

# DOCUMENTATION

## Espèce Vehicle

Comme vu dans le document sur la présentation de l'environnement, l'espèce Vehicle est une espèce générique, ses propriétés et fonctions vont donc être réutilisables par les classes filles. Vous trouverez dans ce document des explications sur chaque propriété et chaque fonction de cette espèce.

### 1 – Attributs/Propriétés

- float speed : vitesse du véhicule
- float max\_speed : vitesse maximale du véhicule
- bool must\_stay\_on\_road : booléen indiquant si le véhicule doit rester sur les voies ou non
- point target : Point cible du trajet du véhicule
- float coeff\_acceleration : coefficient permettant l'accélération ( $\text{nouvelleVitesse} = \text{vitesse} + \text{vitesse} * \text{coeff}$ )
- float coeff\_deceleration : coefficient permettant la décélération ( $\text{nouvelleVitesse} = \text{vitesse} - \text{vitesse} * \text{coeff}$ )
- bool can\_speed\_up : booléen indiquant si les conditions pour augmenter la vitesse du véhicule sont réunies
- float val\_origine : valeur à l'origine de l'équation de la droite de la direction du véhicule
- float coeff\_directeur : coefficient directeur de l'équation de la droite de la direction du véhicule
- bool vertical\_eq : booléen indiquant si le véhicule circule dans une direction totalement verticale
- bool crashed : booléen indiquant si le véhicule a eu un accident
- file car\_icon : icône de l'affichage graphique de l'espèce
- file crashed\_car\_icon : icône en cas de crash de véhicule
- file fleche\_icon : icône de la flèche indiquant la direction du véhicule
- file last\_location : position à l'étape t-1 de la simulation
- float angle\_rotation : angle de rotation du véhicule depuis le vecteur (0, -1, 0)
- point vecteur\_direction : vecteur direction à l'étape t du véhicule

- point secondaryTarget : cible secondaire imposée par une règle (exemple route à sens unique)
- point location : position à un instant t du véhicule
- int crash\_importance : entier correspondant à l'importance d'un véhicule dans un accident, par exemple si un vélo a un accident avec une voiture, grâce à cet indice, seul le vélo rentrera en collision
- int nb\_passagers : nombre de passagers d'un véhicule

## **2 – Fonctions/Actions/Reflexes**

### Action calcul\_eq\_route

Calcule l'équation de la droite de la route sur laquelle le véhicule circule, le calcul est effectué depuis deux points : la position à l'étape i et la position à l'étape i – 1. La formule mathématique utilisée est :  $\text{coeff\_directeur} = (y1 - y2) / (x1 - x2)$ ,  $\text{valeur\_origine} = y2 - \text{coeff\_directeur} * x2$  (équation depuis le deuxième point). Attribue les valeurs aux variables aux attributs coeff\_directeur, valeur\_origine et enfin vertical\_eq.

### Action freinage

Lorsqu'un freinage est demandé, attribue la nouvelle vitesse du véhicule selon le coefficient de freinage en suivant la formule :  $\text{vitesse} = \text{vitesse} - \text{vitesse} * \text{coeff\_deceleration}$ . A chaque freinage, le coefficient est augmenté de 0.10. Si la nouvelle vitesse est en dessous ou égale au seuil de vitesse minimale, le véhicule s'arrête.

### Action acceleration

Lorsqu'une accélération est demandée, attribue la nouvelle vitesse du véhicule selon le coefficient d'accélération en suivant la formule :  $\text{vitesse} = \text{vitesse} + \text{vitesse} * \text{coeff\_acceleration}$ . Si la vitesse calculée est supérieure à la vitesse maximale, alors la nouvelle vitesse du véhicule sera sa vitesse maximale.

### Action calcul\_angle\_rotation

Calcule l'angle de rotation de l'icône du véhicule sur la simulation, pour que le véhicule soit orienté dans le sens de sa direction, pour cela cette action calcule un angle depuis le vecteur {0, -1, 0}. Modifie l'attribut angle\_rotation du véhicule.

### Float distance between two vehicles(Vehicle otherVehicle)

Param in : le deuxième véhicule pour calculer la distance. Calcule la distance entre l'objet appelant et le véhicule passé en paramètre. Retourne un nombre float.

