État de l'art de l'apprentissage par renforcement de politiques déclaratives

R Caillière, N Museux

État de l'art de l'apprentissage par renforcement de politiques déclaratives

R. Caillière[1], N. Museux[1]

[1] THALES

romain.cailliere@thalesgroup.com, nicolas.museux@thalesgroup.com

Résumé

L'apprentissage par renforcement explicable/interprétable (XRL) est basé sur plusieurs techniques utilisant différents moyens pour rendre la politique apprise compréhensible par les humains. L'une d'elles consiste à combiner l"apprentissage par renforcement (RL) et les systèmes à base de règles (RBS) afin de créer une politique déclarative, la rendant facilement lisible et compréhensible. Cette question est cruciale lorsqu'il s'agit de certification et de validation dans des cas d'utilisations industrielles critiques. Ce papier propose un état de l'art de ces différentes techniques.

Mots-clés

Politique interprétable, Apprentissage par renforcement, Système à base de règles

Abstract

EXlpainable/Interpretable Reinforcement Learning (XRL) is based on several techniques using different ways to make the learnt policy understandable by humans. One of them is to combine the reinforcement learning (RL) algorithm with Rule-Based Systems (RBS) in order to create a declarative policy, making it easily readable and understandable. This issue is crucial when dealing with certification and validation in industrial use-cases.

Keywords

Interpretable policy, Reinforcement learning, Rule-based system

1 Introduction

L'apprentissage par renforcement est un domaine bien connu de l'intelligence artificielle qui vise, pour un agent, à apprendre comment se comporter dans un environnement. Le RL s'est imposé comme une technologie majeure de l'IA depuis l'émergence du Deep RL. Mais, l'amélioration des performances par l'intégration des réseaux de neurones fait disparaître l'interprétabilité des politiques apprises. Cela rend impossible l'utilisation de telles politiques dans des domaines critiques tels que les soins de santé, l'armée, la justice. Dans les cas d'utilisation industriel critiques l'opérateur doit être en mesure de comprendre les décisions prises par le système. Cela inclut la capacité d'anticiper le comportement de la politique ainsi que la capacité à comprendre le flux de décisions menant à l'erreur. Dans le domaine de la gestion du trafic aérien, l'IA et notamment le Machine Learning et le Reinforcement Learning sont de plus en plus utilisés pour développer des outils d'aide à la décision pour le contrôleurs de trafic aérien. L'utilisation des technologies basées sur l'apprentissage est facilitée par le volume important de données stockées par les fournisseurs de services de navigation aérienne pour des raisons légales. Toutefois, les exigences de sécurité de la navigation aérienne sont tellement strictes que toute les procédures appliquées manuellement, partiellement automatisées ou complètement automatisées, sont basées sur des politiques de décision déclaratives et normalisées au niveau international (ICAO). Si on prend l'exemple d'un outil de détection et de résolution de conflits entre avions, la première étape est de détecter les conflits potentiels avec une approche classique basée sur un modèle de dynamique de l'avion et des considérations géométriques pour trouver les intersections entre trajectoires. Puis, pour proposer une résolution d'un conflit potentiel au contrôleur, le RL est utilisé pour proposer la meilleure classe de décision pour le scénario envisagé : augmenter/réduire la vitesse, changer de cap, changer d'altitude. Il est important pour le contrôleur d'avoir une proposition de résolution qui soit explicable et interprétable même si les résolutions semblent crédibles car le contrôleur doit faire le lien avec les procédures applicable et les politiques de décision déclaratives sous-jacentes. Pour ces raisons, il est primordial d'explorer l'apprentissage de politiques interprétables si l'on veut faire bénéficier les systèmes critiques des avancées du Deep RL.

Quelques études récentes ont présenté l'état de l'art des travaux sur le XRL [41][1][23][17]. Tous soulignent le fait qu'il existe une certaine ambiguïté entre les termes Interprétabilité et Explicabilité. Ils proposent une taxonomie pour clarifier ces termes, que nous avons résumé dans la Figure 1. Si les mots choisis dans la taxonomie ne sont pas les mêmes pour chaque étude, ils ont la même signification. Premièrement, nous trouvons les méthodes qui apprennent directement une politique interprétable (arbre de décision, ensemble de règles). Une deuxième catégorie est celle des méthodes qui sont indirectement interprétables. Dans ce cas, une politique boîte noire est d'abord apprise et, sur la base de cette politique, une politique interprétable est ensuite élaborée (distillation de politiques [45], ap-

prentissage par imitation [26]). Enfin, la troisième catégorie apprend une politique boîte noire sur laquelle certaines procédures/techniques sont appliquées afin de fournir des explications ciblées sur le comportement appris (cartes de saillance [21], explication additive de Shapley [51], génération d'explications textuelles [22] ou explications causales [36]). Elles sont détaillées dans [17].

