

Projet R&D

SpyPolybot

Présenté par Hugo DENIS--MARTIN et Brice MABILLE



Sommaire

Introduction

Le Robot

GPS - Kalman

LIDAR

Perspective : Camera - IA embarquée

Conclusions



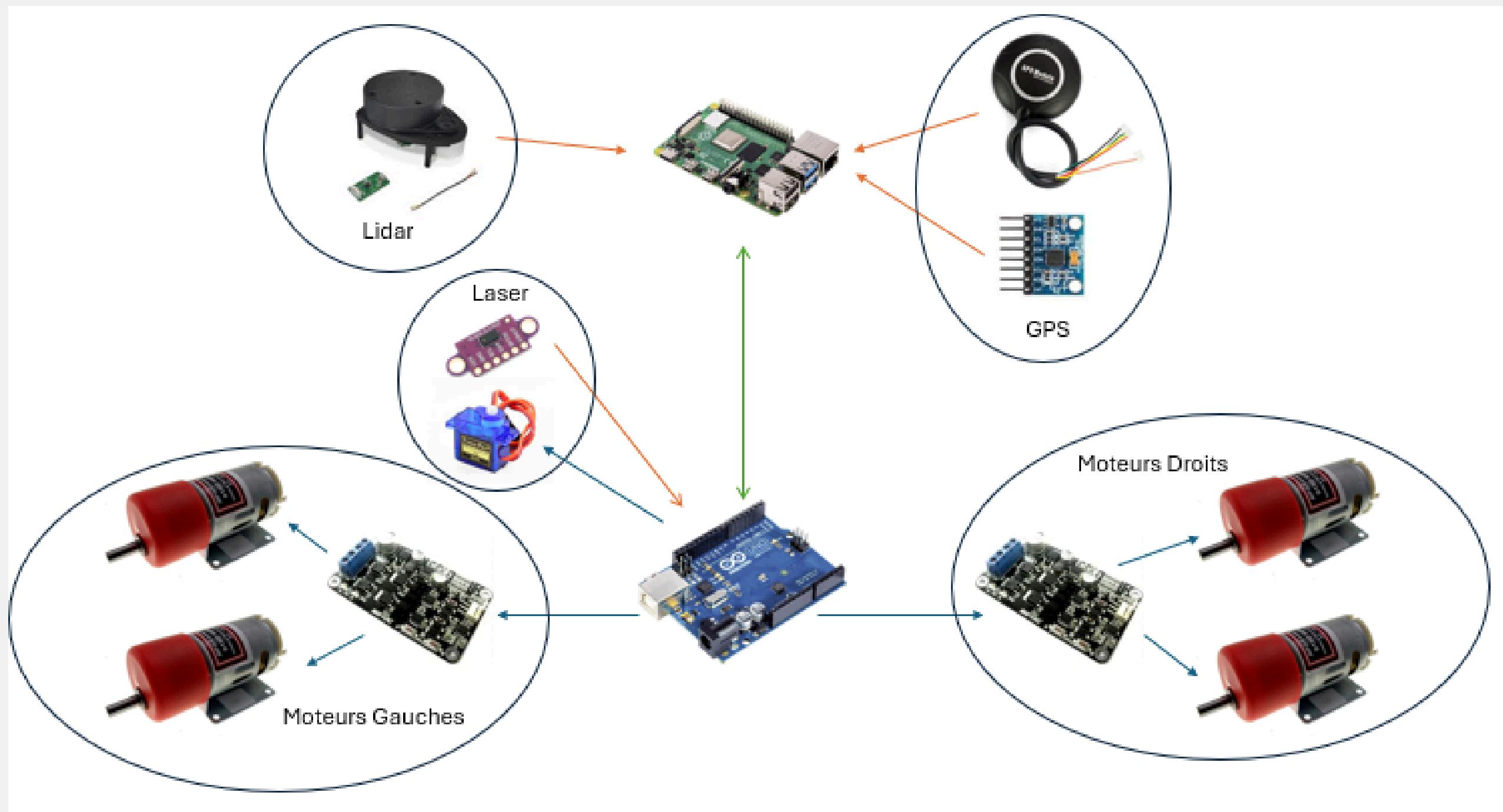
Introduction



Pour notre projet cette année, nous avons travaillé sur l'amélioration du robot BeachPolybot pour lui donner un objectif de cartographie et de surveillance d'environnements.

Notre objectif était d'intégrer des capteurs pour accroître son autonomie et sa précision dont un GPS associé à un filtre de Kalman pour une localisation en temps réel précise, ainsi qu'un LIDAR pour détecter les obstacles et cartographier l'environnement.

