d'oublier certaines parties de leurs connaissances afin de libérer de l'espace mémoire en faveur des tâches qu'ils priorisent. Nous supposons que si elle est suffisamment restreinte, une ontologie ne pourra contenir que les propriétés nécessaires pour être précis sur une seule tâche. Par exemple, décidons si l'agent doit prendre un parapluie (d_1) ou non (d_2), en s'appuyant sur la propriété *humidité*. Si l'ontologie de l'agent ne contient pas la propriété *humidité*, alors la probabilité qu'une décision associée soit correcte est de 50%. Nous implémentons le processus d'oubli comme suit. Nous supprimons les nœuds parents qui satisfont aux critères suivants : (a) leurs descendants sont des nœuds feuilles et (b) leurs descendants sont associés à la même décision concernant toutes les tâches dans le scope de l'agent. Le processus est répété de manière récursive, tant que des nœuds parents satisfaisant aux critères (a) et (b) existent.

4.5 Adaptation

Nous étendons le mécanisme d'adaptation présenté dans [2], en introduisant l'adaptation en présence de plusieurs tâches (figure 5). Un agent peut soit remplacer une décision existante, soit diviser une classe en 2 sous-classes. L'agent le fait sur la base d'une propriété qui permet de distinguer l'objet actuel des objets classés par la classe à diviser. Seules les décisions concernant la tâche en cours sont affectées.

Algorithm 1 Adaptation d'ontologie

- 1: α demande à β la définition de la classe classant l'objet o .
- 2: β répond avec \mathcal{C}^β .
- 3: **if** $\mathcal{C}^\alpha \not\sqsubseteq \mathcal{C}^\beta$ **then**
- 4: α demande à β sa décision $h^\beta(o, t)$.
- 5: β répond avec $h^\beta(o, t)$
- 6: une propriété $p : p \in \mathcal{C}^\beta, p \notin \mathcal{C}^\alpha$ est sélectionnée.
- 7: α divise \mathcal{C}^α en $\mathcal{C}_1^\alpha \equiv \mathcal{C}^\alpha \cap \neg p$ et $\mathcal{C}_2^\alpha \equiv \mathcal{C}^\alpha \cap p$
- 8: α associe \mathcal{C}_1^α à toutes les décisions précédemment associées à \mathcal{C}^α .
- 9: **if** Aucune décision concernant t n'est associée à \mathcal{C}^α . **then**
- 10: α associe \mathcal{C}^α à la décision $h^\beta(o, t)$.
- 11: α associe \mathcal{C}_2^α à la décision $h^\beta(o, t)$.
- 12: **else**
- 13: $\mathcal{C}^\alpha = \mathcal{C}^\alpha \cap p$
- 14: Associer \mathcal{C}^α à toutes les décisions précédemment associées à \mathcal{C}^β .

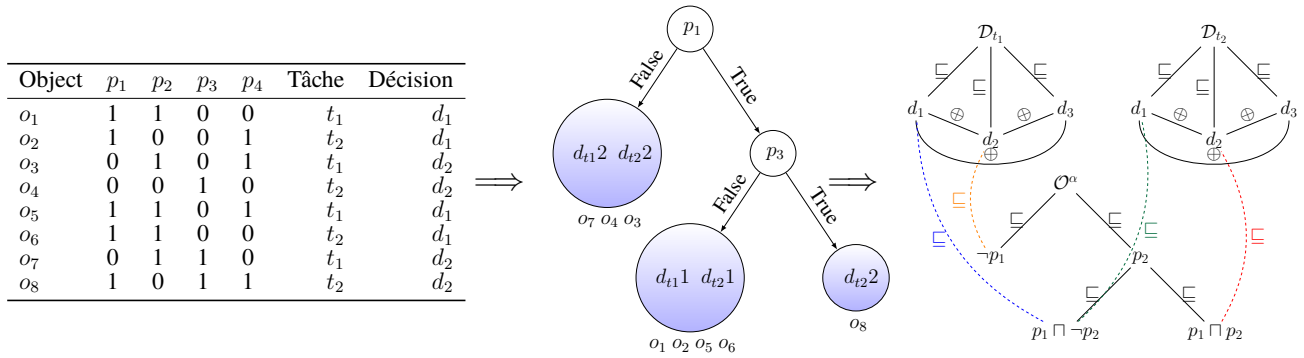


FIGURE (3) Vue d'ensemble de la méthode d'apprentissage proposée. Les agents commencent par induire un arbre de décision basé sur un ensemble d'exemples étiquetés. Cet arbre de décision est ensuite converti en une ontologie.

