

1. Description du contexte

Vous travaillez dans une entreprise de robotique sous-marine et vous devez coder une solution de mesure. Le drone en question est un drone de surveillance permettant de contrôler et de vérifier des installations sous-marines (pipeline...). Le drone est équipé de 4 turbines, d'une batterie, d'un éclairage, d'une caméra et de nombreux capteurs dont un capteur de pression qui permet d'estimer sa profondeur. Le drone est



Figure 1. Outland Submarine

contrôlé à partir d'une télécommande équipée de joysticks, d'un capteur de pression et de leds indiquant l'état actuel de différents éléments du drone, comme le niveau de charge de la batterie, ou des avertissements concernant par exemple une distance trop importante entre le drone et le pilote. par exemple une distance trop importante entre le drone et le pilote. Vous êtes en charge de l'estimation de la profondeur du drone, qui doit être calculée à partir de la pression mesurée à bord et sur la télécommande. mesurée à bord et sur la télécommande. Vous devrez utiliser ces mesures pour créer une alerte afin d'avertir le pilote qu'il est nécessaire d'effectuer un changement de direction. qu'il faut faire remonter le drone avant de perdre la communication avec le drone ou de l'endommager à cause de la haute pression. de l'endommager à cause de la haute pression. La profondeur (en mètres) peut être estimée avec la relation suivante (en considérant que l'eau est incompressible)

$$Depth = \frac{P_{robot} - P_{remote}}{\rho * g} \quad (1)$$

avec ρ la densité de l'eau salée = 1030 kg/m^3 , $g = 9,807 \text{ N/Kg}$, P_{robot} (Pa) la pression mesurée sur le robot et P_{remote} (Pa) la pression mesurée sur la télécommande à la surface de l'eau.

Vos collègues ont déjà développé un nœud ROS `submarinedrone` fonctionnant sur le drone, et communiquant directement avec le matériel du système, et publiant les valeurs des capteurs sur des sujets dédiés. Pour l'instant, les ingénieurs sont capables de publier (i) la pression mesurée sur le robot par le capteur correspondant dans le sujet `/pressure_robot` (en Pa), et (ii) la pression mesurée sur la télécommande par le capteur correspondant dans le sujet `/pressure_remote` (en Pa).

De plus, le nœud ROS `submarinedrone` est également chargé de déclencher certaines LEDs sur la télécommande utilisée par le pilote. Vos collègues n'ont jusqu'à présent implémenté que le déclenchement de l'indicateur `DEPTH WARN`, qui s'allume en vert si le drone n'est pas trop profond. Le changement de couleur de la LED est effectué en appelant le service `/depth_warning`, qui utilise comme entrée la chaîne de texte « NORMAL » si le drone n'est pas trop profond ou « WARNING » s'il est limité.

Votre objectif est d'écrire un nœud ROS appelé `exam`, qui doit publier une estimation de la profondeur du drone sur un nouveau topic `/depth` (en m), et déclencher si nécessaire l'indicateur `DEPTH WARN` sur la télécommande en appelant le service approprié.

2. Evaluation

A la fin de l'examen, vous devez uploader 2 fichiers dans moodle :

- Un rapport PDF, écrit avec openoffice par exemple (en français ou en anglais), où vous répondez aux questions ci-dessous et où vous pouvez copier-coller les résultats des différentes réponses aux questions ci-dessous et où vous pouvez copier et coller les sorties des différentes commandes que vous utilisez pour répondre aux questions. Par exemple, pour la question 6. Listez tous les sujets en cours d'exécution, vous devez indiquer dans votre rapport :
 - la ligne de commande exacte que vous avez utilisée pour répondre à la question,
 - et la sortie correspondante que vous obtenez.

Bien entendu, vous devez commenter toutes vos réponses. Un simple copier-coller sans une phrase expliquant les résultats n'est pas considéré comme une réponse complète aux questions.

- Un fichier ZIP de votre paquet exam qui sera utilisé pour accéder) votre code et aussi pour le lancer.

3. Preparation

1. Pour commencer, téléchargez le submarinedrone.zip depuis Moodle. Extrayez-le dans le dossier `src` dans votre espace de travail catkin. Lancez `catkin_make` pour construire votre espace de travail ROS et sourcez votre fichier `.bashrc`.
2. Allez dans le dossier `src` du package `submarinedrone` que vous venez de télécharger, et rendez exécutables tous les scripts python qui s'y trouvent avec la commande shell suivante `chmod +x python_file.py`.
3. Testez rapidement que tout va bien en exécutant la commande `node submarine_drone_node`. Contactez votre professeur si ce n'est pas le cas.