

---

# Simulation multi-agents

# Simulation d'une voiture autonome

Master 2 Informatique, IMAGINE  
Université de Montpellier  
2021 - 2022

---

## Équipe :

- Charles Sayamath
- Guillaume Lanson
- Emery Bourget-Vecchio

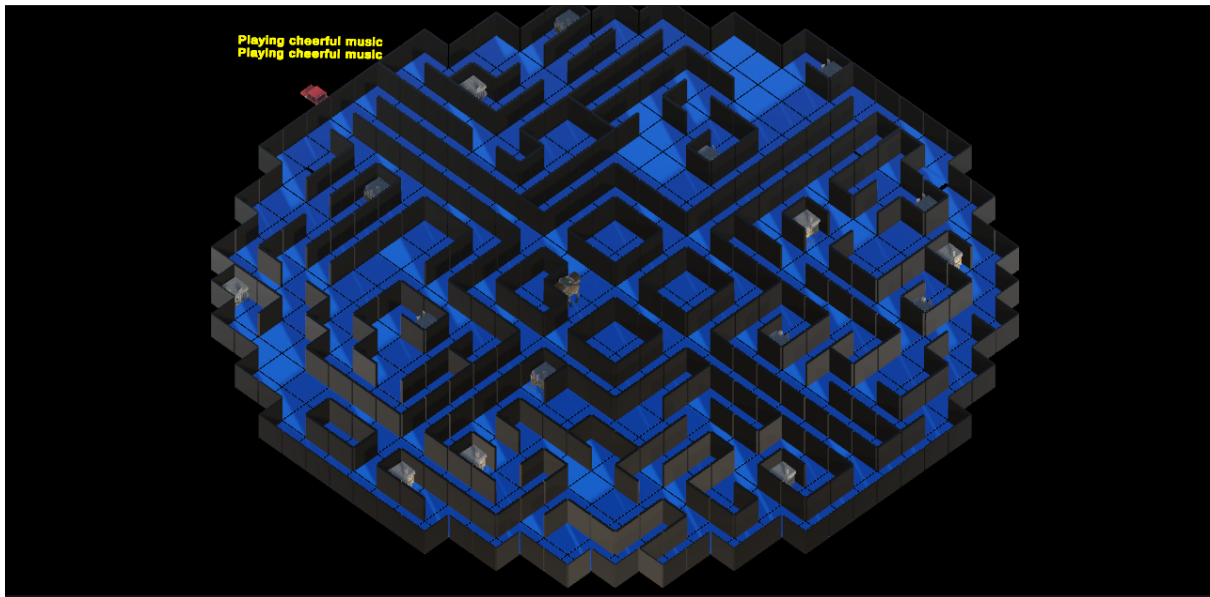


# ***SOMMAIRE***

<b>Simulation</b>	<b>3</b>
<b>Beliefs</b>	<b>7</b>
<b>Desires</b>	<b>7</b>
<b>Intentions</b>	<b>7</b>
<b>Evolution de l'environnement</b>	<b>8</b>
<b>Implémentation</b>	<b>8</b>
<b>Argumentation</b>	<b>8</b>

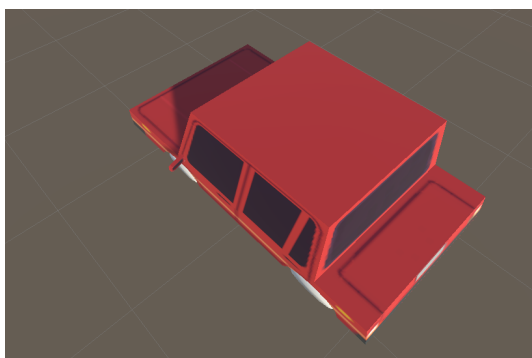
# **1. Simulation**

Pour réaliser cette simulation, nous nous sommes basés sur le projet préexistant disponible en cliquant [ici](#) qui nous permet d'avoir un simple modèle de path finding. Nous l'avons agencé de telle sorte à représenter une ville miniature.



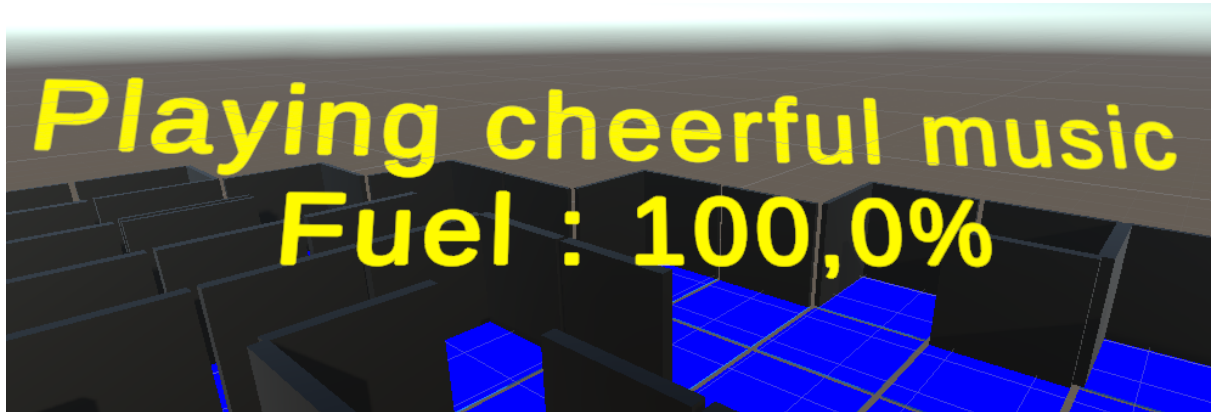
**Environnement de l'agent (ville miniature)**

Nos agents sont représentés par des voitures. Il existe des types de voiture qui diffèrent de par leurs apparences mais également de par leurs stockages maximums d'essence. Si une voiture n'a plus d'essence, elle disparaît. Ils seront capables de jouer de la musique normale si la voiture roule normalement et une musique calme si la voiture est dans les bouchons.

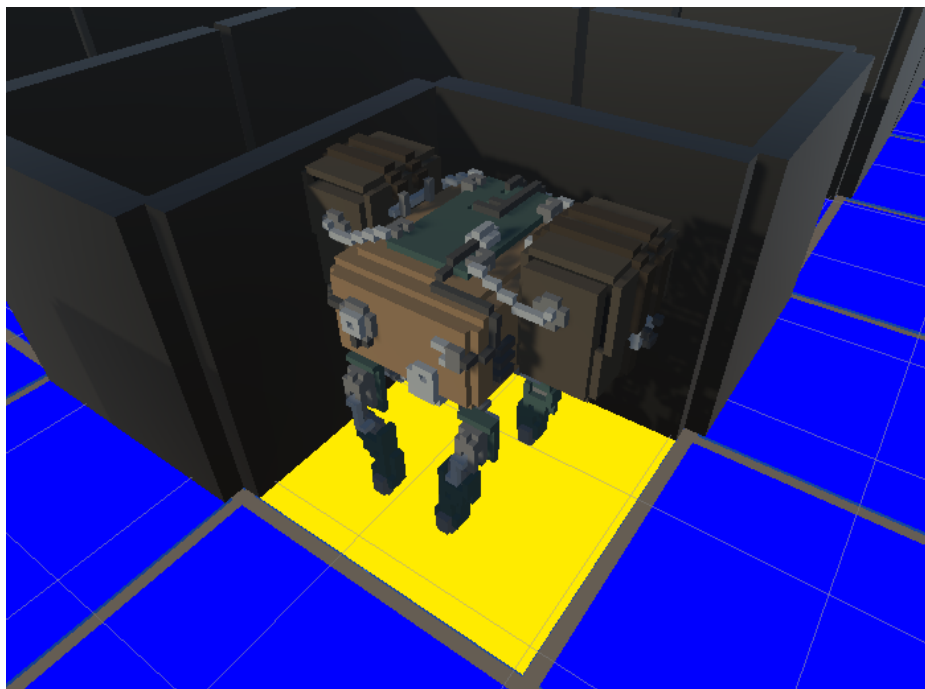


**Exemple de voiture disponible dans la simulation**

Chaque voiture a au-dessus d'elle un HUD qui permet d'indiquer le type de musique jouer(enjouée,calme) et le pourcentage d'essence restant.