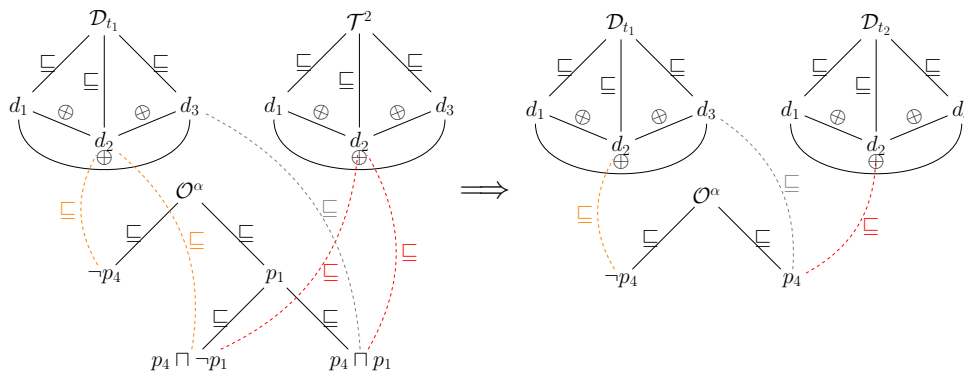


FIGURE (4) Exemple de libération de mémoire par généralisation lors d'une interaction par rapport à la tâche t_2 . Soit la tâche t_2 reposant sur l'ensemble de propriétés \mathcal{P}_{t_2} . La propriété $p_1 \notin \mathcal{P}_{t_2}$, donc p_1 ne permet pas de distinguer différentes décisions pour la tâche t_2 . Dans cet exemple, l'agent a associé la même décision (en rouge), à la fois à $p_4 \sqcap \neg p_1$ et à $p_4 \sqcap p_1$. Ces deux classes peuvent être fusionnées sans aucune perte de connaissance par rapport à t_2 , comme on le voit à droite.

5 Protocole

5.1 Hypothèses

Nous testons les hypothèses suivantes :

- **Hypothèse 1** *Les agents convergent sur leurs décisions après un nombre fini d'interactions.*
- **Hypothèse 2** *Les agents qui entreprennent moins de tâches atteindront une précision plus élevée sur leur meilleurs tâches, que les agents qui entreprennent toutes les tâches.*

5.2 Paramètres

Nous réalisons deux expériences. Dans l'expérience 1, les 18 agents ne sont soumis à aucune limitation de mémoire. Dans l'expérience 2, les ontologies des 18 agents sont limitées à un maximum de 4 classes feuilles. Chaque expérience est exécutée sous 3 configurations, correspondant à des agents qui entreprennent 1 à 3 tâches respectivement. Chaque configuration est exécutée 20 fois et la moyenne de ses résultats est calculée.

Une exécution consiste en 80000 interactions, chaque interaction ayant lieu entre deux agents choisis au hasard parmi 18 agents. Leur environnement contient 64 types d'objets différents, chacun étant perceptible à travers 6 propriétés. Les agents sont initialement formés à toutes les tâches $|\mathcal{T}| = \{3\}$, chaque tâche faisant appel à 2 des 6 propriétés. L'induction de l'ontologie initiale est basée sur un échantillon aléatoire de 10 % de tous les exemples étiquetés existants. Pour chaque tâche, il existe 4 décisions différentes. L'évaluation du score entre deux interactions consécutives se base sur les 60 % de tous les exemples étiquetés existants.

5.3 Mesures

Le taux de réussite (*srate*), tel qu'introduit dans [2] évalue l'interopérabilité entre les agents. Il est défini comme la proportion d'interactions réussies, sur toutes les interactions réalisées jusqu'à la $n^{\text{ème}}$ interaction.

