

# bClass Agent Speed:

## Rappel:

Pour l'instant grâce a Agent Center le kart cherche en permanence a avancer vers l'avant en maintenant une acceleration de base, tout en restant au centre de la piste.

## But:

Agent Turn sert à améliorer la conduite par rapport a AgentCenter, il va anticiper les virages et adapter la vitesse du kart en fonction de la piste. Grâce a ça on pourra maintenir une trajectoire stable et éviter les sorties de pistes.

## Strategie:

### Observation de la piste devant le kart:

L'agent va devoir observer un certain nombres de segments devant lui en utilisant les variable **paths\_start**, **paths\_end** et **paths\_distance**.

Grâce a cette vision anticipé l'agent va pouvoir comprendre la forme de la piste. En comparant la direction et la longueur des segments qui arrivent le kart evalue si la piste est droite ou si elle tourne. Toujours en utilisant **paths\_start**, **paths\_end** et **paths\_distance**.

L'agent va classer la portion de piste qui arrivent de maniere suivante:

- Ligne droite : Faible variation de direction
- Virage serré : Forte variation de direction

## Réactions:

L'agent adapte sa vitesse a l'angle du virage detecté, il accélère si on detecte une ligne droite et ralentit pendant les virages, en utilisant l'**angle calculé** et la **variable acceleration**.

En meme temps l'agent doit comparer le virage a venir avec la capacité de braquage disponible pour éviter de tourner plus que ce qui est possible a sa vitesse acutelle, e, utilisant **acceleration** et **max\_steer\_angle**.

Les variables utilisés seront definies via OmegaConf et pourrons changer apres les test.

# Pseudo Code:

AgentTurn herite d'AgentCenter.

On utilise path\_lookahead pour regarder 3 segments devant l'agent

## #Analyse des segments:

Pour chaque segment, on:

- Calcule de la direction du segment ( $\text{paths\_end}[i] - \text{paths\_start}[i]$ )
- Calcule de l'ecart de direction entre front et direction du segment
- Calcule de la longueur du segment ( $\text{paths\_distance}[i][1] - \text{paths\_distance}[i][0]$ )
- Determiner le type de segment (droite, virage léger, virage serré)

#Avec i l'indice du segment que je regarde parmi les 3 prochains:

i = 0 -> segment actuelle, i > 0 segments qui arrivent

## #Identifier la situation la plus contraignante:

Si au moins un segment est un virage serré -> on reagit comme un virage serré

Si au moins 1 segment est un virage léger -> on reagit comme un virage léger

Si aucun des deux cas precedent -> on reagit comme une ligne droite

## #Reactions:

### #Ligne droite

if ligne droite:

velocity max

### #Virage léger

if virage léger:

#Est ce que je peux le prendre a ma vitesse actuelle ?

if max steer angle petit (si la capacité de braquage actuelle est limitée):

#Ajustement de la vitesse

velocity -

elif max steer angle moyen:

#Ajustement de la vitesse

maintenir

else:

#Ajustement de la vitesse

velocity++ ++ #forte acceleration

### #Virage serré

if virage serré

#Est ce que je peux le prendre a ma vitesse actuelle ?

if max steer angle petit (si la capacité de braquage actuelle est limitée):

#Ajustement de la vitesse

```

velocity- - - #fort ralentissement
elif max steer angle moyen:
    #Ajustement de la vitesse
    maintenir
else:
    #Ajustement de la vitesse
    velocity++ # acceleration plus legere

```

```
seglong = obs["paths_distance"][i][1] - obs["paths_distance"][i][0]
```

## Test maps:

Avec les valeurs suivantes:

ecartpetit=1, ecartgrand=3, msapetit=1, msagrand=3

ligne droite:

act["acceleration"] = 0.3

virages serrer:

msa petit -> accel = accel - 5

msa grand -> accel = accel + 2

### 1) Abyss:

vitesse assez lente en ligne droite (stagne), tourne au premier virage sans décélérer puis descelere a la fin du virage(trop tard), arrive au 2eme virage avec une vitesse tres lente puis re accelere dans le virage ce qui l'enmene a se bloque contre un mur Ralentit trop tard et accelere trop tot. Pas d'anticipation, reagit lorsqu'il est sur le segment dit.

### 2) SandTrack:

accelere un peu plus au debut car ligne droite + grande, les virage sont leger donc il ralentit pas trop, arrive a suivre le circuit plus de 30sec, n'st jamais trop lent. meme probleme, reagit a l'instant

### 3) Fort magma:

grand virage leger des le debut, reussi a vitesse assez rapide, s'arrete en arrivant a la ligne droite au dessus du magma, ralentit fortement (surement le -5 si un virage serré c'est trop) variables a changer.

### 4) Cocoa\_temple:

Avance a une bonne vitesse, sur les virages leger ne ralentit pas trop, juste vitesse + virage meme leger entraine beaucoup de zigzag, Tombe d'un pont qui a 2virages legers de

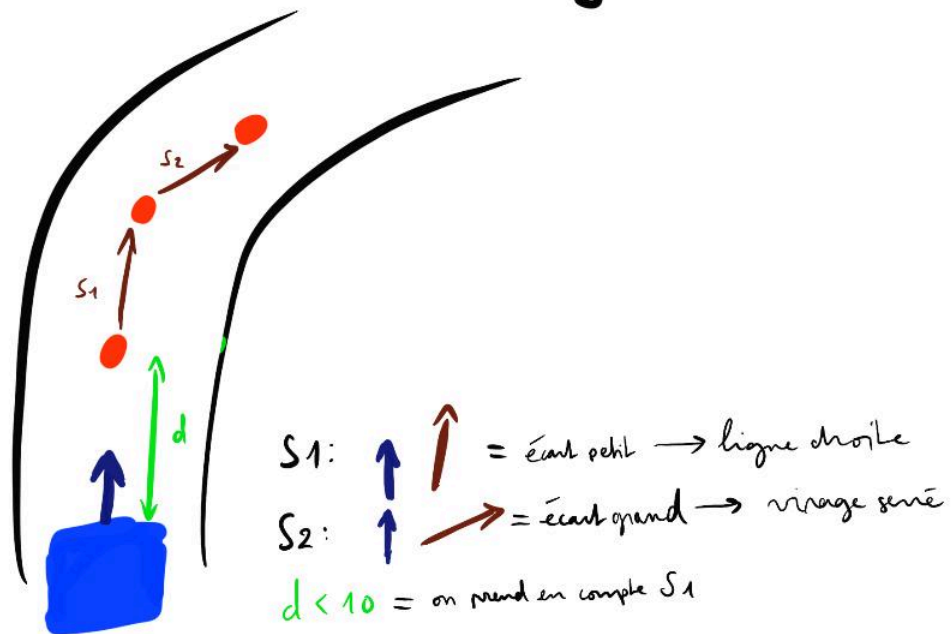
suite a cause de la vitesse qui etait assez rapide et des zigzag. Apres modification accel = accel + 0.3 virage leger + msa petit, c'est mieux le kart ne tombe pas dans le pont

5) Volcano\_island:

kart beaucoup trop rapide ce qui amene a faire bcp de zigzag, la piste est plus mince, essayer de faire la vitesse dans ces cas la fort

## Version 2.0:

Virage :



Pontée:

