Intégration d'un robot Roomba dans un réseau de capteurs

Lucile COSSOU

Stagiaire à Inria Grenoble du 13/05/2013 au 02/08/2013

Sous le tutorat de M. Roger PISSARD-GIBOLLET

Sommaire

- Contexte
- Le robot
- Prise en main et évaluation
- Implémentation
- Résultats et perspectives d'évolution
- Conclusion

Contexte

Contexte – Lieu du Stage

L'Institut National de Recherche en Informatique et Automatique de Grenoble en quelques données :

- 680 employés
- 33 équipes de recherche
- 9 services
- 3 axes de recherche prioritaires : la conception de logiciel, la modélisation et la simulation, les interfaces Homme-Machines
- 4 plateformes d'expérimentation

Mon stage s'est effectué au sein du Service d'Expérimentation et de Développement

Contexte – SensLAB/IoT-LAB

Un réseau de capteurs = des capteurs qui échangent leurs mesures

Application : santé, sport, batiment...

Expérimentation : grande échelle > 12 capteurs, fastidieux

<u>Plateforme Senslab</u>

- Distribuée sur 4 sites distants
- Large échelle 256 noeuds capteurs fixes / site
- Automatique / Ouvert Accès distant

Routage dynamique

Contexte – sujet

Intégration d'un robot Roomba dans ce réseau de capteurs

Autonomie

Fiabilité

Robustesse

Déplacement autonome du robot d'un point A à un point B d'Inria

Le robot

Le robot - Matériel

Chassis

+

Kinect

+

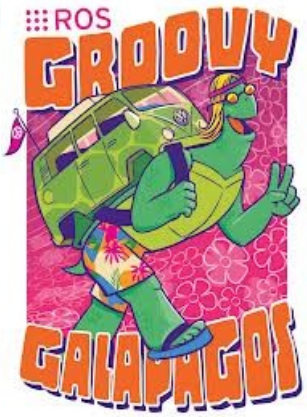
Laptop

+

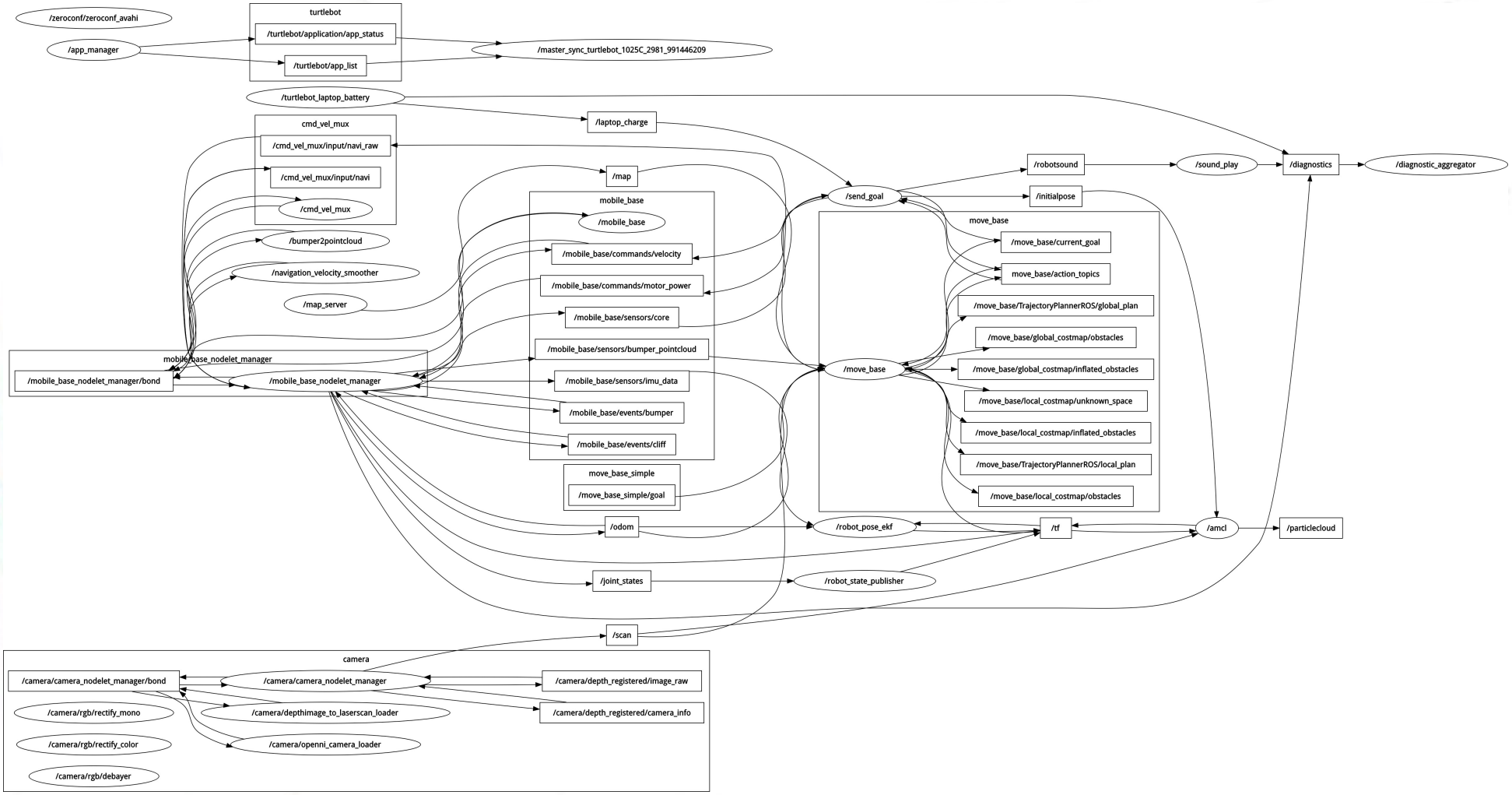
Robot kobuki

Le robot – Outils de départ

- Une première étude d'implémentation avec un Turtlebot 1 et ROS Fuerte
  - Développé par Willow Garage
  - Possibilité d'ajout de packages
  - Permet aux applications robotiques de communiquer entre elles
  - Grande communauté



Passage à ROS Groovy et au Turtlebot 2 (kobuki)


/zeroconf/zeroconf_avahi
/app_manager
turtlebot
/turtlebot/application/app_status
/turtlebot/app_list
/master_sync_turtlebot_1025C_2981_991446209
/turtlebot_laptop_battery
/laptop_charge
cmd_vel_mux
/cmd_vel_mux/input/navi_raw
/cmd_vel_mux/input/navi
/cmd_vel_mux
/bumper2pointcloud
/navigation_velocity_smoother
/map_server
/map
mobile_base
/mobile_base
/mobile_base/commands/velocity
/mobile_base/commands/motor_power
/mobile_base/sensors/core
/mobile_base/sensors/bumper_pointcloud
/mobile_base/sensors/imu_data
/mobile_base/events/bumper
/mobile_base/events/cliff
mobile_base_nodelet_manager
/mobile_base_nodelet_manager/bond
/mobile_base_nodelet_manager
move_base_simple
/move_base_simple/goal
/send_goal
/robotsound
/sound_play
/diagnostics
/diagnostic_aggregator
/initialpose
move_base
/move_base/current_goal
move_base/action_topics
/move_base/TrajectoryPlannerROS/global_plan
/move_base/global_costmap/obstacles
/move_base/global_costmap/inflated_obstacles
/move_base/local_costmap/unknown_space
/move_base/local_costmap/inflated_obstacles
/move_base/TrajectoryPlannerROS/local_plan
/move_base/local_costmap/obstacles
/move_base
/odom
/robot_pose_ekf
/tf
/amcl
/particlecloud
/joint_states
/robot_state_publisher
/scan
camera
/camera/camera_nodelet_manager/bond
/camera/camera_nodelet_manager
/camera/depth_registered/image_raw
/camera/rgb/rectify_mono
/camera/depthimage_to_laserscan_loader
/camera/depth_registered/camera_info
/camera/rgb/rectify_color
/camera/openni_camera_loader
/camera/rgb/debayer

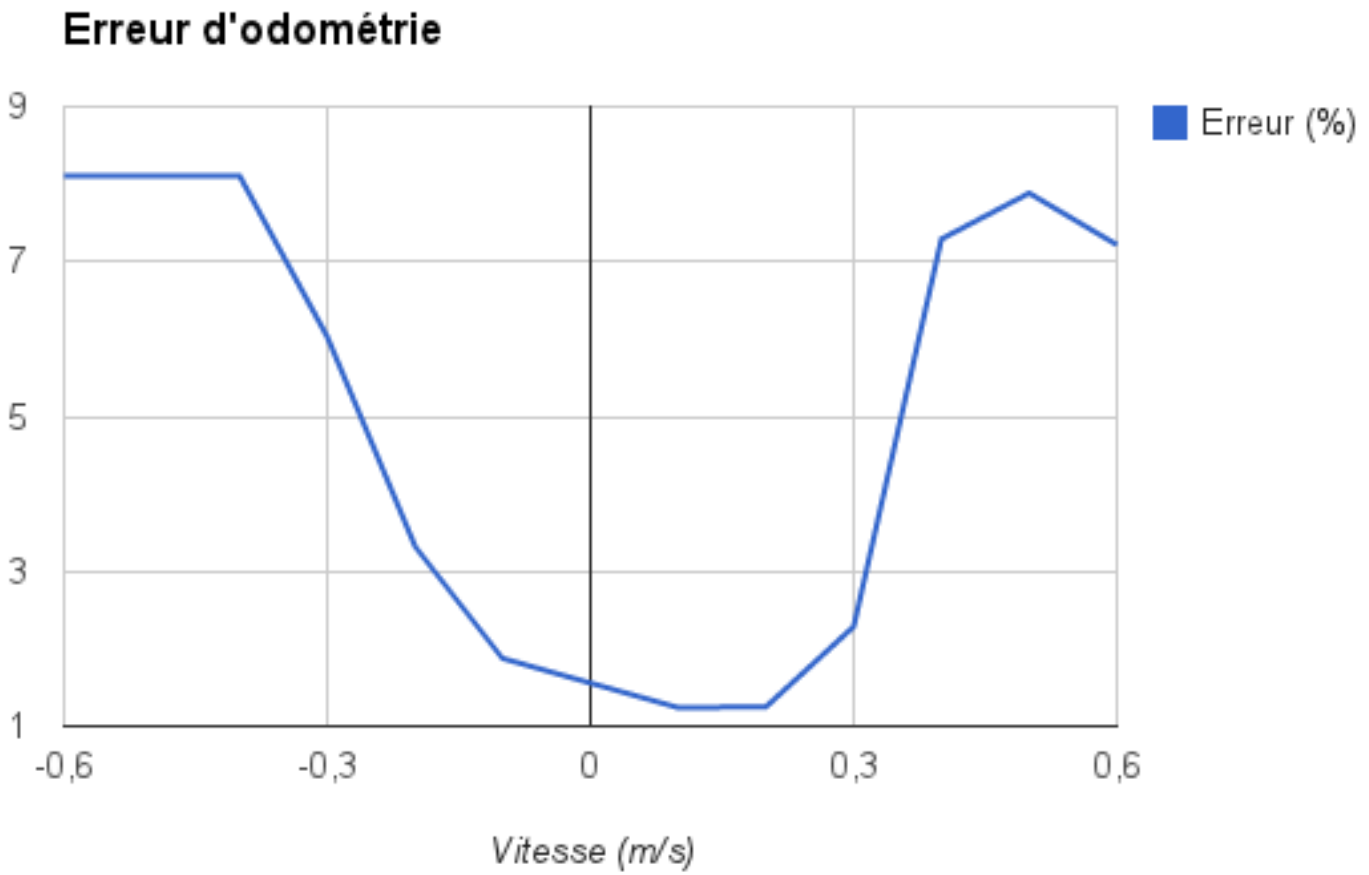
Le robot - Fonctionnalités

Prise en main et évaluation

Tests - Fiabilité

Test de l'Odométrie

Tests - Fiabilité



Tests - Fiabilité

Test de l'Odométrie

Tests - Fiabilité

Test de l'Odométrie

Tests - Fiabilité

Test de l'Odométrie

Test de la Kinect

Tests - Fiabilité

- Champ de vision entre 0.6 et 8 mètres
- Angle de vue H : 57°, V : 43°
- Résolution : 640x480 px

Tests - Fiabilité

Test de l'Odométrie