Le taux de décisions correctes (*cdrate*) évalue la performance d'une société d'agents. Il est défini comme la proportion de décisions correctes

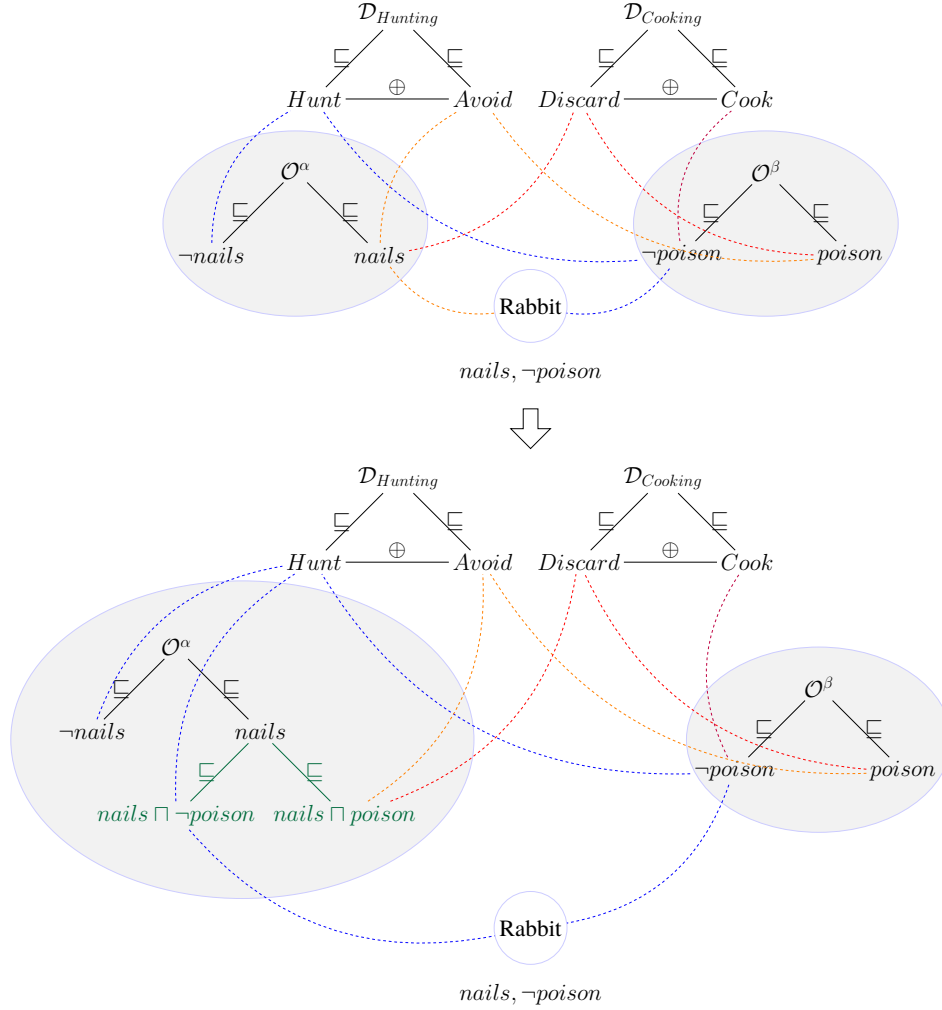


FIGURE (5) Suite à un échec de communication, l'agent α divise la classe *nails* en 2 sous-classes en effectuant une division supplémentaire sur la propriété *poison*. La 1ère sous-classe, $nails \sqcap poison$ sera associée à la même décision que celle de l'agent β (*Hunt*). La 2ème décision sera associée à la décision précédemment associée à la classe *nails* (*Avoid*).

prises par une société d'agents, sur l'ensemble des interactions réalisées jusqu'à la $n^{\text{ème}}$ interaction. Lorsque les agents sont d'accord, cette décision est celle prise par les deux agents. Lorsque les agents sont en désaccord, la décision prise en compte est celle de l'agent ayant le score le plus élevé parmi les deux agents en interaction.

