HAL
open science

Apprentissage adaptatif de comportements éthiques

Rémy Chaput, Olivier Boissier, Mathieu Guillermin, Salima Hassas

To cite this version:

Rémy Chaput, Olivier Boissier, Mathieu Guillermin, Salima Hassas. Apprentissage adaptatif de comportements éthiques. 28e Journées Francophones sur les Systèmes Multi-Agents (JFSMA'2020), Jun 2020, Angers, France. hal-03012127

HAL Id: hal-03012127

https://hal.science/hal-03012127v1

Submitted on 18 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Apprentissage adaptatif de comportements éthiques

R. Chaput[1], O. Boissier[2], M. Guillermin[3], S. Hassas[1]

[1] Univ. Lyon, Université Lyon 1, LIRIS, UMR5205, F-69622, LYON, France
[2] Univ. Lyon, IMT Mines Saint-Étienne, CNRS, Laboratoire Hubert Curien UMR 5516
[3] GEEST, UR Confluence, Sciences et Humanités, Université Catholique de Lyon

{remy.chaput,salima.hassas}@univ-lyon1, Olivier.Boissier@emse.fr, mguillermin@univ-catholyon.fr

Résumé

L'utilisation croissante d'algorithmes d'Intelligence Artificielle (IA) dans des applications impactant des humains requiert de doter ces systèmes d'un comportement pouvant être jugé éthique selon des valeurs humaines. Bien que plusieurs approches existent, la question de l'adaptation au contexte, aux préférences et principes éthiques des utilisateurs, reste posée. Nous proposons de traiter cette question par l'Apprentissage par Renforcement Multi-Agent de tels comportements dans des situations différentes. Nous utilisons des tables de Q-Valeurs et des Cartes Auto-Organisatrices Dynamiques pour permettre l'apprentissage adaptatif de la représentation de l'état de l'environnement, ainsi que des fonctions de récompense pour guider l'éthique du comportement. Cette proposition est évaluée sur un simulateur de répartition d'énergie dans des Smart Grids que nous avons développé. Plusieurs fonctions de récompense visant à déclencher des comportements éthiques sont évaluées. Les résultats montrent la capacité de s'adapter à différentes conditions. En sus des contributions sur le plan de l'adaptation éthique, nous comparons notre modèle à d'autres approches d'apprentissage et montrons de meilleures performances par rapport à une approche d'Apprentissage Profond basée sur le modèle Actor-Critic.

Mots-clés

Éthique, Apprentissage par renforcement, Systèmes Multi-Agent, Répartition de l'énergie.

Abstract

The increase in the use of Artificial Intelligence (AI) algorithms in applications impacting human users and actors has, as a direct consequence, the need for endowing these AI systems with ethical behaviors. While several approaches already exist, the question of adaptability to changes in contexts, users behaviors or preferences still remains open. We propose to tackle this question using Multi-Agent Reinforcement Learning of ethical behavior in different situations using Q-Tables and Dynamic Self-Organizing Maps to allow dynamic learning of the representation of the environment's state and reward functions to prescribe ethical behaviors. To evaluate this proposal, we developed a simulator of intelligent management of energy distribution in Smart Grids, evaluating different rewards functions to trigger ethical behaviors. Results show the ability to adapt to different conditions. Besides contributions on ethical adaptation, we compare our model to other learning approaches and show it performs better than a Deep Learning one (based on Actor-Critic).

Keywords

Ethics, Reinforcement Learning, Multi-Agent Systems, Energy management.

1 Introduction

Les récents progrès dans le domaine de l'Intelligence Artificielle (IA) ont mené à une augmentation rapide de l'utilisation d'algorithmes d'IA dans des applications ayant un impact potentiel sur des acteurs et utilisateurs humains. De tels systèmes incluent, par exemple, le *trading* automatique, la conduite autonome ou assistée, l'allocation de ressources, etc.

Moor [14] affirme que la plupart des applications de nos jours ont des conséquences éthiques, étant donné que leurs actions peuvent causer du tort ou des bénéfices aux humains. Il pointe également le besoin pour ces applications ou agents artificiels d'être conçus avec, ou d'intégrer, des considérations éthiques plus approfondies. Plusieurs approches ont ainsi été proposées (voir section 2). Bien qu'elles soient suffisamment génériques pour gérer une variété de situations, la question de leur capacité à s'adapter à des changements de situations en cours d'exécution reste encore ouverte.

Dans cet article, nous proposons un nouveau système qui apprend dynamiquement des comportements perçus comme éthiques, reflétant des valeurs humaines. Cette approche utilise des Cartes Auto-Organisatrices Dynamiques (*Dynamic Self-Organizing Maps* - DSOMs) pour apprendre la représentation de l'espace d'entrée et discrétiser l'espace d'action, ainsi que des tables de Q-Valeurs (*Q-Tables*) pour mémoriser l'intérêt des actions, avec des fonctions de récompense spécifiquement construites pour guider l'agent vers un comportement éthique ; ces algorithmes permettent l'adaptation quand l'environnement change dynamiquement.

Dans cet article, nous ne ferons pas l'hypothèse que les agents artificiels en eux-mêmes peuvent être considérés comme des "preneurs de décision éthique". Nous utiliserons néanmoins le raccourci "comportement éthique" pour décrire le comportement d'un agent qu'un humain jugerait éthique selon ses valeurs ou principes. Ainsi, notre but est de faire apprendre à ces agents artificiels de tels "comportements éthiques".

Nous présentons dans un premier temps un rapide horizon des approches existantes dans la section 2 et pointons leurs limites quant à l'adaptabilité de leurs comportements éthiques. Dans un second temps, nous présentons en section 3 le modèle éthique et l'algorithme d'apprentissage sur lesquels est basée notre approche. L'évaluation de ce modèle dans le contexte des *Smart Grids* est décrit en section 4. En section 5, nous présentons nos résultats et les comparons à ceux d'autres modèles d'apprentissage. Finalement, nous discutons des avantages et limitations de notre modèle, et présentons des perspectives dans la section 6.