Test de la Kinect

Tests - Robustesse

Navigation

Méthode :

Utilisation de l'Interface Rviz

Problèmes identifiés :

Non prise en compte des bumpers

Angles mal détectés

Vitesse par défaut trop élevée

Tests - Robustesse

Auto-docking

Méthode :

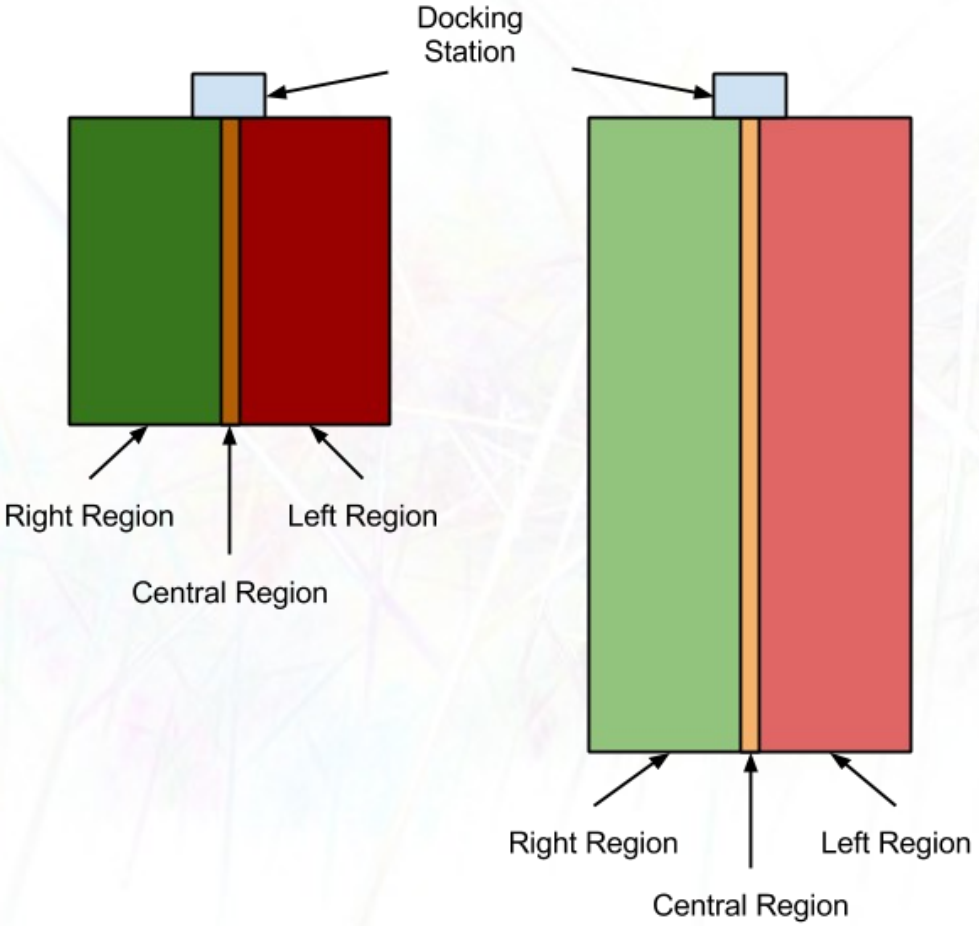
Lecture de la documentation ROS

Script python

Observation du topic où sont publiées les données des capteurs IR → caractérisation du champ IR