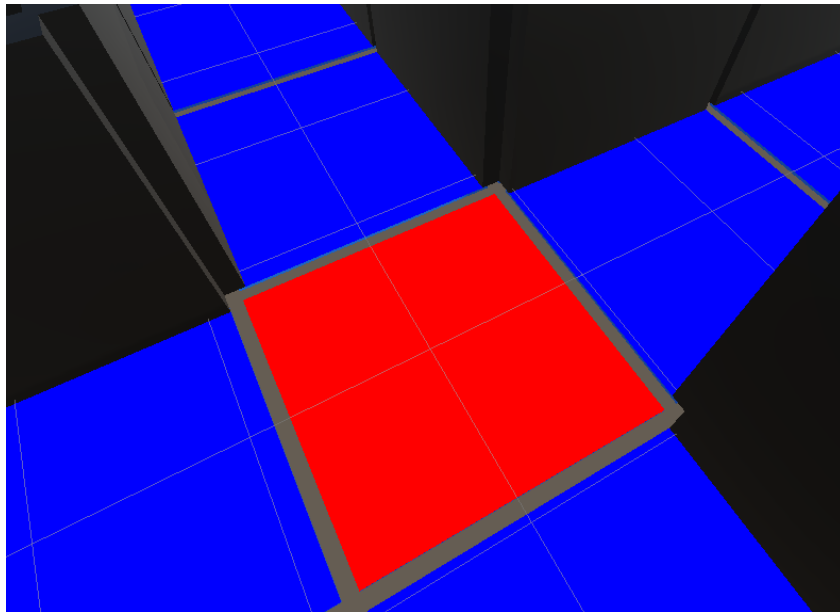


Lorsqu'une voiture manque d'essence, elle doit se rendre à la station service représenté ci-dessous.



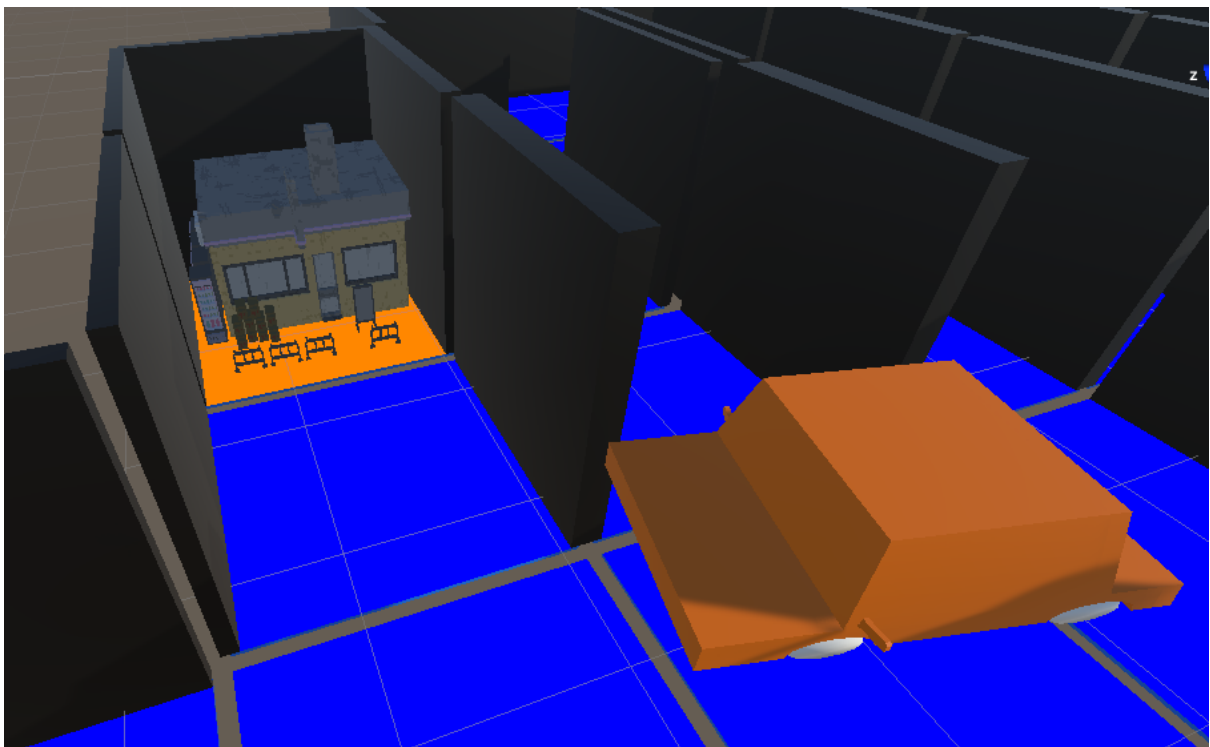
Station essence

Les cases bouchons sont des cases qui ralentissent les voitures qui passent dessus. Les agents vont donc faire en sorte de les éviter. Les bouchons sont créés soit par le croisement de deux voitures, soit par la mort d'un agent (son essence est vide).