2 État de l'art

Depuis les années 2000, plusieurs groupes de travaux discutent de la capacité d'agents artificiels à montrer un comportement éthique, i.e. qui soit compatible avec des valeurs humaines (e.g. Machine Ethics [3], Machine Morality [17], Moral AI [8]). Dans [10, 9], les relations entre IA et éthique sont présentées autour des niveaux suivants : éthique pour la conception (*ethics for design*), éthique dans la conception (*ethics in design*), éthique par conception (*ethics by design*). Par la suite, nous nous concentrons sur les approches relatives à l'éthique par conception, les deux autres niveaux se rapportant à l'aspect éthique du développement et des conséquences des systèmes artificiels, plutôt qu'au contenu explicitement éthique intégré dans ces systèmes. L'éthique par conception rassemble les travaux visant à intégrer des capacités de raisonnement éthique comme partie intégrante de la production du comportement de l'agent [9]. Ces travaux consistent à résoudre une tâche en prenant en compte des considérations éthiques, e.g. prendre une décision ayant des conséquences sur les humains en considérant un sous-ensemble de valeurs, ou déterminer si une action est acceptable ou non selon un principe éthique. Pour ce faire, ils utilisent différentes techniques d'IA, que nous analysons par rapport à leur capacité d'adaptation face aux environnements dynamiques, en considérant les approches d'éthique implicite d'une part et d'éthique explicite d'autre part [14].

Les agents à éthique implicite s'appuient sur des comportements éthiques pré-codés par les concepteurs, associés à des situations particulières. Ainsi, l'agent ne raisonne pas sur l'éthique et ne peut s'adapter aux changements de situation non prévus par les concepteurs. Nous pouvons citer comme exemple d'un tel système [6], qui décrit un *Ethical Governor* pour des drones létaux autonomes.

Les agents à éthique explicite possèdent des règles générales, peuvent raisonner sur l'éthique et dans la plupart des cas justifier leurs décisions. Ils introduisent l'éthique par raisonnement, qui vise à décider du comportement des agents à partir d'un ensemble de règles explicites, qui peuvent être distinguées entre les règles et valeurs morales (générales, e.g. l'Impératif Catégorique de Kant), et les principes éthiques (plus spécifiques et appliqués, e.g. la transparence des algorithmes), qui évaluent la dimension morale (ou éthique) d'actions dans un contexte donné. Ces approches sont également nommées *Top-Down* par [2]. Elles incluent par exemple le travail de [7] sur le jugement et la confiance dans les Systèmes Multi-Agents, en utilisant le *trading* comme cas d'application, ou le travail de [18] sur l'architecture cognitive LIDA dans le cadre de la délibération morale. De telles approches permettent de s'adapter et raisonner sur la dimension éthique du comportement selon la situation. Toutefois, les règles morales, les valeurs morales et les principes éthiques sont fixés et définis par les concepteurs du système à l'instanciation, ce qui empêche l'adaptation des agents *in situ*.

L'éthique par apprentissage franchit une étape de plus dans l'adaptation en visant à apprendre la mise en relation entre les situations et les règles morales ou principes éthiques. Ces approches sont également nommés *Bottom-Up* par [2] : les règles sont extraites à partir des cas. Cela inclut le travail de [4] sur la conduite autonome, en utilisant des jugements d'experts pour composer un jeu de données de situations éthiques, ou encore le travail de [20], utilisant l'apprentissage par renforcement avec une divergence entre l'agent et un comportement humain normalisé comme récompense éthique. De telles approches peuvent s'adapter si les données de départ sont modifiées, mais l'adaptation durant le fonctionnement n'a pas été explorée. L'apprentissage se fait hors-ligne. Elles souffrent de ce fait des mêmes limitations que l'éthique par raisonnement, par rapport à l'adaptabilité.

Le lecteur intéressé est invité à lire [22] pour une étude plus complète sur ce sujet. Tandis que la plupart de ces travaux considèrent un seul agent, peu adoptent une perspective multi-agent, introduisant par exemple la capacité de traiter des conflits entre les normes sociales et les valeurs de plusieurs acteurs, e.g. le travail de [1] sur les SIPAs (*Socially Intelligent Personal Agents*), ou l'influence de plusieurs agents interagissant dans un environnement partagé. Ce sont d'importantes dimensions à considérer dans les applications à venir, particulièrement du fait que chaque agent doit s'adapter aux autres, à la fois en termes de normes sociales mais aussi d'interactions.

Il pourrait être intéressant d'augmenter l'adaptabilité des agents artificiels aux changements environnementaux en-ligne, ou même aux changements au sein de l'environnement physique et culturel, impactant les différentes sortes de considérations éthiques.

3 Modèle

Afin de surmonter les limitations précédemment mentionnées, nous proposons un nouveau modèle basé sur de l'Apprentissage par Renforcement Multi-Agent.

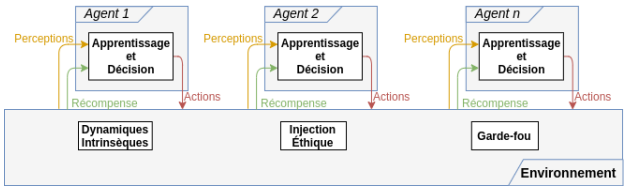

FIGURE 1 – Modèle d'apprentissage de comportements éthiques dans un contexte multi-agent : plusieurs agents interagissent et perçoivent un environnement partagé. Le processus d'apprentissage et de décision de chaque agent s'appuie sur des réceptacles éthiques, qui les rendent capables de traiter les injections éthiques incluses dans la récompense donnée par l'environnement.

3.1 Modèle éthique

Notre modèle éthique est intégré dans un Système Multi-Agent (SMA), dont le but principal est de réaliser une tâche tout en assurant un comportement éthique. Les agents sont composés d'un ensemble de structures de données et algorithmes (groupés dans le cadre "Apprentissage et Décision" dans la figure 1) qui traitent les perceptions pour produire des actions. Ces actions forment un comportement que l'on souhaite éthique.

Pour ce faire, il faut donner une orientation éthique à l'apprentissage. De manière abstraite, on nommera cet apport injection éthique [1]. Cette injection éthique regroupe les considérations éthiques explicites que les concepteurs du système voudraient injecter dans le système, et plus particulièrement dans les agents. Elle peut être située dans l'agent directement (par des connaissances explicites), ou dans l'environnement.

Dans notre cas, notre but étant que les agents apprennent ces considérations et puissent s'adapter aux changements, nous choisissons de la placer dans l'environnement. Elle est rendue accessible aux agents par le biais d'une fonction de récompense, reflétant les valeurs éthiques qui sont considérées importantes par les concepteurs. La récompense est donc calculée par l'environnement puis fournie aux agents ; les agents ignorent donc la façon dont la récompense est calculée, ce qui permet de la modifier si besoin (sans avoir besoin de modifier l'agent).

Le modèle, composé de l'environnement et des agents, et enrichi de l'injection éthique, est également enrichi d'un garde-fou. Il correspond aux limites que les concepteurs ne veulent pas que le système franchisse (e.g. parce qu'elles représentent une impossibilité physique, ou une situation non désirable). De même que pour l'injection éthique, le garde-fou peut être situé dans différents modules : il peut s'agir de contraintes imposées par l'environnement, ou par des agents à travers des interactions avec les autres

1. Il est important de distinguer l'injection éthique des perceptions et actions de l'agent, même si certaines d'entre elles (comme dans notre cas l'équité, voir section 4) possèdent une signification éthique pour le concepteur ou l'humain. Les perceptions et actions ne permettent pas d'orienter le comportement de l'agent vers un comportement éthique. Cette direction est donnée par l'injection éthique, qui représente les considérations éthiques que l'agent doit avoir ou apprendre.

