

Documentation sur la config du HL

Le nom de la config est en majuscule. La valeur entre parenthèse est la valeur par défaut. `config.ini` surcharge ces valeurs par défaut. Format de `config.ini` (une valeur par ligne) :

```
CONFIG_NAME = value
```

1 Mécanique du robot

- DILATATION_ROBOT_DSTARLITE (60) : dilatation des obstacles dans le D* Lite. Comme c'est une heuristique, on peut prendre plus petit que la vraie valeur
- DILATATION_ROBOT_ENNEMIDSTARLITE (130) : dilatation du robot ennemi dans le calcul des masques
- CENTRE_ROTATION_ROUE_X (204) : la position du centre de rotation des roues. Est utilisé pour la rotation des capteurs
- CENTRE_ROTATION_ROUE_Y (64) :
- DEMI_LONGUEUR_NON_DEPLOYE_ARRIERE (80) : distance entre le centre du robot et le bord arrière du robot non-déployé
- DEMI_LONGUEUR_NON_DEPLOYE_AVANT (332-80) : distance entre le centre du robot et le bord avant du robot non-déployé
- LARGEUR_NON_DEPLOYE (228) : distance entre le bord gauche et le bord droit du robot non-déployé

2 Scripts

- RAYON_CERCLE_ARRIVEE (200) : distance souhaitée entre le centre du robot et le centre du cratère
- VITESSE_ROBOT_TEST (300) : vitesse de test en mm/s
- VITESSE_ROBOT_STANDARD (500) : vitesse standard en mm/s
- VITESSE_ROBOT_BASCULE (300) : vitesse pour passer la bascule en mm/s
- VITESSE_ROBOT_REPLANIF (300) : vitesse en replanification en mm/s

3 Log

- FAST_LOG (false) : affichage plus rapide des logs

- AFFICHE_DEBUG (true) : affiche aussi les log.debug
- SAUVEGARDE_LOG (false) : sauvegarde les logs dans un fichier externe
- AFFICHE_CONFIG (false) : affiche la configuration complète au lancement
- COLORED_LOG (false) : de la couleur dans les sauvegardes de logs!

4 Ennemi

- RAYON_ROBOT_ADVERSE (200) : le rayon supposé du robot adverse, utilisé pour créer des obstacles de proximité

5 Pathfinding

- COURBURE_MAX (3) : quelle courbure maximale la trajectoire du robot peut-elle avoir
- TEMPS_ARRET (800) : temps qu'il faut au robot pour s'arrêter et repartir (par exemple à cause d'un rebroussement)
- PF_MARGE_NECCESSAIRE ((int) (0.10/ClothoidesComputer.PRECISION_TRACE)) : combien de points de pathfinding le bas niveau doit-il toujours avoir
- PF_MARGE_INITIALE ((int) (0.15/ClothoidesComputer.PRECISION_TRACE)) : combien de points garde-t-on au début de la replanification
- DUREE_MAX_RECHERCHE_PF (3000) : durée maximale que peut prendre le pathfinding
- TAILLE_FAISCEAU_PF (10) : combien de voisins sont ajoutés à l'open-set à chaque itération
- DEBUG_PF (false) : affichage de plein d'infos
- EVITEMENT_SIMPLE (true) : évitement simplifié : le robot attend juste que le chemin se libère
- DEBUG_CACHE (false) : debug du cache de chemins

6 Série

- SERIAL_TIMEOUT (30) : quel TIMEOUT pour le protocole série des trames ? (en ms) TODO
- BAUDRATE (115200) : le baudrate de la liaison série
- SERIAL_PORT ("/dev/ttyS0") : le port de la liaison série
- SLEEP_ENTRE_TRAMES (0) : la durée minimale entre deux envois de nouvelles trames
- SIMULE_SERIE (false) : la série doit-elle être simulée (utile pour debug)