Autodocking

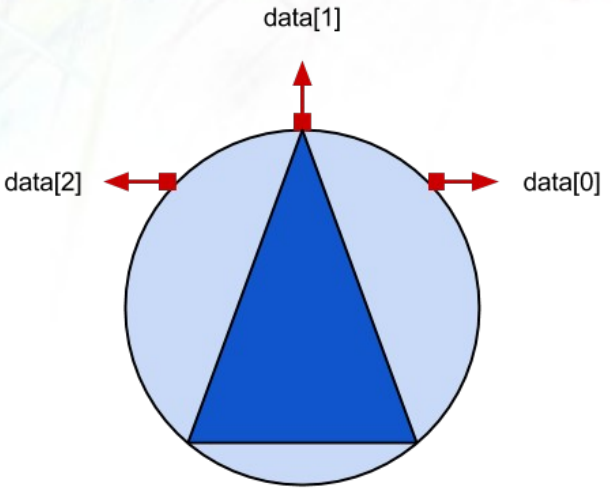
Station de rechargement



Near Field

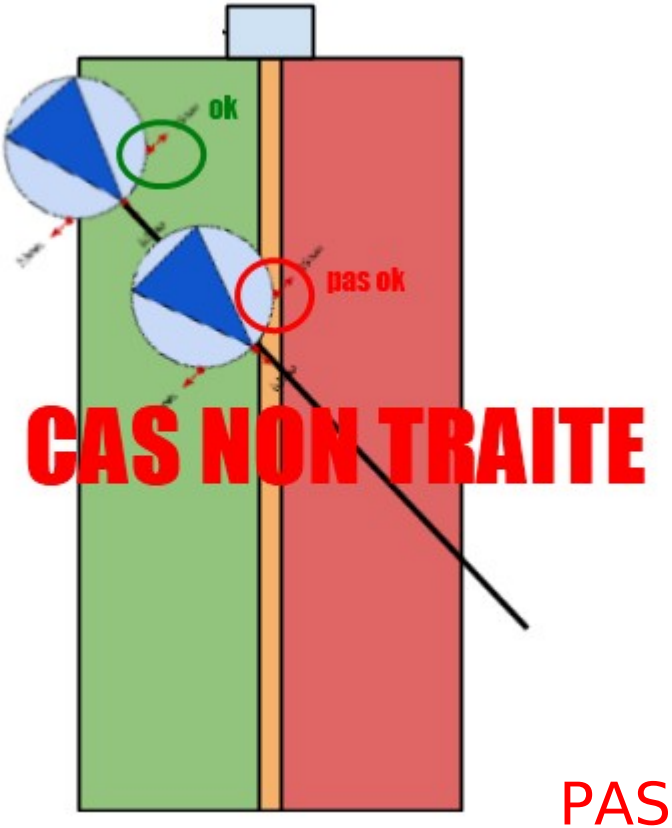
Far Field

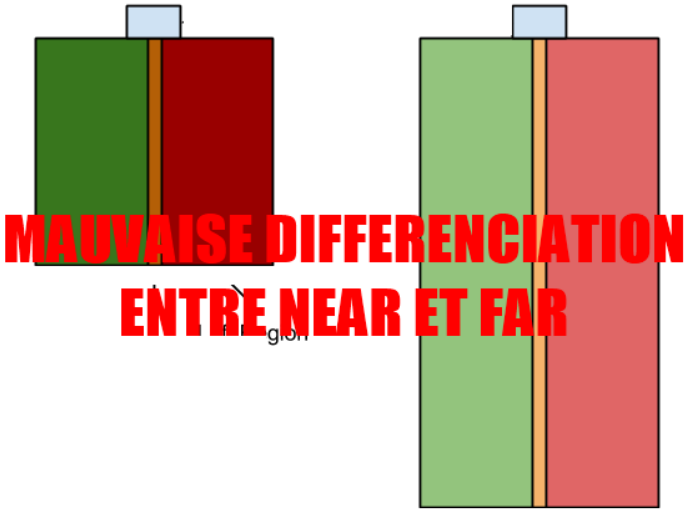
Positionnement des capteurs du robot



Autodocking

VITESSE D'APPROCHE GLOBALEMENT TROP ELEVEE





PAS DE GESTION DES ERREURS

Tests - Robustesse

Auto-docking

Méthode :

Script python

Problèmes identifiés :