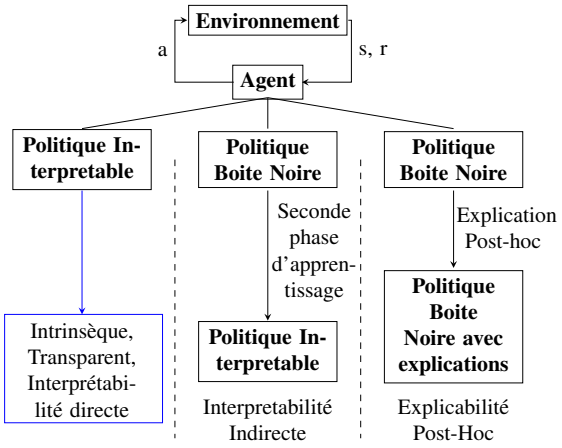

FIGURE 1 – Catégories de l'apprentissage par renforcement interprétable/explicable tirées de [1, 17, 23, 41]

Si ces travaux clarifient bien les notions d'explicabilité et d'interprétabilité ainsi que leurs nuances, elles ont manqué, de notre point de vue, certains travaux dans la catégorie des politiques directement interprétable, et notamment certains travaux sur les politiques à base de règles. Premièrement, [41] n'examine que les articles publiés entre 2010 et 2020. Nous présenterons des travaux antérieurs à 2010. [1] présentent une étude détaillée sur les politiques directement interpretables par arbre de décision, mais omet les travaux sur les politiques basées sur des règles. [23] mentionne les SBR mais oublie tous les travaux étudiant les règles basées sur la logique floue. Enfin, [17], l'étude la plus récente, se concentre sur l'interprétabilité complète. Les auteurs soulignent que non seulement la politique doit être interprétable, mais aussi les intrants et le modèle de transition. Aucune de ces études ne mentionne les travaux réalisés sur l'apprentissage de systèmes de classeurs (LCS) ou les machines de Tsetlin (TM), qui sont deux techniques permettant de créer des politiques à base de règles. En résumé, les contributions de cet article sont : 1- de fournir une étude sur l'apprentissage des politiques fondées sur des règles, qui, à notre connaissance, n'existe pas, 2- de proposer une classification clarifiant la taxonomie de ce champ de recherche et 3- de discuter de l'interprétabilité native revendiquée des systèmes à base de règles. L'organisation de cet article est la suivante : La section 2 introduit les différentes technologies de création de politiques à base de règles. La section 3 présente la littérature.La section 4 discute l'interprétabilité des politiques à base de règles et la section 5 conclut.

2 Présentation des technologies

Dans cette section, nous allons rappeler quelques notions de base des techniques permettant de générer des politiques à base de règles.

2.1 Système à base de règles

Une règle s'écrit sous la forme suivante :

SI Condition ALORS Conclusion

En combinaison avec le RL, la Condition est une prémisse qui est principalement une clause conjonctive de termes, représentant les différentes valeurs des dimensions modélisant l'état de l'environnement. La condition est l'action à entreprendre si la condition est vraie. Les différents termes des prémisses peuvent suivre la logique booléenne (1 ou 0) ou la logique floue (un niveau de véracité $\in [0, 1]$).

Logique Booléenne Dans cette catégorie, nous pouvons trouver des règles de décision où la conclusion est directement appliquée à l'environnement et des règles déductives où la conclusion sera prise comme un terme dans une règle plus générale. Ce type de règles nécessite un modèle de l'environnement avec des dimensions d'état discrètes et des actions discrètes.

Fuzzy Inference System (FIS) Les règles qui utilisent la logique floue sont rassemblées dans un FIS qui opère les étapes suivantes :

- Tout d'abord, la valeur de chaque dimension d'état est « fuzzifiée » à l'aide d'une fonction d'appartenance (FA), qui renvoie le degré d'appartenance (le niveau de véracité) de cette valeur nette relativement aux étiquettes sémantiques des règles.
- Les degrés d'appartenance sont ensuite combinés pour donner la force, ou le degré de vérité, de la règle.
- Enfin, la défuzzification est effectuée en prenant la force de la règle en entrée et renvoie la valeur de l'action en sortie. Trois techniques ont été développées pour la phase de défuzzification (cf. Figure 2). Celle dite Mamdani qui utilise une FA sur laquelle différentes méthodes peuvent être utilisées pour retrouver une valeur à appliquer à partir de la force de la règle. Celle dites de Tsukamoto reposant sur des FA monotones (donc bijectives) qui permet de retrouver directement la valeur à appliquer. Enfin la méthode dite Takagi-Sugeno qui utilise une expression linéaire en fonction de la force de la règle.

L'utilisation des FIS permet de s'attaquer à des environnements aux dimensions continues tout en raisonnant avec des variables sémantiques. Cela en fait un candidat tout désigné pour les problèmes de contrôle.