La précision de la tâche (*tacc*) évalue la qualité de la représentation des connaissances d'un agent. Elle adapte la mesure de précision introduite dans [2] à différentes tâches. Elle est définie comme la proportion de types d'objets pour lesquels une décision correcte serait prise par rapport à une tâche t , par un agent α à la n^{e} itération de l'expérience.

$$tacc(\alpha, n, t) = \frac{|\{o \in \mathcal{I} : h_n^\alpha(o, t) = h^*(o, t)\}|}{|\mathcal{I}|}$$

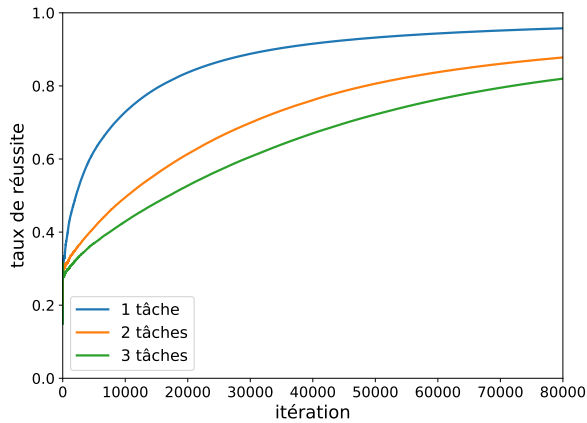
tacc est utilisé pour mesurer la précision moyenne sur la meilleure tâche des agents, ainsi

que leur précision moyenne sur toutes les tâches existantes ou entreprises.

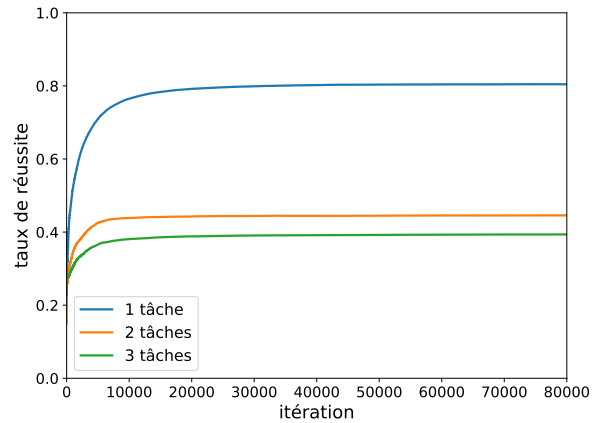
6 Résultats et discussion

Nous présentons et discutons ici les résultats obtenus dans le cadre des deux expériences décrites précédemment. Chacune des figures 6, 7 et 9 est divisée en deux sous-figures. La sous-figure (a) correspond aux résultats acquis lors de l'expérience 1, tandis que la sous-figure (b) correspond aux résultats acquis lors de l'expérience 2. Puisque les agents qui ne font face à aucune limitation ne bénéficient pas de la spécialisation, les autres figures concernent uniquement les résultats de l'expérience 2.

Interopérabilité inter-agent. La figure 6 montre l'évolution du taux de réussite moyen pour différentes tailles du scope. La figure 6a



(a) Agents avec une mémoire illimitée



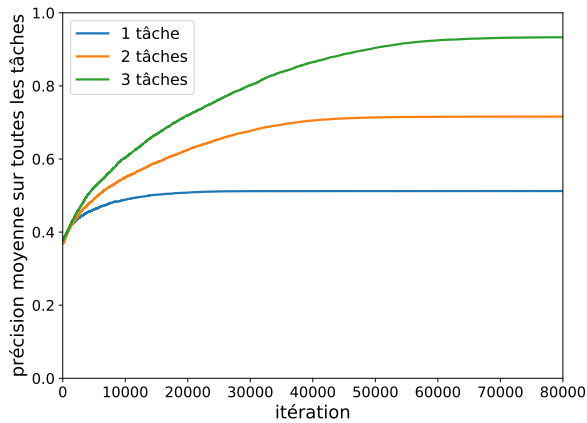
(b) Agents dont la mémoire est limitée à 4 classes

FIGURE (6) Évolution du taux de réussite moyen pour différentes tailles du scope des agents.