#### Action brake collision computing x axe(Vehicle otherVehicle)

Param in : le deuxième véhicule. Détermine si le véhicule appelant doit freiner si le véhicule otherVehicle risque de rentrer en collision avec lui sur une ligne droite. Cette fonction calcule sur l'axe x depuis les équations de droite (axe x car droite non verticale) des véhicules. Pour cela, elle fait des comparaisons sur les directions et sur la distance entre ces deux véhicules.

#### Action brake collision computing y axe(Vehicle otherVehicle)

Param in : le deuxième véhicule. Détermine si le véhicule appelant doit freiner si le véhicule otherVehicle risque de rentrer en collision avec lui sur une ligne droite. Cette fonction calcule sur l'axe y depuis les équations de droite (axe y car droite verticale) des véhicules. Pour cela, elle fait des comparaisons sur les directions et sur la distance entre ces deux véhicules.

#### Action brake if collision coming

Le véhicule appelant va vérifier tous les autres véhicules pour déterminer s'il peut y avoir une collision avec le véhicule appelant. Si oui, le véhicule appelant freine.

#### Action rentre en collision

Change les paramètres du véhicule mis en cause lors d'une collision : le booléen crashed, la vitesse, et son icône sur la simulation.

#### Action which ones crash(Vehicle otherVehicle)

Param in : le deuxième véhicule impliqué dans la collision. Dans une collision, cette action compare les deux coefficient crash\_importance des deux véhicules. S'ils sont égaux, les deux véhicules rentrent en collision. Sinon, seulement celui avec le plus bas coefficient rentre en collision.

#### Action verification collision

Vérifie qu'il n'y a pas eu une collision pendant une étape de la simulation entre le véhicule appelant et tous les autres véhicules. Pour cela des calculs sont effectués sur la distance entre les véhicules et sur les équations de droite qui sont associées aux routes.

#### List<Rule> perception

Retourne la liste des règles de la route sur laquelle le véhicule appelant se trouve. Utilise la fonction at\_distance fournie par Gama pour connaître la route utilisée.

#### Action decision(List<Rule> rules)

Applique les règles passées en paramètre par la liste rules une par une au véhicule appelant.

#### Reflexe perceptionDecision

A chaque étape de la simulation, le véhicule récupère la liste des règles à appliquer et les applique.

#### Reflex setup\_target

Lorsque le véhicule est arrivé à sa destination, un nouveau checkpoint est choisi comme destination pour que le véhicule reste en mouvement.

#### Reflex car\_at destination

Lorsque le véhicule arrive à destination, ce réflexe met à jour un booléen permettant de valider la condition du reflex setup\_target pour changer de destination.

#### Reflex mise a jour

A chaque étape de la simulation, la position du véhicule et certains attributs sont mis à jour en fonctions d'autres propriétés. Ce reflexe est redéfini par les espèces filles GuidableVehicle et NonGuidableVehicle.

### **3 – Aspects**

#### Aspect with icon

Représente le véhicule dans la simulation avec un icône, affiche aussi sa vitesse et sa direction avec une flèche.

## **Espèce GuidableVehicle (hérite de Vehicle)**

### **1 – Attributs/Propriétés**

- Int idGuidable : indice identifiant chaque véhicule guidable par Bluetooth pour pouvoir utiliser les différentes actions disponibles reliées en Bluetooth
- Bool is\_connected : booléen indiquant si le véhicule est connecté ou non, pour éviter d'envoyer des requêtes si le véhicule n'est pas connecté en Bluetooth

### **2 – Actions/Fonctions/Reflexes**

### Reflexe mise a jour

Met à jour de la position de la voiture en fonction des données reçues envoyées par la caméra. (Redéfinition depuis Vehicle)

### Action rentre en collision

Change les paramètres du véhicule mis en cause lors d'une collision : le booléen crashed, la vitesse, et son icône sur la simulation. De plus, déconnecte le véhicule du réseau Bluetooth.

## **Espèce NonGuidableVehicle (hérite de Vehicle)**

### **1 – Attributs/Propriétés**

- Int idNonGuidable : indice identifiant chaque véhicule faisant seulement partie de la simulation interne de Gama, non réelle avec la connexion Bluetooth

### **2 – Attributs/Propriétés**

#### Reflexe mise a jour

Met à jour la position du véhicule en fonction de certaines propriétés. Peut aussi changer d'autres attributs tels que la vitesse. Enfin, calcule l'équation de droite de la route, l'angle de rotation de l'image associé à la direction et vérifie qu'il n'y a pas de collision.