2.2 Learning Classifier Systems

La technologie des LCS permet de créer une politique à base de règles en se basant sur le triptyque RL-RBS-algorithme génétique. Elle à donc, de notre point de vue, toute sa place dans cette étude. Les règles sont encodées

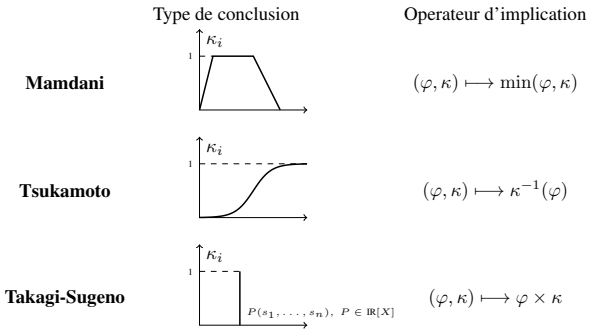

FIGURE 2 – Les trois techniques de défuzzification avec φ la force de la règle et κ la conclusion.

puis manipulées par des opérations génétiques (crossover, mutation). Les LCS fonctionnent selon les étapes suivantes : Premièrement, recevant l'état de l'environnement en entré, la phase de « couverture » s'assure quoi moins une règle se déclenche dans cet état. Si ce n'est pas le cas, une nouvelle règle satisfaisant l'état reçu est créée en se basant sur des paramètres prédéfinis, afin de proposer une action à appliquer. Toutes les règles se déclenchant dans l'état reçu sont rassemblées au sein du *Match Set* [M], puis groupées par action dans un *Prediction Array* [PA] qui calcule l'espérance de gain associé à chaque action. Celle avec la plus grande espérance de gain est choisie pour être appliquée dans l'environnement. Les opérations génétiques sont appliquées sur les règles proposant la meilleure action selon le [PA]. Ensuite, une phase de subsomption et de suppression permettent de généraliser les règles en écartant celles inutiles. Finalement, les paramètres associées aux règles (espérance de gain, erreur de prédiction, fitness) sont mis à jour en fonction de la récompense reçue selon l'équation de Bellman. Ces phases sont répétées jusqu'à la fin de la phase d'apprentissage.

2.3 Tsetlin Machine

Les TM [19] se basent sur les automates de Tsetlin pour la reconnaissance de motifs. Cette technologie se veut basse énergie avec de faibles besoins en mémoire, permettant de générer un modèle de décision interprétable sans perte de précision. Elle est basée sur un mécanisme de mémoire/oubli où les variables (les littéraux) au sein des prémisses (les clauses) ont une note qui varie de 1 (mémorisé) à n (oublié) durant la phase d'apprentissage. Celle-ci se constitue de trois sous phases 1- l'extraction de motif fréquents, 2- la discrimination des littéraux et 3- le choix de l'action à appliquer selon un vote parmi les règles qui se déclenchent. La phase d'extraction de motif est une phase d'apprentissage supervisée réalisant une classification L'objectif pour la TM est de mémoriser les dimensions des entrées utiles/discriminantes. Cette approche vise à reproduire la façon dont les humains se souviennent. Plus on voit quelque chose, plus on le mémorise. A l'inverse, moins on voit quelque chose plus on l'oublie rapidement. Il en va de même pour le motif extrait avec le TM, plus un littéral est vu plus il sera mémorisé et moins il est vu plus il sera

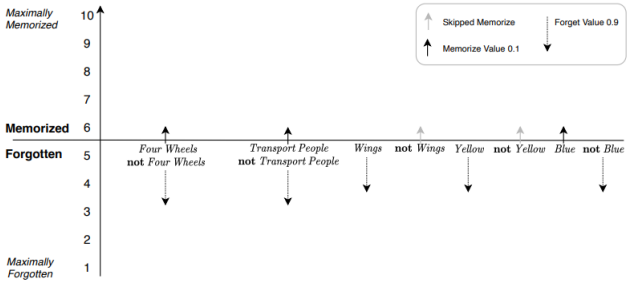

FIGURE 3 – Exemple de mémorisation des littéraux utiles ([20])

oublié. Ce faisant, un motif retenu est constitué des littéraux « *mémorisés* » à la fin de la phase d'apprentissage. Les niveaux de mémorisation sont modélisés de 1 à 10 (cf. figure 3) où 1 à 5 correspond aux littéraux oubliés et 5 à 10 aux littéraux mémorisés. Au début de la phase d'apprentissage, chaque littéral est positionné à 5. Une fois les motifs

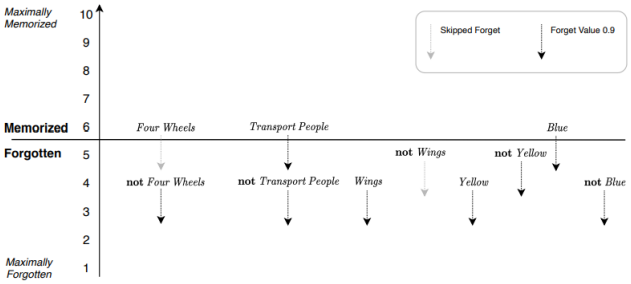

FIGURE 4 – Exemple de mise à jour de la mémorisation des littéraux au sein d'une prémisse ([20])

fréquents identifiés, la première phase de détermination des bonnes prémisses est réalisée. Il faut désormais trouver la meilleure action (ou classe) à leur associer.