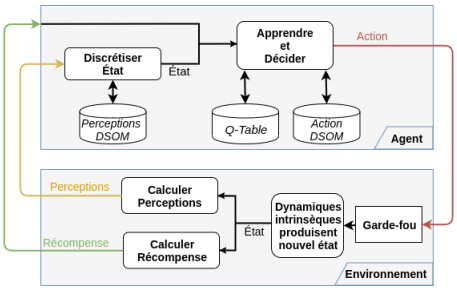

FIGURE 2 – Le cycle d'apprentissage, considérant un unique agent interagissant avec l'environnement.

agents. Selon cette définition, le garde-fou n'influence pas l'apprentissage, et nous le distinguons donc de l'injection éthique, dans le sens où il impose des contraintes, tandis que l'injection éthique guide la décision. Idéalement, les agents devraient apprendre à ne pas franchir ces limites. Dans notre approche actuelle, nous choisissons d'intégrer le garde-fou à l'environnement.

L'utilisation de plusieurs agents permet des cas d'utilisation et des tâches avec aspect éthique plus riches, du fait que les agents peuvent avoir des intérêts divergents, soulevant ainsi des conflits éthiques et éventuellement des dilemmes. De plus, les agents pourraient interagir entre eux, et notamment servir d'injection éthique ou de garde-fou : un agent pourrait émettre un jugement à l'encontre d'un autre agent, fournissant alors une récompense ou bien une contrainte (e.g. en interdisant une action proposée). Pour simplifier dans un premier temps, de telles interactions entre agents ne sont pas étudiées ici.

3.2 Processus d'apprentissage

En suivant le modèle introduit ci-dessus, nous nous intéressons maintenant aux capacités d'apprentissage et de décision des agents, étant donnés les perceptions de l'environnement dans lequel ils sont situés. Ce processus de décision et apprentissage est basé sur l'Apprentissage par Renforcement (*Reinforcement Learning* - RL). L'injection éthique de notre modèle est mise en place dans la génération de récompenses pour le renforcement de la production de comportements éthiques. Les données en entrée de chaque agent sont des propriétés calculées de l'environnement (telle qu'une métrique "sur-consommation"). Comme nous le verrons dans la section suivante, ces métriques sont spécifiquement conçues selon le cas d'application. Les structures et algorithmes inclus dans les agents (voir figure 2) et qui supportent ce processus d'apprentissage sont une table de Q-Valeurs (*Q-Table*) et deux Cartes Auto-Organisatrices Dynamiques (*Dynamic Self-Organizing Maps* - DSOM) : la DSOM de perceptions (P-DSOM) et la DSOM d'actions (A-DSOM). Ils représentent les réceptacles éthiques de par leur capacité à générer les comportements éthiques, telles que la représentation des états, des actions et la politique de choix d'une action dans un état donné.

Formellement, ce système peut être modélisé comme un Processus de Décision Markovien (MDP), dans lequel l'en-

semble des états S consiste en un ensemble d'états multi-dimensionnels et continus représentés par des perceptions de l'environnement.

L'ensemble des actions A est un ensemble où chaque action $a \in A$ est un vecteur de paramètres continus. Nous imposons cette contrainte afin de permettre des environnements et cas applicatifs plus riches, en permettant des actions flexibles (par opposition aux actions discrètes).

Afin de prendre en compte le *Multi-Agent Credit Assignment Problem*[2], la fonction de récompense R est calculée sous la forme de *Difference Rewards* [21] : nous comparons l'état actuel de l'environnement avec un environnement hypothétique dans lequel l'agent n'aurait pas agi. Formellement, considérant une fonction $V : A^n \rightarrow [0, 1]$ qui évalue un ensemble d'actions, la récompense d'un agent i est : $R_i = V(Actions) - V(Actions \backslash \{Actions_i\})$. Intuitivement, l'agent obtient une récompense positive si son action améliore l'état de l'environnement, et négative sinon. Nous rappelons que cette fonction de récompense R est la même pour tous les agents (elle est spécifiée dans l'environnement), mais la récompense est calculée de manière individuelle par l'environnement pour chaque agent. L'environnement hypothétique est donc calculé dans l'environnement ; les agents n'ont pas accès aux actions des autres agents, mais seulement à la récompense calculée R_i.

Afin de manipuler les états et actions, nous utilisons l'algorithme *Q-Learning* [19] augmenté de deux DSOMs [15] (S est appris par la P-DSOM, tandis que A est appris par l'A-DSOM). Les DSOMs sont des grilles 2D de neurones, ayant un vecteur associé (de perceptions ou paramètres d'actions dans notre cas), et sont liées aux états et actions discrets dans la *Q-Table*. Elles sont particulièrement appropriées à notre problème d'adaptation grâce à leur hyper-paramètre d'élasticité η, qui remplace le mécanisme habituel de décroissance du voisinage, afin de permettre à la carte de rester stable quand les données ne changent pas, et de s'adapter quand un changement est détecté. Le seuil de détection de changement est directement déterminé par le coefficient d'élasticité introduit : la fonction de voisinage, paramétrée par η, détermine si le neurone le plus proche de la donnée en entrée est suffisamment proche. Si ce n'est pas le cas, tous les neurones sont mis à jour (la carte s'adapte donc au changement). Cette dynamique de stabilité/adaptation est obtenue en imposant un couplage plus ou moins resserré aux neurones. Cette approche est une extension de [16], où nous remplaçons les Cartes Auto-Organisatrices de Kohonen [11] par les DSOMs, ainsi que la politique d'exploration ϵ-cupide par une politique Boltzmann (ligne 2 de l'algorithme 1 ; le paramètre de température τ contrôle le dilemme exploration-exploitation).