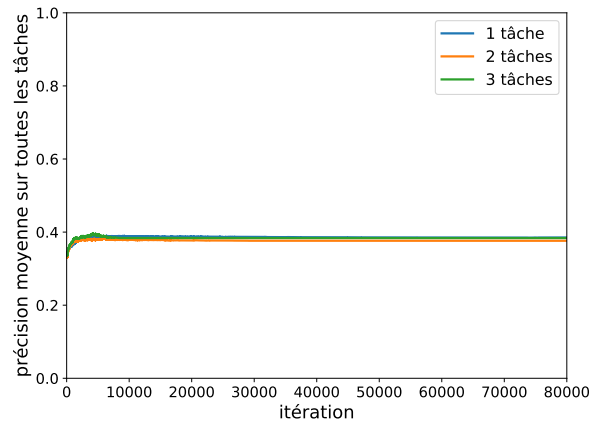
montre qu’une population d’agents en interaction apprendra à se mettre d’accord sur toutes leurs décisions, indépendamment de la taille de leur scope. Ceci est justifié par le fait que les agents représentés ici ne sont confrontés à aucune limitation. Par conséquent, ils adopteront progressivement les mêmes propriétés dans leurs ontologies, même si leurs classes feuilles ne coïncident pas. Nos observations confirment les résultats qui ont été présentés précédemment dans [2]. De plus, elles indiquent que la taille du scope a un impact sur le taux de réussite obtenu. Plus la taille du scope est grande, plus il faut d’interactions pour se mettre d’accord sur tout, donc plus le taux de réussite moyen à la convergence est faible. La figure 6b montre que le taux de réussite se stabilise, sans toutefois converger à 1. Cela indique que soit les agents continuent à adapter leurs ontologies, soit leurs ontologies finales ne leur permettent pas de s’accorder sur toutes les décisions. Dans la section 4.4, nous avons supposé que si elle est suffisamment restreinte, une ontologie sera capable de contenir uniquement les propriétés requises pour être précis sur une seule tâche. Par conséquent, il faut s’attendre à ce que les agents interagissant sur un nombre de tâches nécessitant plus de mémoire, ne pourront pas se mettre d’accord sur toutes les décisions. Il est montré que cela est vrai même pour les agents qui interagissent sur une seule tâche. Nous ne pouvons donc soutenir l’hypothèse 1 que pour les agents qui ne sont confrontés à aucune limitation. Cependant, nous observons que plus le scope des agents est réduit, plus leur taux de réussite est élevé. En raison de l’algorithme d’induction de l’ontologie initiale, l’ontologie d’un agent qui entreprend la tâche t

peut contenir des propriétés qui n’appartiennent pas à \mathcal{P}_t . Rien ne garantit que les agents parviendront à remplacer ces propriétés par des propriétés appartenant à \mathcal{P}_t . Deux cas distincts peuvent être envisagés. Dans le premier cas, un agent éliminera progressivement toutes les propriétés qui ne sont pas liées à la tâche qu’il entreprend. Cet agent va potentiellement apprendre une ontologie qui lui permettra de prendre des décisions correctes par rapport à tous les types d’objets rencontrés. Dans le second cas, un agent ne parviendra pas à éliminer toutes les propriétés qui ne sont pas liées à la tâche qu’il entreprend. Les agents qui relèvent du deuxième cas remplaceront de manière répétée les propriétés qui appartiennent à \mathcal{P}_t par des propriétés différentes qui appartiennent également à \mathcal{P}_t . Par conséquent, ces agents sont capables de prendre des décisions correctes pour différents sous-ensembles de types d’objets existants à un moment donné.

Précision des ontologies. La figure 7 représente l’évolution de la précision moyenne pour différentes tailles du scope des agents. Dans la section 1, nous supposons que les agents interagissant sur un scope limité de tâches seront plus précis sur certaines tâches que sur d’autres, au détriment de leur précision moyenne. Cependant, nous ne l’observons dans aucune des deux sous-figures de la figure 7. 7a montre que la réalisation de tâches supplémentaires améliore significativement la précision moyenne des agents. 7b montre que la taille du scope des agents a une incidence minimale sur leur précision moyenne. Une fois de plus, cela se justifie par les limitations appliquées. Deux cas peuvent être distingués. Dans le premier cas, un agent devient très



(a) Agents avec une mémoire illimitée



(b) Agents dont la mémoire est limitée à 4 classes

FIGURE (7) Évolution de la précision moyenne pour différentes tailles du scope des agents.