La phase de discrimination se déroule lorsque la prémisse est vraie au regard des données d'entrée mais que la conclusion proposée est différetne de celle des données d'entrée. Dans ce cas, la prémisse est trop générale et est vraie alors qu'elle ne devrait pas l'être. Pour corriger ceci, tous les littéraux oubliés (niveau 1 à 5) et faux augmentent leur niveau de mémoire de 1 (cf. Figure 5). En faisant ainsi, certains littéraux discriminants feront partie des prémisses (ou se rapprochera de la mémorisation), ayant pour effet de spécifier la prémisse, évitant ainsi quelle ne se déclenche lorsqu'il ne faut pas. Ces étapes se déroulent jusqu'à la fin de la phase d'entraînement. On notera que les TM sont la seule technologie, à notre connaissance, à prendre en compte nativement les négations de littéraux. Par ailleurs celle-ci est pour le moment à des environnements aux dimensions discrètes ainsi qu'a des conclusions dans le domaine discret.

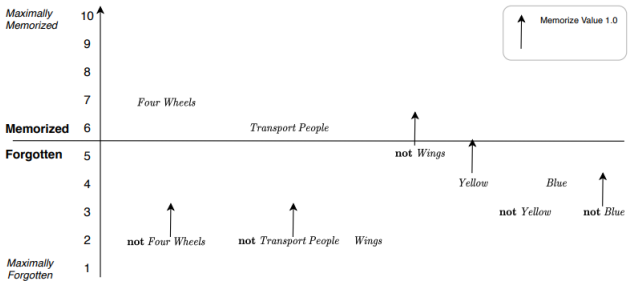

FIGURE 5 – Exemple de spécialisation des prémisses ([20])

3 Apprentissage de politique à base de règles

Cette section expose les différents travaux qui se sont penchés sur la combinaison RL - RBS.

3.1 Les règles de logique standard

On peut considérer le célèbre algorithme Q-Learning [52] comme le premier algorithme de création d'une politique sous forme de règles utilisant le RL. En effet, dans cet algorithme, la politique optimale est déduite de la Q-table, qui fournit pour chaque état l'action offrant le meilleur gain. La politique peut ainsi être lue comme un ensemble de règles.

CLARION [47] est un modèle d'architecture cognitive qui vise à répliquer l'apprentissage humain. Cette architecture repose sur deux niveaux. Un haut niveau qui est composé des savoirs déclaratifs, que l'on peut considérer comme les savoirs théoriques, qui s'appuient sur des règles. Et un bas niveau qui est composé des savoirs procéduraux, les « savoir-faire », qui s'appuient sur le RL.

Une branche spécifique du RL, appelé RL relationnel [13], se concentre sur une description explicite des états et des actions basée sur des prédicats relationnels. La politique apprise prend la forme de règles logiques du premier ordre. Cela génère une politique assez générale et interprétable mais souffre de l'évolutivité limitée de la représentation symbolique. Pour résoudre ce problème, les auteurs de [27] proposent le Neural Logic RL, qui s'appuie sur des machines logiques récurrentes différentiables entraînées avec des méthodes de *policy-gradient*, ouvrant la possibilité à l'agent d'apprendre et de créer ses propres prédicats. La description des états reste « fait main ». Les auteurs de [40] proposent dNL-ILP, qui supprime le besoin d'une représentation relationnelle explicite des états, démontrant des capacités à extraire cette représentation dans des images. Plus récemment, les auteurs de [35] exploitent la capacité du mécanisme d'attention pour remplacer la programmation logique inductive différenciable, et la combine avec le RL symbolique neuronal, pour extraire un ensemble de règles basées sur la logique du premier ordre. Cela atténue le besoin en mémoire et en ressources de calcul, et supprime le besoin d'un expert pour définir des règles ou un modèle de transition.

