Robot: 100.75.155.207

Etape 1 : algorithme de déplacement autonome

Afin de nous convaincre d’utiliser votre algorithme, nous demandons :

• **des garanties théoriques de fonctionnement ;**

**• des validations en simulation ;**

**• des validations expérimentales sur le robot.**

Etape 2 : supervision du véhicule

Ainsi, nous vous demandons de :

• **Visualiser la position (x, y, θ) du véhicule dans l’espace**

**• Utiliser Rviz et RQt pour visualiser les capteurs, et positionner les obstacles autours**

**du robot**

**• Créer une interface de votre choix (e.g. Python) pour commander le robot selon des**

**instructions simples avec ROS**

**Travail demandé**

• Développement de l’algorithme de commande pour

amener le robot vers une position et une orientation cible

• Mise en œuvre de la solution sous ROS

• Développement d’une interface de supervision

• Afficher sur un poste de supervision l’état du robot

• Commande du robot à distance : démarre, stop, action 1, …

N équipes de 3 personnes

• Découpage en tâches et répartition entre les membres

• Intégration, tests, validation et documentation : toute l’équipe

Grâce à la formation sur **la commande de véhicules autonomes**, vous obtiendrez les compé-

tences suivantes :

• **la connaissance des principaux types de véhicules sur roues ;**

**• la capacité à modéliser un véhicule de type voiture ;**

**• la capacité à concevoir une loi de commande pour déplacer un véhicule de type voiture**

**de manière autonome ;**

**• la capacité à analyser vos lois de commandes afin d’obtenir certaines garanties théo-**

**riques ;**

**• la capacité à simuler un véhicule de type voiture et sa commande et à interpréter les**

**simulations.**

Grâce à la formation concernant **la programmation sous ROS et Python**, vous obtiendrez les

compétences suivantes :

• **Rechercher et utiliser les documentations et les bibliothèques concernant un produit**

**(ici notre robot LIMO) ;**

**• Comprendre les avantages et inconvénients d’un middleware comme ROS 1 et 2 ;**

**• Rappels et usages de Linux ;**

**• Comprendre les principes de bases de ROS et le prendre en main rapidement ;**

**• Afficher et manipuler les différents objets propres à ROS ;**

**• Programmer en Python des nodes, topics, paramètres, etc ;**

**• Utiliser des outils (Rviz et RQt) pour visualiser les données dans l’espace en 3D, afficher**

**les informations du système, tracer des courbes, etc**

**Formation autom fin 18/10. Début tâches possible le 21/10**

**Formation programmation fin 8/10. Début tâches possibles le 12/11**