Le processus de décision (voir algorithme 1 et figure 3) utilise la P-DSOM pour discrétiser les perceptions p en une hypothèse d'état i (ligne 1). Cet état est utilisé pour consulter la *Q-Table* et sélectionner un index d'action j selon une distribution de probabilité Boltzmann (ligne 2). Les

2. Chaque agent doit recevoir une récompense proportionnelle à sa contribution dans la solution globale.

Algorithm 1 Partie décision du processus d'apprentissage

Variables partagées : U l'ensemble des neurones de la P-DSOM, W l'ensemble des neurones de l'A-DSOM, et Q la Q-Table
Hyper-paramètres : τ température de Boltzmann
Entrée : p les perceptions
Sortie : a les paramètres d'actions

1: Déterminer le plus proche neurone $i := \arg\min ||p - U_i||$
2: Choisir index d'action j avec probabilité $P(j) := \frac{e^{\frac{Q_{i,j}}{\tau}}}{\sum_k e^{\frac{Q_{i,k}}{\tau}}}$
3: Soient les paramètres d'actions $:= W_j$
4: **for all** dimension k de W_j **do**
5: Ajouter bruit uniforme : $W'_{j,k} := W_{j,k} + rand(-\epsilon, +\epsilon)$
6: **end for**
7: **return** les paramètres d'action perturbés $a := W'_j$

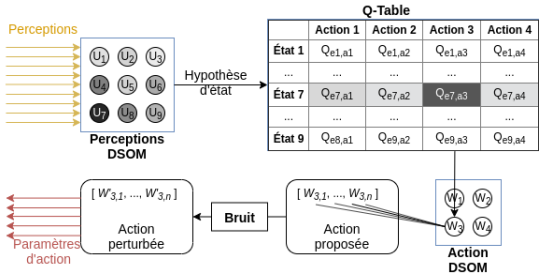

FIGURE 3 – Processus de décision, déterminant l'action à effectuer à partir des perceptions.

paramètres d'action correspondants sont le vecteur associé au neurone j dans l'A-DSOM (ligne 3). Nous ajoutons un bruit uniforme ϵ pour explorer l'espace des actions (lignes 4-6), et le vecteur perturbé est retourné comme action définitive $a \in A$ dans l'environnement.

Dans le processus d'apprentissage (voir algorithme 2), à la suite du processus de décision, les conteneurs éthiques (Q-Table et DSOMs) sont mis à jour selon la récompense r. Si r est meilleure que l'intérêt mémorisé, l'Action DSOM est mise à jour (l'action perturbée a était intéressante ; lignes 4-8). Ensuite, *toutes* les Q-Valeurs sont mises à jour (lignes 9-11), en intégrant le voisinage ψ des Perceptions et Actions DSOMs (lignes 1-2) à l'équation traditionnelle de Bellman. Finalement, la P-DSOM est mise à jour selon les perceptions reçues p (lignes 12-14). Ce cycle de décision-apprentissage-décision peut être vu dans la figure 2.

4 Cas d'application

En association avec notre partenaire industriel, nous appliquons notre modèle au cadre de la distribution d'énergie parmi les utilisateurs d'une micro-grille. Dans ces grilles, la production électrique est décentralisée, au lieu de s'appuyer sur le réseau national. Les *prosumers* (i.e. utilisa-

Algorithm 2 Processus d'apprentissage

Variables partagées : U la P-DSOM, W la A-DSOM, et Q la Q-Table

Fonctions : $c_U(k), c_W(k)$ position du neurone k dans les P-DSOM et A-DSOM

Hyper-paramètres : $\alpha_Q, \alpha_I, \alpha_A$: taux d'apprentissage, η_I, η_A : élasticité, γ : facteur d'actualisation

Entrées : p' nouvelles perceptions, r récompense, p perceptions précédentes, a action retournée par le processus de décision

1: $\forall m \in U \quad \psi_I(m) := e^{-\frac{1}{\eta_I^2}\frac{||c_U(m)-c_U(i)||}{||p-U_i||}}$
2: $\forall n \in W \quad \psi_A(n) := e^{-\frac{1}{\eta_A^2}\frac{||c_W(n)-c_W(j)||}{||a-W_j||}}$
3: Déterminer $i := \arg\min ||p - U_i||$.
4: Déterminer $i' := \arg\min ||p' - U_{i'}||$.
5: **if** $(r + \gamma \max_{j'} Q_{i',j'}) > Q_{i,j}$ **then**
6: **for all** neurone $k \in W$ **do**
7: $W_k := W_k + \alpha_A \, ||a - W_k|| \, \psi_A(k)(a - W_k)$
8: **end for**
9: **end if**
10: **for all** état $m \in U$, action $n \in W$ **do**
11: $Q_{m,n} := Q_{m,n} + \alpha_Q \psi_I(m)\psi_A(n)[r + \gamma \max_{j'} (Q_{i',j'}) - Q_{m,n}]$
12: **end for**
13: **for all** neurone $k \in U$ **do**
14: $U_k := U_k + \alpha_I \, ||p - U_k|| \, \psi_I(k)(p - U_k)$.
15: **end for**

teurs), en plus de consommer, produisent de petites quantités d'énergie (e.g. avec des panneaux photo-voltaïques) dans leur batterie personnelle. Considérant que la production et la demande peuvent fluctuer sur de courtes périodes, ces *prosumers* s'organisent en grilles pour échanger de l'énergie. Ces échanges locaux supposent une forme de coopération pour éviter les situations inégales. De manière similaire, quand la grille est trop sollicitée, les utilisateurs doivent réduire leur consommation (au moins temporairement), et de ce fait réduire leur confort, pour éviter les situations de coupure d'électricité.

Dans ces situations, l'intérêt de chaque *prosumer* peut générer des conflits avec toute la société : bien qu'il veuille maximiser son confort en consommant, il peut engendrer une dégradation des conditions de la grille (e.g. coupures d'électricité) ou un sacrifice de l'intérêt de plusieurs agents pour le bien des autres. Ainsi, ils doivent apprendre individuellement à prendre en compte l'intérêt de la société et adapter leur consommation de manière éthique. Ce cas d'application offre un cadre intéressant avec des considérations éthiques mobilisant des tensions entre des valeurs (e.g. l'équité, le respect des autres, l'écologie).

Suivant le modèle éthique que nous avons décrit en section 3.1, nous avons implémenté un simulateur multi-agent, avec un pas de temps discret, dans lequel l'environnement représente une micro-grille produisant de l'énergie. Chaque agent est un *prosumer* qui implémente le processus de décision effectué par un bâtiment pour consommer et échanger de l'énergie. Plusieurs profils de bâtiment sont considérés : École primaire, Bureaux, Habitations ; chacun ayant un besoin spécifique chaque heure (voir figure 4) et consommant de l'énergie pour satisfaire ce besoin. Leur confort est calculé à partir du besoin et de leur consommation, en utilisant une fonction différente pour chaque profil, ce qui permet par exemple que les écoles aient une priorité supérieure aux habitations (car leur confort sera inférieur pour le même ratio). Ils possèdent également une batterie personnelle.

4.1 Actions

Les paramètres d'actions sont représentés par un vecteur $a = [q, p, s, u, r, z] \in R^6$, tel que q est la quantité d'énergie consommée depuis la micro-grille, p est la quantité d'énergie consommée depuis la batterie personnelle de l'agent, s est la quantité d'énergie que l'agent donne depuis sa batterie à la grille, u est la quantité d'énergie que l'agent stocke depuis la grille dans sa batterie, r est la quantité d'énergie que l'agent achète depuis le réseau national pour le stocker dans sa batterie, et z est la quantité d'énergie que l'agent vend au réseau national depuis sa batterie personnelle.

4.2 Perceptions

Les agents perçoivent l'état de l'environnement à travers plusieurs métriques. Bien que certaines puissent être considérées en référence avec des considérations éthiques (e.g. équité), les perceptions n'orientent pas l'agent vers un comportement éthique et ne font pas nécessairement partie de la récompense, i.e. l'injection éthique influant sur l'apprentissage. Afin de décrire ces métriques, nous introduisons quelques notations : $Conforts$ est l'ensemble des conforts des agents ; $Pris_i = q_i + u_i$ est la quantité d'énergie prise par l'agent i depuis la micro-grille ; $Donne_i = s_i$ est la quantité donnée par l'agent i à la micro-grille ; $Transactions_i = r_i + z_i$ est la quantité d'énergie échangée par l'agent i avec le réseau national.

— *Heure* : L'heure actuelle de l'environnement. Calculée comme $(Temps \mod 24)/24$, ainsi chaque pas de temps représente une heure.
— *Énergie disponible* : La quantité d'énergie initialement disponible dans la micro-grille, notée E.
— *Équité* : Mesure statistique de dispersion de l'ensemble $Conforts$, en utilisant l'index de Hoover. Calculée comme $1 - Hoover(Conforts)$.
— *Énergie perdue* : Énergie disponible qui n'a pas été utilisée (consommée, stockée ou vendue) par les agents. Calculée comme $\min\left(\frac{E+\sum_i Donne_i - \sum_i Pris_i}{\sum_i Pris_i}, 0\right)$.
— *Autonomie* : Absence de transactions avec la grille nationale. Calculée comme $1 - \frac{\sum_i Transactions_i}{\sum_i (q_i+p_i+s_i+u_i+Transactions_i)}$.
— *Bien-être* : La médiane des conforts des agents.
— *Exclusion* : La proportion d'agents dont le confort est inférieur à 50% de la médiane. Calculée comme $\frac{|Conforts_i < 0.5 \times Bien-Etre|}{|Agents|}$.
— *Sur-Consommation* : La proportion d'éner-

gie utilisée par les agents mais qui n'était pas disponible initialement. Calculée comme $\min\left(\frac{\sum_i Pris_i - \sum_i Donne_i - E}{\sum_i Pris_i}, 0\right)$.

Ces métriques sont les mêmes pour chaque agent puisqu'elles décrivent l'environnement ; toutefois, les agents perçoivent également des données additionnelles sur eux-mêmes :

— *Batterie personnelle* : Le ratio entre la quantité d'énergie disponible dans la batterie personnelle de l'agent et la capacité de cette batterie.

— *Confort* : Le confort de l'agent au pas de temps précédent.

— *Bénéfice* : La quantité d'argent que l'agent a gagné (ou perdu) à travers ses transactions avec le réseau national (en vendant ou achetant de l'énergie).

Ces 3 données sont différentes pour chaque agent ; la combinaison des métriques de l'environnement et des données de l'agent produit un vecteur de perceptions $p \in R^{11}$.

4.3 Dynamiques intrinsèques et garde-fou

À chaque pas de temps, une certaine quantité d'énergie est disponible pour les agents, à la fois dans la grille (production globale) et dans leur batterie (production par agent). Les agents décident d'échanger de l'énergie entre la grille, leur batterie, et le réseau national, ce qui produit de nouveaux états de l'environnement. Si les agents n'agissent pas explicitement sur une quantité d'énergie (elle n'est pas consommée, stockée ou vendue), elle est considérée comme perdue. À l'inverse, si un agent consomme plus d'énergie qu'il n'y en a de disponible dans la grille (après les dons des autres agents), sans l'acheter explicitement, cette différence est appelée la sur-consommation. Nous implémentons un simple mécanisme de garde-fou pour prévenir les coupures d'électricité en simulant une transaction dans ce cas, entre la grille locale et le réseau national.

4.4 Injection éthique - Récompenses

Afin d'explorer la capacité de notre modèle à apprendre des comportements éthiques, nous avons conçu plusieurs fonctions de récompense :

— *Équité* : Cette récompense utilise un indicateur de dispersion statistique, nommé index de Hoover, sur les conforts des agents. Elle vise l'équité dans la distribution des conforts, c'est-à-dire qu'aucun agent ne se sacrifie tandis que les autres vivent dans le luxe. $R_{equite,a} = (1 - Hoover(Conforts)) - (1 - Hoover(Conforts \backslash \{Conforts_a\}))$

— *Sur-Consommation (SC)* : Cette récompense vise à réduire la sur-consommation en impactant négativement l'énergie consommée par l'agent mais qui n'était pas disponible dans la grille. $R_{SC,a} = 1 - \frac{\max\left(\sum_i Pris_i - \sum_i Donne_i - E, 0\right)}{\sum_i Pris_i} - \frac{\max\left(\sum_i Pris_i - \sum_i Donne_i - E, 0\right) - Pris_a}{\sum_i Pris_i}$

— *Multi-Objectif-Somme (MOS)* : Cette récompense complexe combine deux sous-récompenses : la sur-consommation et le confort. Cela vise à maximiser le confort tout en minimisant la sur-consommation et évite les solutions triviales que les agents peuvent trouver pour optimiser ces sous-récompenses, e.g. consommer le maximum pour maximiser le confort, sans prendre en compte la sur-consommation, ou à l'inverse ne rien consommer pour minimiser la sur-consommation. Ces solutions échouent à capturer l'attention éthique qui est de satisfaire son confort tout en faisant attention à l'intérêt de la société. $R_{MOS,a} = \frac{1}{2} \times R_{SC,a} + \frac{1}{2} \times Conforts_a$

— *Multi-Objectif-Produit (MOP)* : De manière similaire à Multi-Objectif-Somme, cette récompense combine sur-consommation et confort. $R_{MOP,a} = R_{SC,a} \times Conforts_a$

— *Adaptabilité1 (Ada1)* : Cette récompense change au cours du temps, afin de vérifier l'adaptabilité de notre processus d'apprentissage. Nous apprenons d'abord à minimiser la sur-consommation, puis après un nombre de pas de temps fixe (2000), nous ajoutons l'équité. $R_{Ada1,a} = \begin{cases} R_{SC,a}, & \text{si } etape \leq 2000 \\ \frac{R_{SC,a} + R_{equite,a}}{2}, & \text{sinon} \end{cases}$

— *Adaptabilité2 (Ada2)* : Suivant le même principe, cette récompense change au cours du temps, considérant 3 phases. $R_{Ada2,a} = \begin{cases} R_{SC,a}, & \text{si } etape \leq 2000 \\ \frac{R_{SC,a} + R_{equite,a}}{2}, & \text{si } etape \leq 6000 \\ \frac{R_{SC,a} + R_{equite,a} + Confort_a}{3}, & \text{sinon} \end{cases}$

5 Expérimentations et résultats

5.1 Expérimentations

Nous comparons les performances du modèle originel de Smith avec des SOMs (ci-après "Q-SOM") à notre extension avec des *Dynamic* SOMs (ci-après "Q-DSOM"). Nous considérons également une implémentation par Apprentissage Profond d'une architecture *Actor-Critic*, nommée DDPG [12], ainsi qu'une stratégie aléatoire comme base de référence.