3.2 Fuzzy Inference System

Les premiers travaux combinant le RL et les FIS ont cherchés à améliorer les contrôleurs, en évitant d'avoir à travailler avec des formules mathématiques complexes, en incorporant des connaissances humaines dans le modèle appris. Ce faisant, la compréhension humaine est facilitée. Cette approche consiste a modifier les règles en se basant sur la récompense reçue. Les premiers travaux se concentrent sur la conclusion des règles. Le but est d'adapter la conclusion (l'action). Cela se fait, soit en créant et modifiant la FA de défuzzification dans le cadre du paradigme Mamdani, soit en pondérant l'action avec la Q-value dans le cadre du paradigme Takagi-Sugeno. Dans le paradigme Actor-Critic [29], le FIS joue le rôle de l'acteur (celui qui choisit l'action *in-fine*). Le travail consiste à affiner les coordonnées des sommets de la FA (triangulaire) utilisée pour la phase de défuzzication. Dans le paradigme Value-based, que ce soit l'algorithme Q-Learning [18][28] ou SARSA [48][11][34], la combinaison se fait avec le processus de défuzzification Takagi-Sugeno. De plus, d'autres travaux basés sur l'Actor-Critic [3] [4] et les méthodes Value-based [12], permettent de peaufiner la structure en deux phases. Une première phase, qui consiste en un apprentissage « offline », définit la structure, suivit d'une seconde phase pour l'affiner. Les travaux suivants ont consisté à ne plus seulement modifier les conclusions des règles mais également la phase de fuzzification en modifiant les paramètres des FA. Ce faisant, les règles floues sont optimisées en fonction de la récompense reçue. En incorporant la *Fuzzy Similarity Measure*[32], le système est capable de déterminer si une nouvelle FA doit être créée ou non. Ainsi, le système est capable d'enrichir les règles prédéfinies en les complétant. On peut également noter que le « critic » est alors remplacé par un *fuzzy predicator* [31] incorporant les règles floues dans le « critic » également. A ce stade, la *malédiction de la dimensionnalité* est toujours un problème dans un environnement complexe. Les auteurs de [50] tirent parti de la similitude du système d'inférence floue et du « Radial Basis Function Network » pour résoudre ce problème, en approximant l'actor et le critic, en même temps. [14] étend considérablement la capacité de cette technologie en permettant de créer une FA pendant l'apprentissage. Cela peut également être utilisé pour compléter un ensemble de règles, prédéfini ou incomplet. Enfin, certaines règles créées peuvent devenir inutiles. La possibilité de les supprimer [25] favorise la création d'une politique plus facilement lisible. Les FIS sont très intéressant de part leur propriété à être des approximateurs universel. Étonnamment, la combinaison n'a pas été étendue avec les dernières avancées du Deep RL, qui permettraient d'aborder des problèmes plus complexes, tout en gardant la propriété d'interprétabilité des FIS.

3.3 Learning Classifier Systems

Les LCS ont été proposé par [24]. Les premiers travaux ont pu traiter des états discrets et des actions discrètes. Ceci montrent des résultats intéressants en classification, tout en ayant la propriété d'être facilement interprétables. Une première amélioration a conduit à baser le classeur (i.e. la

règle) sur la précision (et non la force, initialement) étendant le SI-ALORS original, en ajoutant la précision des règles pour prédire la récompense attendue, devenant :

SI état ALORS action prédit **p**

Ce faisant, les LCS deviennent des *eXtended Classifier Systems* (XCS) [53] [7]. Dans la version *accuracy-based*, les règles sont évaluées en fonction de leur capacité à prédire avec précision la récompense attendue. Cela a permis d'atteindre des performances optimales, avec une grande précision et une forte capacité de généralisation, notamment via l'ajout du mécanisme de *subsomption* dans l'algorithme génétique. D'autre part, certains travaux se sont concentrés sur la limitation due aux états discrets [46]. Ils proposent une représentation par intervalles pour la modélisation de la dimension de l'état d'entrée. La représentation « centre-écart » dans laquelle les centres et les écarts sont codés dans le chromosome, et les représentations « ordonnées » et « non-ordonnées », dans laquelle ce sont les limites inférieure et supérieure de l'intervalle qui sont codées dans le chromosome. Enfin, afin de pouvoir également traiter des actions continues, les LCS floues (FLCS) [5] [9] [6] ont été explorées, remplaçant la logique booléenne par la logique floue. La différence entre LCS et FLCS ne concerne pas l'évolution des règles, mais se produisent aux niveaux (i) du déclenchement de la règle, grâce à l'utilisation de la fonction d'appartenance, (ii) de la manière dont les actions sont choisies et appliquées, et (iii) au niveau de l'inférence floue et de la défuzzification. Il convient de noter que [15] a développé un XCS qui s'appuie sur un principe d'induction pour combiner les règles en remplacement de l'algorithme génétique, supprimant la stochasticité dans leur évolution. Cette technique conserve la bonne capacité de LCS à créer des règles générales et précises.

3.4 Tsetlin Machine

Granmo [19] introduisent les TM pour des problèmes de classification. Ce travail a été étendu pour l'approximation de la fonction valeur [43], dans le but d'attaquer des problèmes RL. Pour ce faire, la *Regression Tsetlin Machine* (RTM) est utilisée afin d'approximer la fonction valeur. Cette adaptation de la classification à la régression implique, dans un premier temps, de collecter des trajectoires d'épisodes pour y travailler ensuite, et de prédéfinir des bornes supérieures et inférieures nécessaires à l'adaptation. Récemment, [42] explore l'utilisation des TM pour les algorithmes de RL dits « On-policy », comme SARSA [44]) et les algorithmes dits « Off-policy" comme l'algorithme *value iteration*, qui utilisent ou non la politique pour la mise à jour des valeurs. Si les TM montrent de bons résultats sur des cas jouets (principalement l'environnement GridWorld), des travaux supplémentaires sont nécessaires pour démontrer sa capacité à traiter des cas d'utilisation plus complexes. Le risque ici étant la limitation de la logique propositionnelle à fournir des règles interprétable.

