Ce module extrait les informations de la piste et analyse les différents éléments afin de le lier.

------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*------

/!\ Il faut utiliser Python 3.6. pour que ce module marche. Python 2.7 ne donne pas les bons résultats à cause de la gestion des float sur 32 bits contrairement à Python 3.6.

------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*------

-Le fichier d'entrée doit être au format .dxf. Il contient la carte de la piste. Si le fichier d’origine est au format .dwg, il faut changer son extension en l’ouvrant sous DraftSight et faire un *enregistrer sous* en sélectionnant le bon format.

-Le fichier de sortie doit être au format .npz. Il contient toutes les informations après traitement de la piste.

-La librairie dxfgrabber doit être installée dans l'environnement python.

/!\ Le fichier fourni les données en effectuant une rotation et une translation dans l’espace. Ces informations sont disponibles dans le header du fichier et expliquent certaines étapes du traitement qui ont dû être faites en plusieurs étapes.

-Les legs de la trajectoire ne peuvent se couper au milieu. Si c'est le cas, il faut modifier le fichier d'entrée pour séparer en deux le leg coupé.

- On lance sur un terminal python3.6 le script en faisant:

main\_extraction.py -i <Fichier\_entree> -o <Fichier\_Sorti>

OU

main\_extraction.py --ifile=<Fichier\_entree> --ofile=<Fichier\_Sorti>

OU

La combinaison des 2 premiers. Attention ne pas mettre -i et --ifile, idem pour le output.

En cas de doutes, pour obtenir un exemple, lancer:

main\_extraction.py -h

-Le fichier de sorti au format .npz peut être extrait via la bibliothèque numpy et de nombreux exemple d’extraction sont disponibles sur le net, notamment sur StackOverflow.

-Ce fichier est la sauvegarde du traitement du fichier d’entrée. Il contient un dictionnaire python nommé “mapping” et possédant différentes clés tels que :

- Leg, qui renvoie la liste des legs de la piste

- Waypoint, qui renvoie la liste des différents waypoints

- Balise, qui renvoie la liste des différentes balises

- Contraintes, qui renvoie la liste des différentes contraintes

-L’architecture des différentes classes est disponible dans le compte rendu du PIE, ou bien en allant directement dans le code où l’on a essayé de commenter le plus richement possible.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Remarques pratiques :

-Deux points ont été jugé égaux si leur distance est inférieur au cm

-Un point est considéré être sur une ligne ou un arc si leur distance est inférieur au cm.

-Pour que deux legs soient considérés comme voisins, il faut impérativement qu’au moins une de leur extrémité soit commune ET qu’en ce point les legs ne forment pas un angle aigu. Cela permet au robot de ne pas faire de demi-tour ou d’arrêt brutal sur la piste.

-On considère que deux legs se rejoignant en leur milieu ne sont pas liés, ie nous avons procédé à une modification du fichier d’origine pour remplir cette spécification.