

# LAOP ET LASP : PLATEFORMES DE DÉVELOPPEMENT ET DE PARTAGE D'ALGORITHMES D'APPRENTISSAGE PROFOND

EDITION 2018-2019  
Prix étudiants de l'ARC

**CLÉMENT BISAILLON**  
et **LÉONARD OEST O'LEARY**  
Étudiants en sciences informatiques et mathématiques  
Collège de Maisonneuve  
Sous la supervision de  
**Jihene Rezgui**,  
enseignante d'informatique et chercheuse,  
responsable du Laboratoire de recherche informatique Maisonneuve

Communication affichée présentée dans le cadre du colloque *Pour que la formation de la relève scientifique soit sur toutes les lèvres* de l'Association pour la recherche au collégial, 87<sup>e</sup> Congrès de l'Acfas, Université du Québec en Outaouais, Gatineau, le 28 mai 2019

**résumé** Depuis quelques années, les chercheuses et chercheurs en intelligence artificielle se servent d'algorithmes d'apprentissage profond qui s'inspirent du cerveau humain pour résoudre des tâches auparavant impossibles à réaliser avec l'ordinateur.

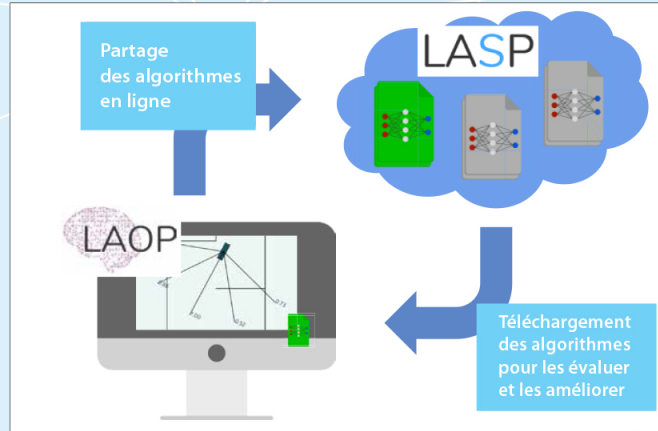
En développant ce type d'algorithme, nous avons constaté le manque d'outils informatiques disponibles pour élaborer, comparer et partager efficacement les nombreux algorithmes d'apprentissage profond. Nous avons mis au point deux plateformes

permettant de résoudre ces problèmes. Afin de démontrer leur utilité, nous les avons utilisées pour développer deux algorithmes – Fully Connected Neural Network (FUCONN) et Mimicking Human Behavior (MHUB) – et trouver le plus performant.

## 1 les plateformes LAOP et LASP

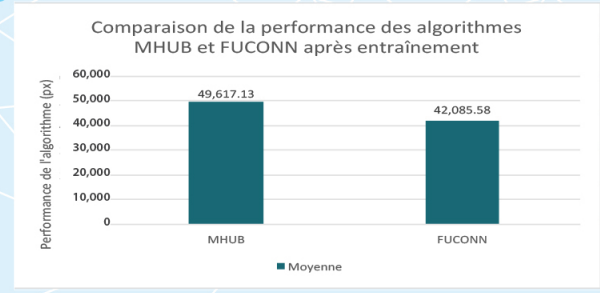
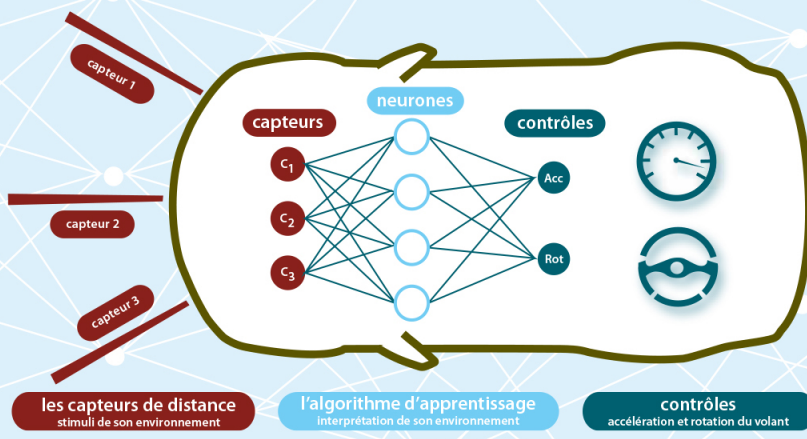
## 2 l'environnement

## 3 résultats



Nous avons développé deux plateformes au sein du Laboratoire de recherche informatique Maisonneuve, dirigé par Jihene Rezgui, Ph. D.: Learning Algorithm Optimization Platform (LAOP) et Learning Algorithm Sharing Platform (LASP). La première est un programme qui permet le développement, l'analyse et la comparaison d'algorithmes d'apprentissage profond. La deuxième est une plateforme en ligne au moyen de laquelle les chercheuses et chercheurs peuvent facilement partager les algorithmes développés à l'aide de LAOP. Ces algorithmes peuvent alors être téléchargés pour être comparés dans LAOP. Tout ce cycle accélère le développement et la conception d'algorithmes d'apprentissage.

Pour comparer les algorithmes, nous devons évaluer leur performance relative à résoudre une certaine tâche. Dans notre cas, la problématique choisie est celle des voitures autonomes. Concrètement, une voiture est contrôlée par un algorithme dans un environnement composé de murs qui forment un labyrinthe. Nous évaluons ensuite l'algorithme selon la distance parcourue par la voiture sans heurter un mur. Pour percevoir l'environnement dans lequel la voiture évolue, l'algorithme reçoit en entrée la valeur de capteurs de distance rattachés à la voiture. Il peut ensuite effectuer différents calculs dépendant de ces valeurs pour dire à la voiture comment se déplacer. L'apprentissage d'un algorithme consiste à optimiser ses calculs, qui correspondent à l'interprétation de la valeur des capteurs.



Pour montrer l'utilité de notre plateforme, nous avons créé et comparé deux algorithmes d'apprentissage profond: Fully Connected Neural Network (FUCONN) et Mimicking Human Behavior (MHUB). FUCONN apprend à résoudre le problème des voitures autonomes avec des principes de génétique et MHUB le fait en imitant la conduite humaine. Pour les comparer, la plateforme entraîne chacun des algorithmes jusqu'à ce que leur amélioration stagne. Cette étape est importante, car chaque algorithme doit être testé à son meilleur. Ensuite, LAOP évalue les algorithmes dans 100 labyrinthes générés aléatoirement pour obtenir une moyenne de l'efficacité de chacun. Nous avons ainsi trouvé qu'avec FUCONN, la voiture parcourt en moyenne une distance de 42 085px sans frapper un mur, tandis que MHUB a une moyenne de 49 617px. Nous pouvons donc conclure qu'une fois entraîné, MHUB est en moyenne 15,2 % plus performant que FUCONN. Cependant, l'apprentissage de MHUB était beaucoup plus fastidieux, puisque nous devons contrôler la voiture avec notre clavier pendant 10 minutes en moyenne.