4 Interpretabilité des RBS

Le Graal dans l'apprentissage d'une politique basée sur des règles est de capturer l'ensemble le plus général de règles qui fournit le comportement optimal. Il y ici a un compromis à faire entre l'obtention d'un ensemble de règles général et l'obtention d'un ensemble de règles performant. Plus il y a de règles, meilleures sont les chances d'avoir la règle spécifique dédiée à une situation particulière. A l'inverse, si les règles sont trop générales, elles risquent de commettre des erreurs en couvrant plus de cas qu'elles ne le devraient. Dans un tel contexte, généraliser signifie compacter l'ensemble de règles, permettre à une prémisse de règles de couvrir plus d'états avec des paramètres moins descriptifs et ce faisant, cela améliore l'interprétabilité de la politique. Au contraire, se spécialiser signifie augmenter la précision de l'ensemble de règles en ciblant précisément la meilleure association état-action. À notre connaissance, il n'existe aucun travail comparant ces technologies en termes d'efficacité et d'interprétabilité des politiques. Si comparer leur efficacité est assez simple en mesurant les indicateurs usuels, il est moins évident de les comparer du point de vue de l'interprétabilité, puisqu'il n'y a pas de consensus sur une définition mathématique de ce qu'est l'interprétabilité [8].

C'est une hypothèse courante de considérer les RBS comme naturellement interprétables. Lipton soutient dans [33] que l'interprétabilité repose sur la transparence du modèle. Il définit la transparence par la capacité du modèle à être (i) **simulable**, représentant la capacité pour un humain de saisir immédiatement le sens du modèle ; (ii) **décomposable**, impliquant que les entrées, les paramètres et le calcul soient facilement appréhensibles et (iii) **algorithmiquement transparent**, fournissant une garantie sur le résultat du modèle sur des données inconnues. Si la transparence algorithmique et la décomposabilité sont des notions bien établies dans les RBS, ce n'est pas le cas de la simulabilité. Trois paramètres principaux des RBS pourraient entraver leur capacité à être simple et facilement saisi par un humain.

Premièrement, le nombre de termes dans la prémisse. Comme le souligne [39], la capacité de l'humain est limitée à traiter sept informations plus ou moins deux, induisant potentiellement qu'une prémisse soit constituée de sept plus ou moins deux termes, au maximum. Le nombre maximum de termes traités dans une prémisse par un humain est probablement dépendant de l'expertise de celui-ci sur le domaine dédié. Afin d'aborder ce problème, [15] souligne que les méthodes doivent montrer de bonnes capacités de généralisation, impliquant la capacité de détecter et de supprimer des variables non discriminantes de l'état.

Deuxièmement, le nombre de règles composant le modèle doit également être maîtrisé. Un nombre élevé de règles diminue la simplicité et la capacité pour un humain à l'appréhender facilement. Plus le comportement optimal est complexe, plus le nombre de règles différentes pour le capturer est élevé. Cela constitue un problème sérieux car il y a un compromis entre avoir toutes les règles nécessaires pour être optimal et garder un nombre raisonnable de règles pour

maintenir la propriété d'interprétabilité. Ici, le RL hiérarchique [16] pourrait aider en définissant différents niveaux de règles [49] permettant de les grouper par *skill*.

Troisièmement, l'enchevêtrement des règles, c'est-à-dire le fait que la conclusion d'une règle est un terme spécifique de la prémisse d'une autre règle, augmente la complexité. Plus il y a de niveaux d'intrication, ou plus la granularité du jeu de règles est élevée, plus il est difficile d'appréhender le modèle. Atténuer cette limite peut passer par la définition d'un hyper-paramètre pour contraindre le niveau de granularité lors de la phase d'apprentissage.

Si Lipton fournit des pistes pour garantir l'interprétabilité d'un modèle, il ne fournit pas de métriques pour le calculer. Certains travaux ont tenté de définir les propriétés d'un ensemble de règles interprétables [38] pour la classification, fournissant quatre métriques calculables qui sont : la prédictivité, la q-stabilité, la simplicité et le score d'interprétabilité. La prédictivité mesure la précision, la q-stabilité mesure la distance entre deux ensembles de règles générés par une approche spécifique, tandis que la simplicité mesure la longueur des règles. Enfin, le score d'interprétabilité est une somme pondérée des ces trois métriques. Ces métriques permettent de mesurer/comparer des jeux de règles déjà établis. Il n'est pas évident de les utiliser lors de la phase d'apprentissage.

L'interprétabilité des FIS a également été étudiée. Plusieurs critères ont été proposés pour quantifier leur lisibilité, à tous les niveaux, y compris les modalités impliquées dans les règles (par exemple la redondance ou la correspondance avec des termes experts), la règle elle-même (le nombre de termes) et l'ensemble de la base de règles (la redondance ou la cohérence) [2] [10] [37]. Ces questions peuvent également être liées à celles envisagées dans le cas des résumés linguistiques flous [30], qui peuvent fournir des pistes pertinentes pour définir des mesures d'interprétabilité.