Dans chacune des expérimentations, nous avons considéré un petit nombre de bâtiments : 20 habitations, 5 bureaux et 1 école. Le besoin en énergie de chaque profil a été déterminé en utilisant un jeu de données public de consommation [3]. Trois types de bâtiments ont été sélectionnés dans la même ville (Anchorage) afin de minimiser le risque de biais entre les profils (e.g. pas le même besoin en chaleur) : *Residential*, *Primary School*, *Small Office*. Le profil de consommation horaire a été moyenné sur toute l'année pour chacun (voir figure 4), ce qui donne 24 valeurs de besoin (1 par heure) pour chaque profil d'agent.

Nous avons choisi de rendre disponible dans la micro-grille, à chaque pas de temps, une quantité d'énergie calculée comme une valeur aléatoire entre 80% et 110% de

3. https://openei.org/datasets/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states

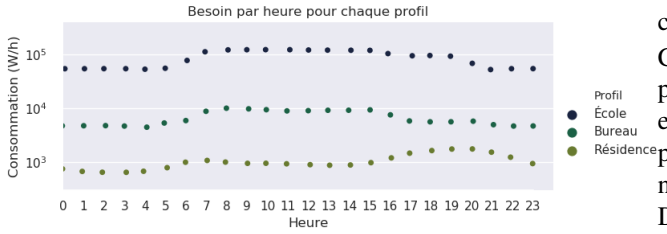

FIGURE 4 – Besoin en énergie pour chaque heure et chaque profil.

Modèle	Paramètre	Valeur
DDPG	Taille de lot	128
	Taux d'apprentissage	0.001
	γ	0.99
	τ	0.001
Q-DSOM et Q-SOM	Dimensions Perceptions SOM	7x7
	Perceptions SOM α	0.8
	Dimensions Action SOM	5x5
	Action SOM α	0.5
	Q-Learning α	0.5
	Q-Learning γ	0.9
	Boltzmann τ	0.5
	Bruit des actions ϵ	0.08
Q-DSOM	Perceptions DSOM η	2.0
	Action DSOM η	2.0
Q-SOM	Perceptions SOM σ	1.0
	Action SOM σ	1.0

TABLE 1 – Valeurs des différents hyper-paramètres pour chacun des 3 modèles. Ces valeurs contrôlent par exemple le dilemme exploitation-exploration, le bruit ajouté aux paramètres d'action, etc. La politique aléatoire ne dispose pas d'hyper-paramètre.

l'énergie totale dont ont besoin les bâtiments. Cet intervalle a été choisi afin de faire apparaître plus de situations de conflits (où il n'y a pas assez d'énergie pour tous) que de situations où les agents peuvent consommer plus que nécessaire.

Les trois modèles (DDPG, Q-SOM, Q-DSOM) utilisent différents hyper-paramètres dont nous relevons les valeurs dans la table 1. Toutes les fonctions de récompense ont été évaluées avec les mêmes valeurs d'hyper-paramètres.

Nous pouvons notamment noter que la carte des perceptions (SOM ou DSOM) a une taille 7x7, ce qui correspond à 49 neurones (états discrets). Ce petit nombre d'états (à comparer au vecteur de perceptions à 11 dimensions, composé de valeurs continues) permet à la table de Q-Valeurs d'apprendre une politique d'action optimale pour chaque état.

5.2 Résultats

Chacun des scénarios a été exécuté 10 fois afin de produire des résultats robustes. La figure 5 montre ces résultats comme diagramme de quartiles.

Comme la figure le montre, Q-SOM et Q-DSOM ont la plupart de temps de meilleures performances que DDPG et aléatoire. La seule exception où DDPG obtient un score presque parfait est la récompense de Sur-Consommation : nous supposons qu'il est plus facile pour l'algorithme DDPG d'atteindre une solution triviale (e.g. consommer 0) grâce à son mécanisme de rétro-propagation de l'erreur, tandis que Q-SOM et Q-DSOM appliquent des bruits aléatoires pour explorer l'espace des actions. Cette hypothèse est soutenue par le fait que DDPG obtient un score inférieur à Q-SOM et Q-DSOM sur les récompenses Multi-Objectif (Somme et Produit). La politique aléatoire obtient toujours un score inférieur à Q-SOM et Q-DSOM, mais meilleur que DDPG sur les récompenses d'équité, Multi-Objectif Somme et Multi-Objectif Produit.

Les modèles Q-SOM et Q-DSOM obtiennent des résultats similaires ; Q-DSOM a un score médian supérieur à celui de Q-SOM sur les fonctions de récompense "Multi-Objectif Somme" et "Adaptabilité 2" (MOS et Ada2), mais Q-SOM a un score médian supérieur sur "Multi-Objectif Produit" et "Adaptabilité 1" (MOP et Ada1). Ces résultats suggèrent un fort impact du choix de la fonction de récompense sur les performances d'apprentissage.

La figure 6 montre l'évolution de la récompense globale (i.e. pour l'ensemble des agents), en utilisant la fonction de récompense *Adaptabilité2* et le modèle Q-DSOM. Pour rappel, cette fonction contient 3 phases, considérant dans chacune des phases un sous-ensemble de valeurs à apprendre (d'abord la sur-consommation, puis sur-consommation et équité, et enfin sur-consommation, équité et confort). Autrement dit, la fonction de récompense qui est utilisée pour guider les agents change selon le temps ; les agents doivent s'adapter à ce changement et apprendre les nouvelles valeurs. Ces phases sont visibles sur la figure (brusques changements dans la récompense reçue). La croissance des récompenses sur cette figure montre l'adaptabilité effective des agents face à de tels changements.