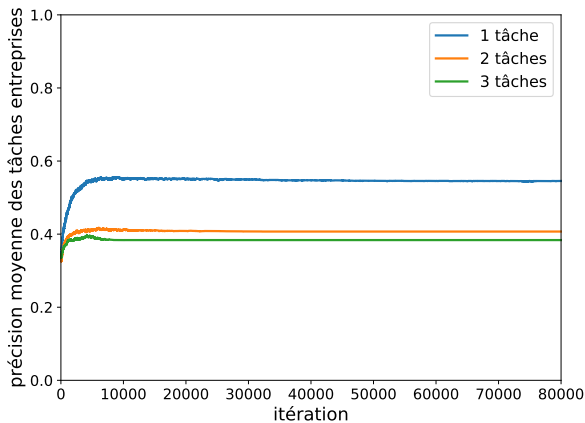
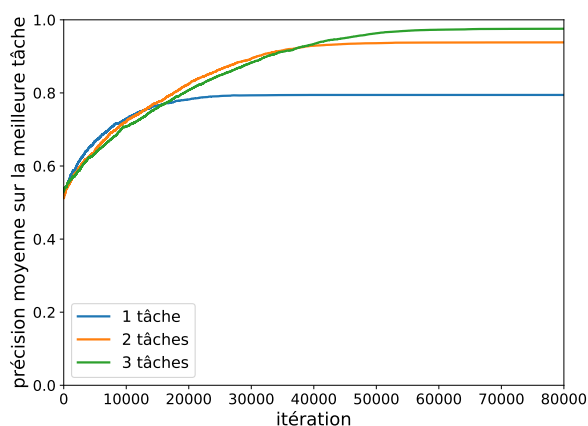


FIGURE (8) Évolution de la précision moyenne des tâches entreprises pour différentes tailles du scope des agents.

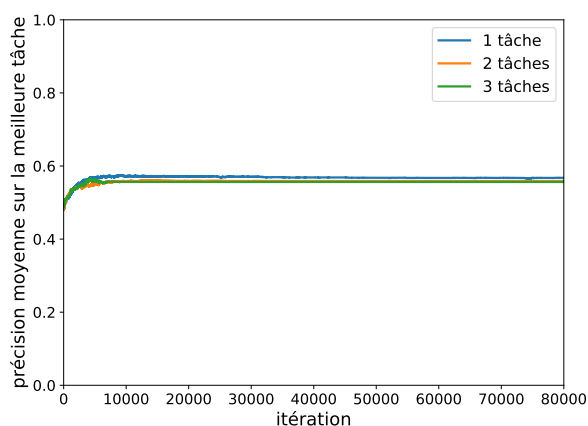
précis dans une tâche et nettement moins précis dans les autres. Dans le second cas, un agent apprend une ontologie lui permettant de devenir moyennement précis sur plusieurs tâches. Cela se traduit par des agents qui démontrent des précisions moyennes qui sont statistiquement indiscernables les unes des autres, indépendamment de la taille de leur scope.

La figure 8 montre l'évolution de la précision moyenne sur les tâches que les agents entreprennent pour différentes tailles de leur scope. Elle montre que plus le scope des agents est petit, plus leur précision moyenne sur les tâches effectuées est élevée. Ce résultat est attendu pour la raison suivante. Un agent dans la configuration examinée peut devenir très précis sur une tâche au maximum. Cependant, plus le scope des agents est grand, plus le nombre de tâches sur la