**Exemple de tuile représentant un bouchon**

Les voitures ont pour objectif d'atteindre une certaine destination. Les destinations possibles sont représentées par des cases où se trouve une maison. Un agent doit se rendre à une destination si la couleur de la destination est la même que la couleur de la voiture.



**Exemple de cas où une voiture orange doit se rendre à une destination qui est de couleur orange**

## **2. Beliefs**

L'agent possède 3 beliefs :

- Un champ de vision global qui permet de prédéfinir un chemin pour amener son passager à bonne destination. Dans ce champ de vision, il ne connaît ni la position des autres voitures, ni la localisation de potentiel bouchon.
- Un champ de vision local, c'est à dire que l'agent à un champ de vision de n-cases ( dans notre cas 3). Il ne pourra voir les bouchons que dans ce champ de vision.
- Une information sur la quantité de carburant qu'il lui reste.

## **3. Desires**

- Les désirs de nos agents sont d'éviter les bouchons et d'arriver à destination le plus rapidement possible.  
Il souhaite également garder son passager le plus satisfait possible.  
La destination de chaque agent est choisie aléatoirement parmi une liste de destination pré-enregistrée.  
Une fois que la destination a été atteinte, l'agent choisira aléatoirement une autre destination et cela ainsi de suite.
- L'agent possède néanmoins un réservoir de carburant qu'il doit garder rempli sous peine de tomber en panne et de ne plus pouvoir réaliser son premier désir. Il aura ainsi comme autre désir de conserver son réservoir d'essence au-dessus d'un certain seuil de remplissage.

## **4. Intentions**

Pour remplir le désir d'arriver à destination rapidement, les agents utilisent le système de pathfinding et choisissent le trajet le plus court.

Si ce trajet implique de traverser des bouchons, afin d'éviter que le passager ne devienne frustré, l'agent joue une musique plus calme pour le détendre.

L'agent aura par moment l'intention de se déplacer vers la station d'essence la plus proche pour remplir son réservoir.

## **5. Evolution de l'environnement**

Notre environnement évolue en fonction du trafic routier. En effet des bouchons vont apparaître sous certaines situations :

- Lorsqu'au moins deux voitures passent sur une même case, un bouchon va alors être simulé sur celle-ci et les voitures se déplaceront alors plus lentement
- Lorsqu'une voiture tombe en panne d'essence, un bouchon va également être créé
- Les bouchons disparaissent après un certain temps

Ainsi, les beliefs des différents agents vont être dynamiques.

## **6. Implémentation**

L'environnement et les agents sont implémentés en C# avec le logiciel Unity, les éléments de la simulation ayant été étendus à partir du pathfinding basique de l'asset mentionné plus haut, le code présente peu de séparation entre les différents éléments de BDI.

## **7. Argumentation**

L'agent pourra tomber dans différentes situations lors de son trajet.

- Recherche d'une nouvelle destination
- Calcul d'une chemin vers la destination
- **Si** le niveau d'essence est suffisant :
  - Déplacement de la voiture au long du chemin
  - Si un bouchon rentre dans son champ de vision :
    - Calcul du chemin le plus court ne passant pas par le bouchon en question
    - **Si** ce chemin prend moins de temps à parcourir que l'actuel :
      - L'agent prend comme trajet le nouveau chemin
    - **Sinon** :
      - L'agent continue sur sa lancée
- **Sinon** :
  - L'agent se dirige vers la station essence

## **8. Améliorations**

Nous avons réfléchi à plusieurs moyens d'améliorer la simulation :

- Ajouter de la communication entre les agents, ainsi chaque agent pourrait étendre ses beliefs avec la position de bouchons vus ou créés par les agents qu'il croise.
- Ajouter un agent qui communiquerait avec l'humain, lui demandant une destination et lui demandant de faire des choix, par exemple s'il préfère prendre un chemin un peu plus long ou passer par un bouchon.
- Améliorer la communication des agents, en montrant en surbrillance le chemin qu'ils ont choisi, et en envoyant des messages lorsqu'ils re-calculent une trajectoire.
- Placer plusieurs stations essence, avec éventuellement une quantité de carburant dans chaque station qui se régénère avec le temps.

## **9. Démonstration**

Une vidéo de démonstration présente les différents cas, la vidéo se trouve ici :

<https://www.youtube.com/watch?v=Og2FciK3hn4>