7 Capteurs

- `SENSORS_SEND_PERIOD` (20) : période d'envoi des infos des capteurs (ms)
- `SENSORS_PRESCALER` (5) : sur combien de trames a-t-on les infos des capteurs
- `DUREE_PEREMPTION_OBSTACLES` (3000) : pendant combien de temps va-t-on garder un obstacle de proximité
- `DISTANCE_MAX_ENTRE_MESURE_ET_OBJET` (50) : quelle marge d'erreur autorise-t-on entre un objet et sa détection
- `DISTANCE_BETWEEN_PROXIMITY_OBSTACLES` (50) : sous quelle distance fusionne-t-on deux obstacles de proximité ?
- `IMPRECISION_MAX_POSITION` (20.) : quelle imprecision maximale sur la position du robot peut-on attendre (en mm)
- `IMPRECISION_MAX_ORIENTATION` (0.1) : quelle imprecision maximale sur l'angle du robot peut-on attendre (en radians)
- `TAILLE_BUFFER_RECALAGE` (5) : combien de mesures sont nécessaires pour obtenir une correction de recalage
- `PEREMPTION_CORRECTION` (100) : temps maximale entre deux mesures de correction au sein d'un même buffer (en ms)
- `ENABLE_CORRECTION` (true) : la correction de position et d'orientation est-elle activée ?

8 Gestion de la mémoire

- `NB_INSTANCES_NODE` (500) :
- `NB_INSTANCES_NODE` (50000) : nombre d'instances pour les nœuds du pathfinding
- `NB_INSTANCES_OBSTACLES` (`NB_INSTANCES_NODE.getDefaultValue()`) : nombre d'instances pour les obstacles rectangulaires

9 Verbose

- `DEBUG_SERIE_TRAME` (false) : debug verbeux sur le contenu des trames
- `DEBUG_SERIE` (false) : debug sur la série
- `DEBUG_CAPTEURS` (false) : debug verbeux sur les capteurs

10 Graphe de dépendances

- `GENERATE_DEPENDENCY_GRAPH` (false) : génère le graphe des dépendances

11 Interface graphique

- GRAPHIC_HEURISTIQUE (false) : affichage des orientations heuristiques données par le D* Lite
- GRAPHIC_ENABLE (false) : désactive tout affichage si faux (empêche le thread d’affichage de se lancer)
- GRAPHIC_D_STAR_LITE (false) : affiche les calculs du D* Lite
- GRAPHIC_D_STAR_LITE_FINAL (false) : affiche l’itinéraire final du D* Lite
- GRAPHIC_PROXIMITY_OBSTACLES (true) : affiche les obstacles de proximité
- GRAPHIC_TRAJECTORY (false) : affiche les trajectoires temporaires
- GRAPHIC_TRAJECTORY_ALL (false) : affiche TOUTES les trajectoires temporaires
- GRAPHIC_TRAJECTORY_FINAL (true) : affiche les trajectoires
- GRAPHIC_FIXED_OBSTACLES (true) : affiche les obstacles fixes
- GRAPHIC_GAME_ELEMENTS (true) : affiche les éléments de jeux
- GRAPHIC_ROBOT_COLLISION (false) : affiche les obstacles du robot lors de la vérification des collisions
- GRAPHIC_BACKGROUND_PATH ("img/background-2017-color.png") : affiche d’image de la table
- GRAPHIC_ROBOT_PATH ("img/robot_sans_roues.png") : image du robot sans les roues
- GRAPHIC_ROBOT_ROUE_GAUCHE_PATH ("img/robot_roue_gauche.png") : image de la roue gauche
- GRAPHIC_ROBOT_ROUE_DROITE_PATH ("img/robot_roue_droite.png") : image de la roue droite
- GRAPHIC_PRODUCE_GIF (false) : produit un gif?
- GRAPHIC_BACKGROUND (true) : affiche d’image de la table
- GRAPHIC_SIZE_X (1000) : taille par défaut (sans image) de la fenêtre
- GRAPHIC_ALL_OBSTACLES (false) : affiche absolument tous les obstacles créés
- GRAPHIC_ROBOT_AND_SENSORS (true) : affiche le robot et ses capteurs
- GRAPHIC_CERCLE_ARRIVEE (false) : affiche le cercle d’arrivée
- GRAPHIC_TIME (false) : affiche le temps écoulé
- GRAPHIC_TRACE_ROBOT (true) : affiche la trace du robot
- GRAPHIC_EXTERNAL (true) : l’affichage doit-il être déporté par le serveur d’affichage?