base desquelles la précision moyenne des tâches effectuées est calculée est élevé. Par conséquent, les agents qui s'attaquent à une seule tâche font preuve d'une précision moyenne sur les tâches effectuées supérieure. La figure 9 dépeint l'évolution de la précision sur la meilleure tâche pour différentes tailles du scope des agents. En examinant 9a, deux observations peuvent être tirées. La première observation est que la précision moyenne des agents à mémoire illimitée (figure 7a) est toujours inférieure à la précision moyenne sur la meilleure tâche des mêmes agents montrés ici. Ces agents sont donc capables de se spécialiser en restreignant le scope de leurs tâches. La deuxième observation est que les agents qui s'attaquent à moins de tâches, ont une précision moyenne sur leur meilleure tâche plus faible que les agents qui s'attaquent à toutes les tâches. Plus précisément, plus le scope des agents est restreint, plus cette précision est faible. Ainsi, nous pouvons conclure que la spécialisation des agents disposant d'une mémoire illimitée n'apporte aucun avantage en termes de précision. Dans la configuration examinée, chaque configuration dépend de propriétés différentes. Par conséquent, notre observation n'est pas liée à la transférabilité des connaissances d'une tâche à l'autre, puisque l'apprentissage de la décision par rapport à une tâche n'est pas lié à l'apprentissage de la décision pour une autre tâche. Au contraire, elle est justifiée par le fait que les agents s'attaquant à toutes les tâches construisent des ontologies plus complètes et par conséquent associent les décisions apprises à une classification plus détaillée. La figure 9b montre que les agents à mémoire limitée se spécialisent également. Cependant, cette spécialisation n'est pas



(a) Agents avec une mémoire illimitée



(b) Agents dont la mémoire est limitée à 4 classes

FIGURE (9) Précision moyenne sur la meilleure tâche en fonction du scope des agents.

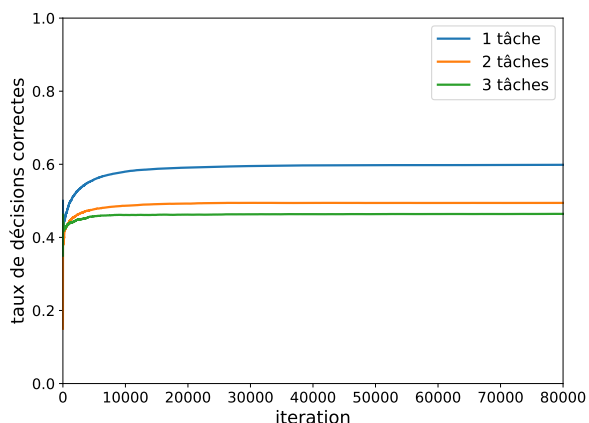


FIGURE (10) Évolution du taux de décision correcte pour différentes tailles du scope des agents.

liée à la taille de leur scope, mais plutôt aux limites de la mémoire. Les agents deviendront très précis au maximum sur une tâche, qu'ils s'attaquent à une ou plusieurs tâches. En d'autres termes, le fait que les agents puissent s'abstenir de toute interaction, sauf celles qui concernent la seule tâche qu'ils entreprennent, ne leur permet pas d'améliorer leur précision sur leur meilleure tâche. Nous avons vérifié que cela est également vrai pour des capacités de mémoire plus élevées (8 et 12 classes respectivement). L'hypothèse 2 est donc soutenue uniquement pour des agents qui ne sont confrontés à aucune limitation.

La figure 10 représente l'évolution du taux moyen de décisions correctes pour différentes tailles du scope des agents. Jusqu'à présent, les résultats montrent que les agents confrontés à des limitations de mémoire (1) spécialisent leurs

connaissances indépendamment de la taille de leur scope et (2) leur précision moyenne ne dépend pas du nombre de tâches qu'ils entreprennent. La figure 10 nous permet d'examiner si les sociétés d'agents bénéficient d'un scope réduit. Les résultats montrent que plus ce scope est réduit, plus le taux de décisions correctes pour une société est élevé. En d'autres termes, si la modification de la taille du scope des agents ne les rend ni moins, ni plus spécialisés, elle permet aux agents de prendre des décisions pour les tâches qu'ils maîtrisent mieux.

Analyse statistique. Notre analyse de variance (*one-way ANOVA*) montre que la taille du scope a un impact statistiquement significatif ($p \leq 0.01$) sur (1) le taux de réussite, (2) la précision moyenne sur les tâches effectuées et (3) le taux moyen de décisions correctes. La taille du scope n'a pas d'impact statistiquement significatif ($p > 0.01$) sur (1) la précision sur leur meilleure tâche et (2), la précision moyenne sur toutes les tâches existantes.