6 Discussion

Premièrement, nous pouvons noter que, alors que nous choisissons les DSOMs, d'autres travaux proposent de remplacer les SOMs par des *Growing Self-Organization Maps* (GSOMs) à la place [13, 5], ce qui peut améliorer la convergence dans certains environnements. En effet, les DSOMs ne garantissent pas la convergence ; considérant 4 états en forme de carré avec seulement 3 neurones, le dernier "rebondira" indéfiniment entre 2 coins. Les GSOMs créent plus de neurones, tandis que les SOMs utilisent le mécanisme de décroissance pour forcer la convergence vers une solution approchée. La création de plus de neurones dans les GSOMs implique une plus grande difficulté à apprendre l'intérêt de chaque état (chaque neurone représentant un état discret, un plus grand nombre d'états requiert au minimum plus de temps), tandis que le mécanisme de décroissance dans les SOMs requiert un ajustement des hyper-paramètres (la décroissance ne doit pas

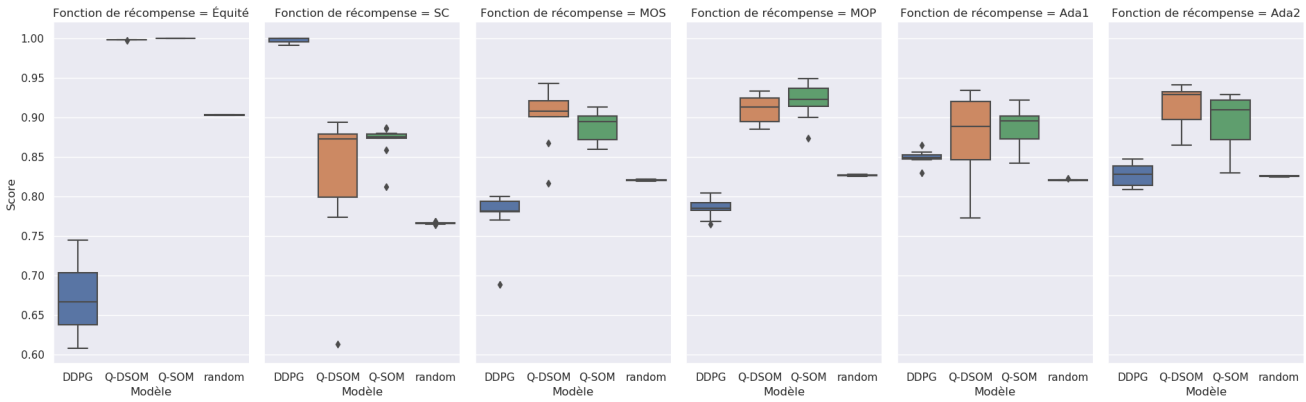

FIGURE 5 – Résultats comparatifs entre les modèles pour chaque fonction de récompense, sur plusieurs simulations. Le score d'une simulation correspond à la moyenne de la récompense globale (calculée pour l'ensemble des agents) sur l'ensemble des pas de temps (10 000). Chacun des 6 sous-graphiques correspond à une fonction de récompense (avec SC pour la Sur-Consommation, MOS pour Multi-Objectif Somme, MOP pour Multi-Objectif Produit, Ada1 et Ada2 pour Adaptabilité 1 et 2).

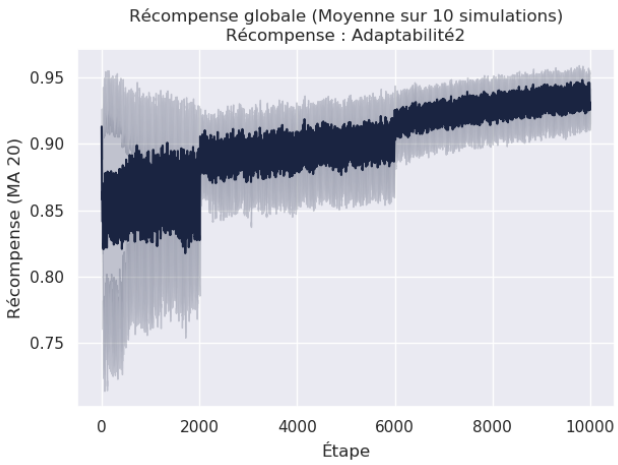

FIGURE 6 – Récompense globale à chaque pas de temps. Les récompenses affichées sont moyennées sur 20 pas de temps (*Moving Average - MA*) et sur les 10 simulations exécutées : la courbe en gris foncé indique la moyenne sur ces 10 simulations, les courbes en gris clair indiquent l'écart-type.

être trop rapide sans quoi l'apprentissage ne sera pas efficace, ni trop lent sans quoi la convergence ne sera pas efficace). Nous pensons que les DSOMs sont de ce fait un choix intéressant, en particulier parce qu'elles permettent d'apprendre la distribution des données au lieu de la densité (permettant ainsi d'apprendre même les états rares).

En sus des vertus relevant de l'éthique par conception, notre choix d'utiliser de multiples agents nous permet d'intégrer plus facilement des vertus relevant de l'éthique dans la conception (*ethics in design*). En effet, il est ainsi plus facile de satisfaire des valeurs éthiques telles que le respect de la vie privée (chaque agent pouvant manipuler directement des données sensibles de l'utilisateur sans les partager avec une instance centralisée), ou encore le respect de l'environnement (avec l'informatique distribuée et le *Green Computing*). Ces deux propriétés sont liées aux implications éthiques de l'utilisation du système plutôt qu'au comportement éthique du système.

Notre approche démontre la capacité d'apprendre l'espace des perceptions et de déterminer l'action optimale pour un comportement éthique. Toutefois, quelques limitations peuvent être soulignées :

— La représentation des différentes actions à travers une correspondance entre les Q-Valeurs et les neurones de l'Action DSOM peut mener à des questionnements (éthiques). Si un certain neurone est déplacé trop loin de sa précédente position, l'action correspondante aurait une signification complètement différente, mais serait toujours en correspondance avec la même Q-Valeur (i.e. le même intérêt), et aurait toujours le même voisinage. En pratique, les mécanismes de mise à jour des DSOMs et de la *Q-Table* modifient également l'intérêt et le voisinage, mais nous n'excluons pas ceci des questions (éthiques) pouvant survenir.

— De plus, l'exploration d'actions multi-dimensionnelles en utilisant un bruit ne semble

pas optimale. En particulier, cela signifie que nous pouvons améliorer l'action proposée sur une certaine dimension, mais complètement dégrader la même action sur une seconde dimension (ce qui annulerait la première amélioration).

— Contrairement à la DSOM d'actions et la *Q-Table*, la DSOM de perceptions n'utilise pas la récompense pour se mettre à jour ; seules les exemples de données en entrée sont utilisés. Toutefois, on pourrait arguer qu'une représentation correcte est nécessaire pour déterminer l'action correcte ; ainsi, une récompense basse pourrait être le signe que la représentation n'est pas utile à l'agent et devrait être apprise différemment.

