

## **Compte rendu - Sprint 1 - Team l'éclair**

Notre premier objectif était de simplement faire en sorte que notre agent puisse réussir à faire un tour sans sortir de la piste quand bien même notre kart pouvait être lent.

Pour ce faire, nous avons pensé à utiliser le vecteur `center_path` mais nous nous sommes rendus compte que la voiture n'était pas stable et même qu'il sortait de la piste lorsque la pente était descendante.

En explorant et en comprenant l'utilité des autres vecteurs d'observations, nous nous sommes rendus compte de l'existence du vecteur `paths_end`. Nous pensions au début que ce vecteur pouvait représenter le dernier vecteur (c'est-à-dire celui avant la ligne d'arrivée) mais nous nous sommes rendus compte que ce n'était pas le cas.

Nous avons ensuite changé la valeur de l'accélération à 1 et extrait la variable `x` du premier vecteur de la liste. Avec simplement l'implémentation de ce vecteur dans notre code, notre kart pouvait rouler très rapidement et rester sur la piste même s'il se contentait de suivre bêtement le centre de la route et fonçait droit sur des peaux de bananes s'ils étaient en plein milieu du chemin.

Notre deuxième objectif était que notre agent puisse conduire de manière plus réaliste, nous avons donc introduit le freinage à notre code. L'usage du freinage dépend de la présence imminente d'un virage ou non.

Nous avons ensuite fait plusieurs tests et nous nous sommes rendus compte que dans certaines pistes, notre kart pouvait être bloqué dans un mur. Nous avons essayé tant bien que mal de faire en sorte que le kart puisse être secouru par l'oiseau bleu via l'usage de la variable prédéfinie "rescue". Cependant nous avons remarqué que même si la variable pouvait être à `True` ou `False`, l'oiseau bleu venait sauver le kart uniquement si ce dernier sortait complètement de la route mais pas lorsque ce dernier est bloqué dans un mur.

Nous avons donc décidé de trouver une solution alternative et la solution qui nous a paru la plus évidente était que notre kart puisse faire marche arrière. Pour ce faire, nous avons implémenté une nouvelle variable (`time_blocked`) dans `init` car si nous la définissions dans `choose_action`, cette dernière revenait toujours à 0 donc l'incréméntation était impossible et la logique de notre code ne pouvait fonctionner. Ainsi, si le kart était bloqué pendant une courte période, il faisait marche arrière et pouvait reprendre sa route normalement.

Ce premier sprint aura été l'occasion pour nous de découvrir le système du jeu et d'en apprendre plus sur le fonctionnement des vecteurs d'observations, quand bien même nous ne savons toujours pas l'utilité et le fonctionnement de l'ensemble des vecteurs d'observations. Cependant nous sommes satisfait de notre travail pour cette semaine et espérons que le prochain sprint sera tout aussi productif que celui-ci.