7 Conclusions

Ici nous examinons si les sociétés d'agents bénéficient de la spécialisation. Nous proposons une expérience où les agents font évoluer des ontologies en interagissant sur un ensemble limité de tâches. En exploitant ce cadre, nous montrons que les agents à mémoire limitée spécialiseront leurs connaissances quel que soit le nombre de tâches qu'ils effectuent. Cependant, nos résultats montrent qu'en assignant différentes tâches à différents agents, les sociétés améliorent leur taux de décision correcte. Jusqu'à présent, nous

n'avons examiné que les sociétés homogènes, c'est-à-dire les sociétés où tous les agents effectuent le même nombre de tâches. Ce travail peut être un tremplin vers l'exploration de sociétés d'agents plus complexes. Par exemple, il serait intéressant d'examiner des sociétés d'agents où des agents multitâches coexistent avec des agents qui se concentrent sur des tâches spécifiques.

Remerciements

Ce travail a été partiellement supporté par la chaire MIAI "Knowledge communication and evolution" (ANR-19-P3IA-0003).

Références

- [1] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.
- [2] Yasser Bourahla, Manuel Atencia, and Jérôme Euzenat. Knowledge improvement and diversity under interaction-driven adaptation of learned ontologies. In *Proc. 20th ACM international conference on Autonomous Agents and Multi-Agent Systems (AAMAS), London, United Kingdom*, pages 242–250, 2021.
- [3] Michael Cogswell, Jiasen Lu, Stefan Lee, Devi Parikh, and Dhruv Batra. Emergence of compositional language with deep generational transmission. *CoRR*, abs/1904.09067, 2019. arXiv:1904.09067.
- [4] Jianping Fan, Yuli Gao, and Hangzai Luo. Integrating concept ontology and multitask learning to achieve more effective classifier training for multilevel image annotation. *IEEE Transactions on Image Processing*, 17(3):407–426, 2008.
- [5] Mohamed Ghalwash, Zijun Yao, Prithwish Chakraporty, James Codella, and Daby Sow. Phenotypical ontology driven framework for multi-task learning. In *Proceedings of the Conference on Health, Inference, and Learning, CHIL '21*, page 183–192, New York, USA, 2021.
- [6] Laura Harding Graesser, Kyunghyun Cho, and Douwe Kiela. Emergent linguistic phenomena in multi-agent communication games. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3700–3710, Hong Kong, China, 2019.
- [7] Serhii Havrylov and Ivan Titov. Emergence of language with multi-agent games: Learning to communicate with sequence of symbols. In *5th International Conference on Learning Representations (ICLR 17, workshop track)*, Toulon, France, 2017.
- [8] Andreas Kalaitzakis. 20230110-MTOA experiment description, 2023. URL : <https://sake.re/20230110-MTOA>.
- [9] Andreas Kalaitzakis. 20230120-MTOA experiment description, 2023. URL : <https://sake.re/20230120-MTOA>.
- [10] Lazy lavender. 2020. URL : <https://gitlab.inria.fr/moex/lazylav>.
- [11] David Lewis. Convention : A philosophical study. *Synthese*, 26(1):153–157, 1969.
- [12] Chengzhi Mao, Amogh Gupta, Vikram Nitin, Baishakhi Ray, Shuran Song, Junfeng Yang, and Carl Vondrick. Multitask learning strengthens adversarial robustness. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 158–174, Cham, 2020.
- [13] Luc Steels. What triggers the emergence of grammar? In *AISB'05 : Proceedings of the Second International Symposium on the Emergence and Evolution of Linguistic Communication (EELC'05)*, pages 143–150, Hatfield, United Kingdom, 2005.
- [14] Jonathan Thomas, Raul Santos-Rodriguez, Mihai Anca, and Robert Piechocki. Multilingual agents through multi-headed neural networks. volume 4, Tromsø, Norway, 2023.
- [15] Jun Wang and Les Gasser. Kmutual online ontology alignment. In *Proc. 1st ACM international conference on Autonomous Agents and Multi-Agent Systems (AAMAS), Bologna, Italy*, 2002.