Vitesse d'approche non adaptée

Cas d'erreur non traité

Cas particuliers non traités

Implémentation

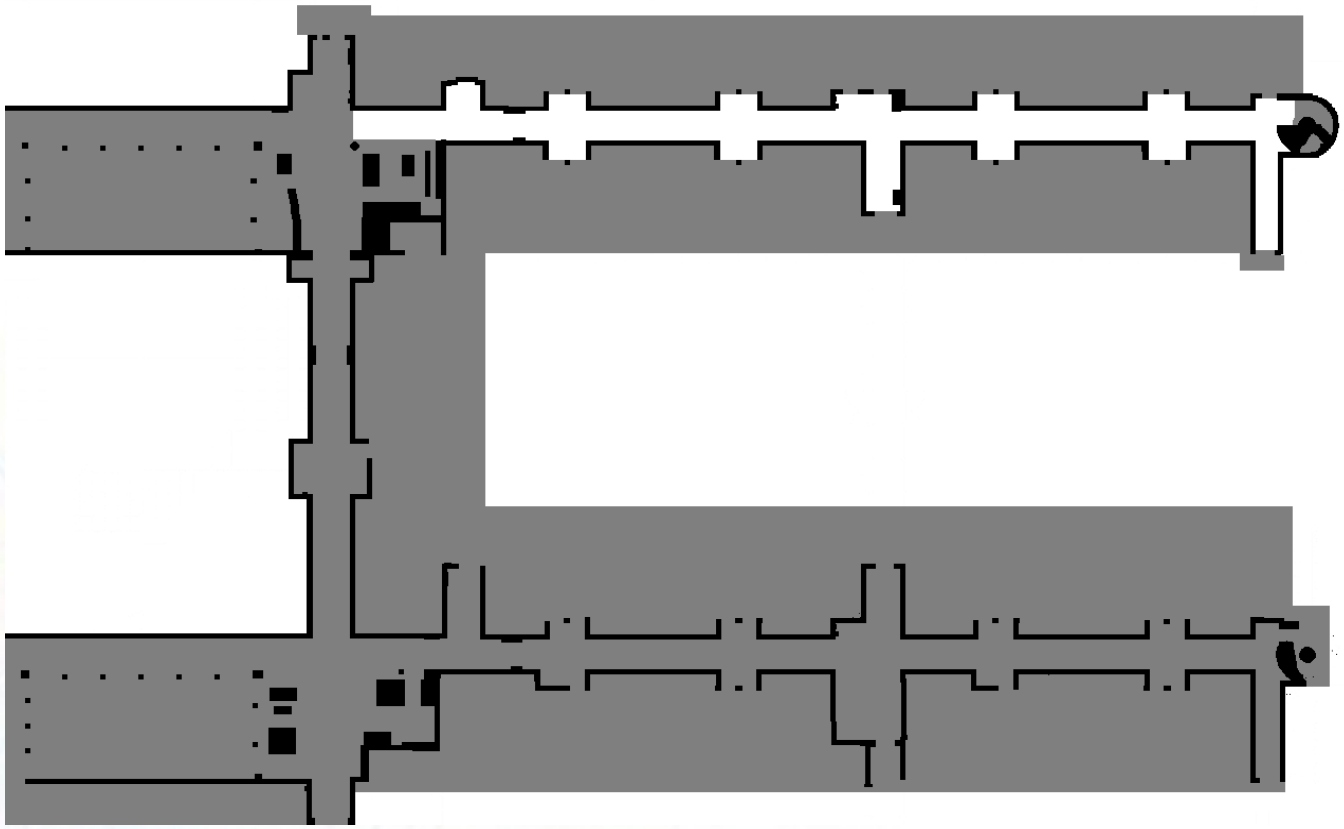
Implémentation : Correction autodocking

Code disponible sur github

- Réduction de la vitesse
- Ajout de cas supplémentaires
- Ajout de la gestion des erreurs
- Correction de bug avec les développeurs de ROS

Tests supplémentaires en cours à Strasbourg avant contribution au github de ROS

Implémentation : Navigation



- mise à jour de la carte
- paramétrage du node move_base (vitesses, gestion des obstacles, capteurs surveillés...)

Implémentation : Navigation

- création d'une carte et de son fichier de configuration pour map_server
- envoi de la coordonnée à atteindre au node move_base

- surveillance des données issues des capteurs (callback)
- si besoin, backup
- surveillance des niveaux des batteries
- si besoin retour à la station de rechargement

