

## 8INF846 – Intelligence Artificielle – Hiver 2022

### Travail Pratique #3

















#### Agent logique – Le simulateur d'un robot qui sauve des vies

##### Le contexte

Son dernier projet au sein de l'entreprise d'intelligence artificielle est de construire un simulateur de robot qui sauve des vies. Le robot doit être un agent intelligent sous la forme d'un **système expert** utilisant la logique pour trouver des survivants. Il pourra rechercher des lieux dévastés, tels que des bâtiments en ruine et des bâtiments en feu ou même des villes victimes de catastrophes naturelles, pour trouver des survivants. Pour cela, grâce à ses capteurs, il doit détecter la chaleur et identifier les incendies ou détecter les poussières et identifier les décombres. Lors de la détection des incendies, il doit utiliser son extincteur pour jeter de l'eau et éteindre le feu. Lors de la détection de décombres, vous devez esquiver pour ne pas rester coincé. Le robot atteint son objectif en trouvant une personne survivante. Le robot sait qu'il a trouvé la personne car il dispose d'un microphone qui détecte ses cris. Le robot se déplace vers le haut, le bas, la gauche et la droite. Le robot ne se déplace pas en diagonale.

##### Détails de l'environnement

L'environnement doit être généré aléatoirement et doit être représenté par un tableau comme représenté ci-dessous. Au fur et à mesure que vous développez un simulateur, plusieurs scénarios avec différents niveaux de difficulté doivent être générés. Ainsi, initialement le tableau commence par une dimension 3x3 et à chaque niveau il augmente d'une dimension : 4x4, 5x5, successivement. Les décombres, le feu et la personne vivante doivent être ajoutés au hasard au moment de la génération de la table à chaque niveau. Le robot doit toujours démarrer à la position 0x0. Il ne devrait y avoir qu'une seule personne survivante à chaque niveau et cette personne devrait se voir attribuer un poste généré au hasard. Les cellules adjacentes aux décombres émettent de la poussière. Les cellules adjacentes aux feux émettent de la chaleur.

|   |   |   |  |   |
|---|---|---|--|---|
|  |  |  |  |  |
|   |   |  |  |  |
|  |   |  |  |   |
|  |  |  |   |   |
|  |   |  |  |  |

# L'agent

En plus des cellules déjà parcourues, l'agent ne peut voir que les cellules adjacentes à la position actuelle. L'agent décide vers quelle cellule il se déplacera grâce à l'inférence du système expert, qui permet de déterminer la meilleure cellule pour que le robot utilise son actionneur. Dès qu'il constate un incendie, l'agent peut éteindre le feu avec de l'eau grâce à son actionneur ou emprunter un autre chemin (l'algorithme de raisonnement déductif décidera de la décision à prendre en fonction de la probabilité générée). Dès qu'il remarque des décombres, l'agent doit contourner les décombres en empruntant un autre chemin. Une fois la personne trouvée, l'agent la sauve et passe à un niveau de difficulté plus élevé, augmentant la taille du tableau.

## Des contraintes à respecter

- Capteurs : capteur de chaleur, capteur de poussière, capteur de cri
- Actionneurs : extincteur, monter, descendre, se déplacer à gauche, se déplacer à droite, sauvegarder
- Environnement : partiellement observable (l'agent ne voit qu'une seule maison), déterministe, séquentiel (chaque épisode est indépendant), statique (le tableau ne change pas au même niveau), discret (temps et espace finis), agent simple (le feu et les décombres sont passifs)
- Le simulateur est infini, il peut passer aux niveaux de table NxN
- L'agent meurt et le jeu se termine s'il s'enlise dans les décombres ou tombe dans le feu
- Vous devez intégrer les notions de probabilité
- Modélisation : l'agent doit contenir une base de connaissances pour composer le moteur d'inférence
- Modélisation : l'agent doit stocker les faits (connaissances assertionnelles)
- Modélisation : l'agent doit contenir les règles de déduction (connaissances opérationnelles)
- Toutes les étapes du cycle du moteur d'inférence doivent être présentes (Filtrage, Choix d'une règle, Appliquer la règle)

## Entrée et sortie

- **Entrée** : l'application ne doit pas avoir d'interface graphique. Cela devrait nécessiter d'appuyer sur le bouton « Enter » du clavier pour passer au niveau suivant.
- **Sortie** : la sortie doit afficher le tableau du niveau actuel avec tous les éléments sur la console. Il doit également afficher la séquence d'actions que le robot a effectuée à ce niveau. Voici un exemple :

```
TP3 - Agent logique
le simulateur de robot sauve des vies

Tableau niveau 1 - 3x3
A - agent, F - feu, C - chauffer
P - poussière, D - décombres, G - gens

A  C  _
CP F  C
D  CP G

L'agent se déplace à droite
L'agent éteint le feu
L'agent se déplace à descend
L'agent se déplace à droite
L'agent se déplace à descend
L'agent sauve la personne

Appuyez sur Entrée pour passer au niveau suivant.
```

```
Tableau niveau 2 - 4x4
A - agent, F - feu, C - chauffer
P - poussière, D - décombres, G - gens

A  _  _  _
_  _  C  _
_  CP F CP
P  D  G  D

L'agent se déplace à droite
L'agent se déplace à droite
L'agent se déplace à droite
L'agent se déplace à descend
L'agent se déplace à descend
L'agent se déplace à descend
L'agent a été piégé dans les décombres. Fin du jeu.
```

## Autres informations importantes

- La modélisation et la mise en œuvre doivent être cohérentes avec les concepts vus en classe. Vous perdrez des points si les modèles sont inadaptés ou si l'algorithme est copié d'internet, même s'il fonctionne parfaitement.
- Le langage de programmation le plus recommandé est **Python**. Si vous souhaitez utiliser une autre langue, vous devez demander l'approbation du professeur.
- Vous devez former un groupe de **3 ou 4 personnes**.

## Soumission du travail

- Vous devez soumettre une **copie électronique de votre code source**.
- Un **rapport** (en format .pdf) contenant les sections suivantes :
  - Page de titre avec **les noms, les codes permanents et le titre du cours** ;
  - Section expliquant **comment exécuter votre programme** : quelle version de langue (Python, Java, etc.) doit être installée, quelles bibliothèques supplémentaires doivent être installées, paramètres supplémentaires, etc. Indiquez quel fichier .py exécuter. Attention : si le professeur a besoin de modifier une ligne de votre code pour exécuter votre programme, vous perdrez des points.
  - Section avec une **capture d'écran du code source** de chacun des éléments ci-dessous et **une brève explication** de chacun :
    - Le code (*IF*, *FOR*) qui compose les règles de déduction
    - La structure de données (liste, *array*, *matrix*, etc.) qui composent les faits
    - L'algorithme de raisonnement déductif
    - Les méthodes de chaque étape du cycle du moteur d'inférence (Filtrage, Choix d'une règle, Appliquer la règle)
    - Capteurs et actionneurs
- La **soumission** doit être **sur Moodle** uniquement.
- Une **seule soumission par groupe**.

**Date limite de remise : 04 avril 2022 à 23h59 (heure de Chicoutimi, QC, Canada)**

**Bon travail !**