5 Conclusion

Les domaines industriels critiques nécessitent une politique de décision interprétable pour pouvoir être déployés. Cette étude se concentre sur les politiques basées sur des règles. Une diversité de techniques permet de créer de telles politiques : combinant le RL et les FIS, le RL et les règles logiques booléennes, les LCS et les TM. Il est souvent établi que les RBS sont nativement interprétables, nous discutons ce point et montrons qu'il y a des points d'attention à avoir en tête afin d'avoir cette garantie. L'apprentissage de politiques interprétables par conception est possible en utilisant différentes techniques d'apprentissage capables de traiter des espaces continus et discrets, d'états et d'actions. Cette diversité devrait permettre de traiter une grande variété de problèmes, cependant, des travaux ultérieurs devront augmenter l'évolutivité de la politique basée sur des règles pour s'attaquer à des problèmes plus complexes, tout en conservant ses propriétés d'interprétabilité.

Références

[1] Alnour Alharin, Thanh-Nam Doan, and Mina Sartipi. Reinforcement learning interpretation methods : A survey. *IEEE Access*, 8 :171058–171077, 2020.

[2] J. M. Alonso, A. Ramos-Soto, E. Reiter, and K. van Deemter. An exploratory study on the benefits of using natural language for explaining fuzzy rule-based systems. In *FUZZ-IEEE*, 2017.

[3] Hamid R Berenji and Pratap Khedkar. Learning and tuning fuzzy logic controllers through reinforcements. *IEEE Transactions on neural networks*, 1992.

[4] HR Berenji. An architecture for designing fuzzy controllers using neural networks. *International journal of approximate reasoning*, 6(2) :267–292, 1992.

[5] Andrea Bonarini. An introduction to learning fuzzy classifier systems. In *International Workshop on Learning Classifier Systems*. Springer, 1999.

[6] Andrea Bonarini and Matteo Matteucci. Fixcs : a fuzzy implementation of xcs. In *2007 IEEE International Fuzzy Systems Conference*. IEEE, 2007.

[7] Martin V Butz and Stewart W Wilson. An algorithmic description of xcs. In *International Workshop on Learning Classifier Systems*. Springer, 2000.

[8] Diogo Carvalho, Eduardo Pereira, and Jaime Cardoso. Machine learning interpretability : A survey on methods and metrics. *Electronics*, 8 :832, 07 2019.

[9] Jorge Casillas, Brian Carse, and Larry Bull. Fuzzy-xcs : A michigan genetic fuzzy system. *IEEE Transactions on Fuzzy Systems*, 15(4) :536–550, 2007.

[10] K. Cpałka. *Design of Interpretable Fuzzy Systems*. Studies in Computational Intelligence. 2017.

[11] Vali Derhami, Vahid Johari Majd, and Majid Nili Ahmadabadi. Fuzzy sarsa learning and the proof of existence of its stationary points. *Asian Journal of Control*, 2008.

[12] Sameh F Desouky and Howard M Schwartz. Q (λ)-learning adaptive fuzzy logic controllers for pursuit–evasion differential games. *International Journal of Adaptive Control and Signal Processing*, 2011.

[13] Sašo Džeroski, Luc De Raedt, and Kurt Driessens. Relational reinforcement learning. *Machine learning*, 2001.

[14] Meng Joo Er and Chang Deng. Online tuning of fuzzy inference systems using dynamic fuzzy q-learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(3) :1478–1489, 2004.

[15] Nugroho Fredivianus. *Heuristic-based Genetic Operation in Classifier Systems*. PhD thesis, 2015.

[16] Claire Glanois, Zhaohui Jiang, Xuening Feng, Paul Weng, Matthieu Zimmer, Dong Li, Wulong Liu, and Jianye Hao. Neuro-symbolic hierarchical rule induction. In *International Conference on Machine Learning*. PMLR, 2022.

[17] Claire Glanois, Paul Weng, Matthieu Zimmer, Dong Li, Tianpei Yang, Jianye Hao, and Wulong Liu. A survey on interpretable reinforcement learning. *arXiv preprint arXiv :2112.13112*, 2021.

[18] Pierre Yves Glorennec and Lionel Jouffe. Fuzzy q-learning. In *Proceedings of 6th international fuzzy systems conference*. IEEE, 1997.

[19] Ole-Christoffer Granmo. The tsetlin machine–a game theoretic bandit driven approach to optimal pattern recognition with propositional logic. *arXiv :1804.01508*, 2018.

[20] Ole-Christoffer Granmo. An introduction to tsetlin machines. 2021.

[21] Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. Visualizing and understanding atari agents. In *International Conference on Machine Learning*. PMLR, 2018.

[22] Bradley Hayes and Julie A Shah. Improving robot controller transparency through autonomous policy explanation. In *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*.

[23] Alexandre Heuillet, Fabien Couthouis, and Natalia Díaz-Rodríguez. Explainability in deep reinforcement learning. *Knowledge-Based Systems*, 2021.

[24] John H. Holland. Adaptation. In Robert Rosen and F.M. Snell, editors, *Progress in theoretical biology*. Academic Press, New York, 1976.

[25] Yu Hosoya and Motohide Umano. Dynamic fuzzy q-learning with facility of tuning and removing fuzzy rules. In *2012 IEEE International Conference on Fuzzy Systems*, pages 1–8. IEEE, 2012.