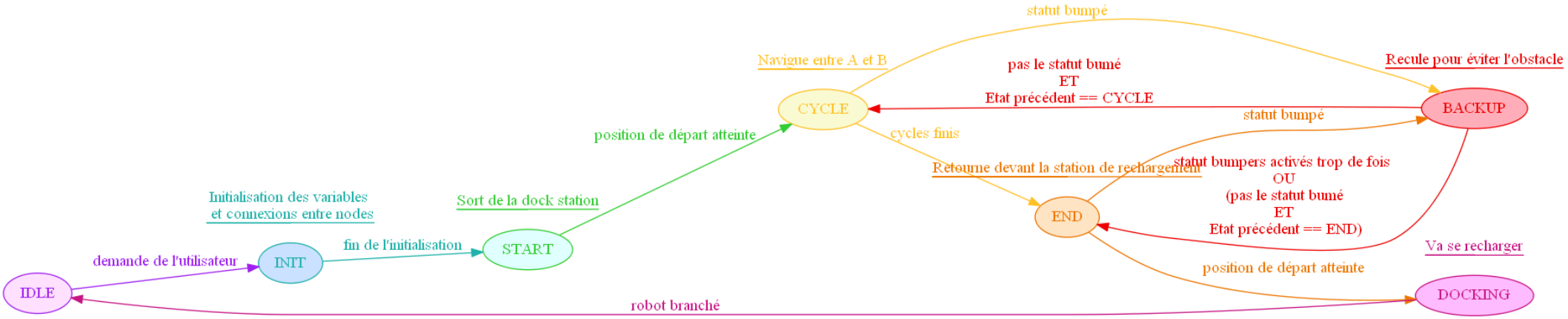
- move_base annonce avoir atteint le but
- nouvelles coordonnées
- on répète tant que le nombre de trajets n'est pas effectué

- envoie des coordonnées de la station de rechargement à move_base

- demande de docking à kobuki_auto_docking

- kobuki_auto_docking annonce avoir atteint le but → arrêt, sinon → erreur

Implémentation : Navigation



Résultats et perspectives d'évolution

Résultats et perspectives d'évolution

Vidéo

Résultats et perspectives d'évolution

- Scénario implémenté et robot autonome
- Sauvegarde des évènements dans un fichier log à implémenter
- Portabilité possible vers ROS Hydro
- Le travail réalisé servira de base pour l'ensemble des robots des plateformes SensLAB

Conclusion

- Découverte de la robotique
- Découverte du C++ et apprentissage du Python
- Approfondissement de mes connaissances sur Linux
- Contribution à la communauté de ROS

Merci pour votre attention