De plus, les résultats présentés dépendent des expérimentations réalisées ; nous pouvons citer quelques scénarios qui seraient intéressants à expérimenter pour vérifier la robustesse du modèle :

— Le profil de consommation étant moyenné par jour, il est possible que certaines variations (e.g. les saisons) soient effacées. Il serait intéressant de comparer avec une simulation où le profil ne serait pas moyenné, en particulier par rapport au problème bien connu de l'oubli catastrophique.

— L'énergie disponible est toujours entre 80 et 110% de l'énergie totale dont les agents ont besoin. On pourrait imaginer qu'une pénurie d'énergie survienne (e.g. absence soudaine de vent ou de soleil), sur une période plus ou moins longue. Cela permettrait d'analyser la capacité d'adaptation des agents face à un changement dans la dynamique de l'environnement elle-même.

— Nous avons considéré une petite grille d'agents, il serait possible de faire des expérimentations avec un ensemble plus grand afin de déterminer si le nombre d'agents a un impact sur les résultats.

Pour conclure, l'approche proposée ici suggère l'exploration de nouvelles questions (notamment éthiques) :

— Il serait intéressant de se demander si la fonction de récompense, associée aux structures et algorithmes que l'agent implémente, lui permet effectivement d'apprendre le comportement éthique désiré. En d'autres termes, l'agent est-il susceptible d'effectuer une forme de *reward hacking* (i.e. de converger vers des solutions triviales qui maximisent la fonction de récompense mais auxquelles les concepteurs ne s'attendaient pas, e.g. consommer 0 Watts afin que l'équité soit maximale et la sur-consommation nulle).

— Nous pouvons également nous demander si les choix reflétés dans les fonctions de récompense et les comportements éthiques obtenus sont effectivement éthiques et désirés. Cette question nécessiterait une discussion plus large avec des philosophes, des décideurs politiques, et des utilisateurs.

— Nous avons choisi de placer un garde-fou dans l'environnement, ce qui implique que le comportement des agents dépend de cet élément qui leur est externe. Il pourrait être intéressant d'ajouter un garde-fou sous forme d'agent ou bien de l'intégrer directement aux agents.

— La fonction de récompense est actuellement la même pour tous les agents, or les utilisateurs (humains) ont des valeurs différentes (en d'autres termes, n'ont pas les mêmes préférences). Il pourrait être intéressant de permettre de paramétrer chaque agent, d'une part pour l'acceptabilité d'une telle solution par les utilisateurs finaux, et d'autre part pour étudier la capacité d'apprentissage et d'adaptation des agents face à d'autres agents ayant des "valeurs" différentes (à travers la fonction de récompense), et de ce fait un comportement différent.

Remerciements

Ces travaux sont financés par la région Auvergne-Rhône-Alpes (Pack Ambition Recherche) dans le cadre du projet Ethics.AI. Les auteurs remercient leurs partenaires académiques et industriels au sein du projet.

Références

[1] Nirav Ajmeri, Hui Guo, Pradeep K Murukannaiah, and Munindar P Singh. Designing ethical personal agents. *IEEE Internet Computing*, 22(2) :16–22, 2018.

[2] Colin Allen, Iva Smit, and Wendell Wallach. Artificial morality : Top-down, bottom-up, and hybrid approaches. *Ethics and information technology*, 7(3) :149–155, 2005.

[3] Michael Anderson and Susan Leigh Anderson. Guest editors' introduction : Machine ethics. *IEEE Intelligent Systems*, 21(4) :10–11, 2006.

[4] Michael Anderson and Susan Leigh Anderson. Toward ensuring ethical behavior from autonomous systems : a case-supported principle-based paradigm. In *2014 AAAI Fall Symposium Series*, 2014.

[5] Tomoyuki Arai, Yuichiro Toda, Iwasa Mutsumi, Shuai Shao, Ryuta Tonomura, and Naoyuki Kubota. Reinforcement learning based on state space model using growing neural gas for a mobile robot. In *2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)*, pages 1410–1413. IEEE, 2018.

[6] Ronald Arkin. *Governing lethal behavior in autonomous robots*. Chapman and Hall/CRC, 2009.

[7] Nicolas Cointe, Grégory Bonnet, and Olivier Boissier. Ethics-based cooperation in multi-agent systems. In *14th Social Simulation Conference (SSC18), August 2018.*, 2018.

[8] Vincent Conitzer, Walter Sinnott-Armstrong, Jana Schaich Borg, Yuan Deng, and Max Kramer. Moral decision making frameworks for artificial intelligence. In *The Workshops of the The Thirty-First*

AAAI Conference on Artificial Intelligence, Saturday, February 4-9, 2017, San Francisco, California, USA, 2017.

[9] Virginia Dignum. Ethics in artificial intelligence : introduction to the special issue, 2018.

[10] Virginia Dignum. *Responsible Artificial Intelligence : How to Develop and Use AI in a Responsible Way*. Springer International Publishing, 2019.

[11] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9) :1464–1480, Sep. 1990.

[12] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv :1509.02971*, 2015.

[13] Hesam Montazeri, Sajjad Moradi, and Reza Safabakhsh. Continuous state/action reinforcement learning : A growing self-organizing map approach. *Neurocomputing*, 74(7) :1069–1082, 2011.

[14] James Moor. Four kinds of ethical robots. *Philosophy Now*, 72 :12–14, 2009.

[15] Nicolas Rougier and Yann Boniface. Dynamic self-organising map. *Neurocomputing*, 74(11) :1840–1847, 2011.

[16] Andrew James Smith. Applications of the self-organising map to reinforcement learning. *Neural networks*, 15(8-9) :1107–1124, 2002.

[17] Wendell Wallach, Colin Allen, and Iva Smit. Machine morality : bottom-up and top-down approaches for modelling human moral faculties. *Ai & Society*, 22(4) :565–582, 2008.

[18] Wendell Wallach, Stan Franklin, and Colin Allen. A conceptual and computational model of moral decision making in human and artificial agents. *Topics in cognitive science*, 2(3) :454–485, 2010.

[19] Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989.

[20] Yueh-Hua Wu and Shou-De Lin. A low-cost ethics shaping approach for designing reinforcement learning agents. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[21] Logan Yliniemi and Kagan Tumer. Multi-objective multiagent credit assignment through difference rewards in reinforcement learning. In *Asia-Pacific Conference on Simulated Evolution and Learning*, pages 407–418. Springer, 2014.

[22] Han Yu, Zhiqi Shen, Chunyan Miao, Cyril Leung, Victor R. Lesser, and Qiang Yang. Building ethics into artificial intelligence. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 5527–5533, 2018.