[26] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning : A survey of learning methods. *ACM Computing Surveys (CSUR)*, 2017.

[27] Zhengyao Jiang and Shan Luo. Neural logic reinforcement learning. In *International Conference on Machine Learning*, pages 3110–3119. PMLR, 2019.

[28] Min-Soeng Kim and Ju-Jang Lee. Solving continuous action/state problem in q-learning using extended rule based fuzzy inference system. *Transactions on Control, Automation and Systems Engineering*, 3(3) :170–175, 2001.

[29] C-C Lee and HR Berenji. An intelligent controller based on approximate reasoning and reinforcement learning. In *Proceedings. IEEE International Symposium on Intelligent Control 1989*. IEEE, 1989.

[30] Marie-Jeanne Lesot, Gilles Moyse, and Bernadette Bouchon-Meunier. Interpretability of fuzzy linguistic summaries. *Fuzzy Sets and Systems*, 292, 2016.

[31] Cheng-Jian Lin and Chin-Teng Lin. Reinforcement learning for an art-based fuzzy adaptive learning control network. *IEEE Transactions on Neural Networks*, 1996.

[32] Chin-Teng Lin and CS George Lee. Reinforcement structure/parameter learning for neural-network-based fuzzy logic control systems. *IEEE Transactions on Fuzzy Systems*, 2(1) :46–63, 1994.

[33] Zachary C. Lipton. The mythos of model interpretability, 2017.

[34] Quan Liu, Xiang Mu, Wei Huang, Qiming Fu, and Yonggang Zhang. A sarsa (λ) algorithm based on double-layer fuzzy reasoning. *Mathematical Problems in Engineering*, 2013.

[35] Zhihao Ma, Yuzheng Zhuang, Paul Weng, Hankz Hankui Zhuo, Dong Li, Wulong Liu, and Jianye Hao. Learning symbolic rules for interpretable deep reinforcement learning. *arXiv preprint arXiv :2103.08228*, 2021.

[36] Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. Explainable reinforcement learning through a causal lens. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[37] L. Magdalena. Fuzzy systems interpretability : What, why and how. In *Fuzzy Approaches for Soft Computing and Approximate Reasoning : Theories and Applications*, pages 111–122. Springer, 2020.

[38] Vincent Margot and George Luta. A new method to compare the interpretability of rule-based algorithms. *AI*, 2021.

[39] George A Miller. The magical number seven, plus or minus two : Some limits on our capacity for processing information. *Psychological review*, 1956.

[40] Ali Payani and Faramarz Fekri. Incorporating relational background knowledge into reinforcement learning via differentiable inductive logic programming. *arXiv preprint arXiv :2003.10386*, 2020.

[41] Erika Puiutta and Eric MSP Veith. Explainable reinforcement learning : A survey. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 77–95. Springer, 2020.

[42] Saeed Rahimi Gorji and Ole-Christoffer Granmo. Off-policy and on-policy reinforcement learning with the tsetlin machine. *Applied Intelligence*, pages 1–18, 2023.

[43] Saeed Rahimi Gorji, Ole-Christoffer Granmo, and Marco Wiering. Explainable reinforcement learning with the tsetlin machine. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, 2021.

[44] Gavin A Rummery and Mahesan Niranjan. *On-line Q-learning using connectionist systems*. University of Cambridge, Department of Engineering Cambridge, UK, 1994.

[45] Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. *arXiv preprint arXiv :1511.06295*, 2015.

[46] Christopher Stone and Larry Bull. For real ! xcs with continuous-valued inputs. *Evolutionary Computation*, 11(3) :299–336, 2003.

[47] Ron Sun, Edward Merrill, and Todd Peterson. From implicit skills to explicit knowledge : a bottom-up model of skill learning. *Cognitive Science*, 2001.

[48] L Tokarchuk, J Bigham, and L Cuthbert. Fuzzy sarsa : An approach to fuzzifying sarsa learning. In *Proceedings of the International Conference on Computational Intelligence for Modeling, Control and Automation*, 2004.

[49] José Ramón Trillo, Alberto Fernandez, and Francisco Herrera. Hfer : Promoting explainability in fuzzy systems via hierarchical fuzzy exception rules. In *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2020.

[50] Xue-Song Wang, Yu-Hu Cheng, and Jian-Qiang Yi. A fuzzy actor–critic reinforcement learning network. *Information Sciences*, 177(18) :3764–3781, 2007.

[51] Yuyao Wang, Masayoshi Mase, and Masashi Egi. Attribution-based salience method towards interpretable reinforcement learning. In *AAAI Spring Symposium : Combining Machine Learning with Knowledge Engineering (1)*, 2020.

[52] Christopher John Cornish Hellaby Watkins. *Learning from delayed rewards*. PhD thesis, King's College, Cambridge, United Kingdom, 1989.

[53] Stewart W Wilson. Classifier fitness based on accuracy. *Evolutionary computation*, 3(2) :149–175, 1995.