Le Robot



Partie Commande:

Carte Arduino
Carte Raspberry pi 4

Partie Laser:

Laser VL53L0
Servomoteur SG90

Partie Positionnement:

GPS Ublox 7M
Accéléromètre mpu-6050

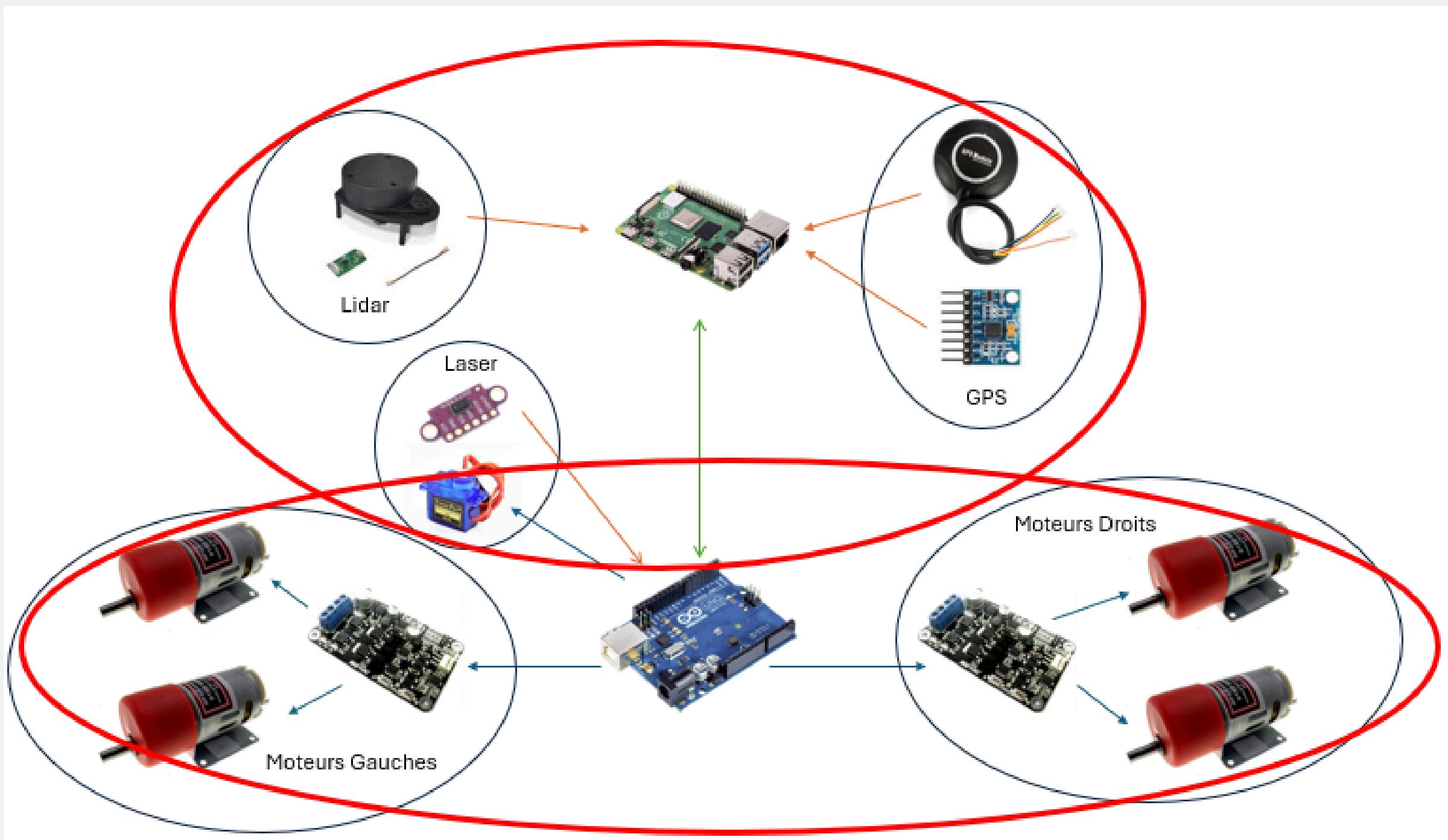
Partie Lidar:

RPLidar A1

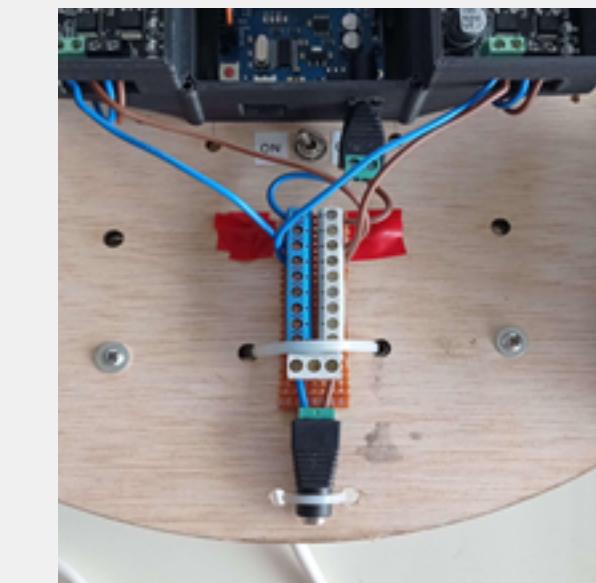
Partie Puissance:

Carte de puissance cyt-132 (x2)
Moteurs 919D (x4)

Le Robot

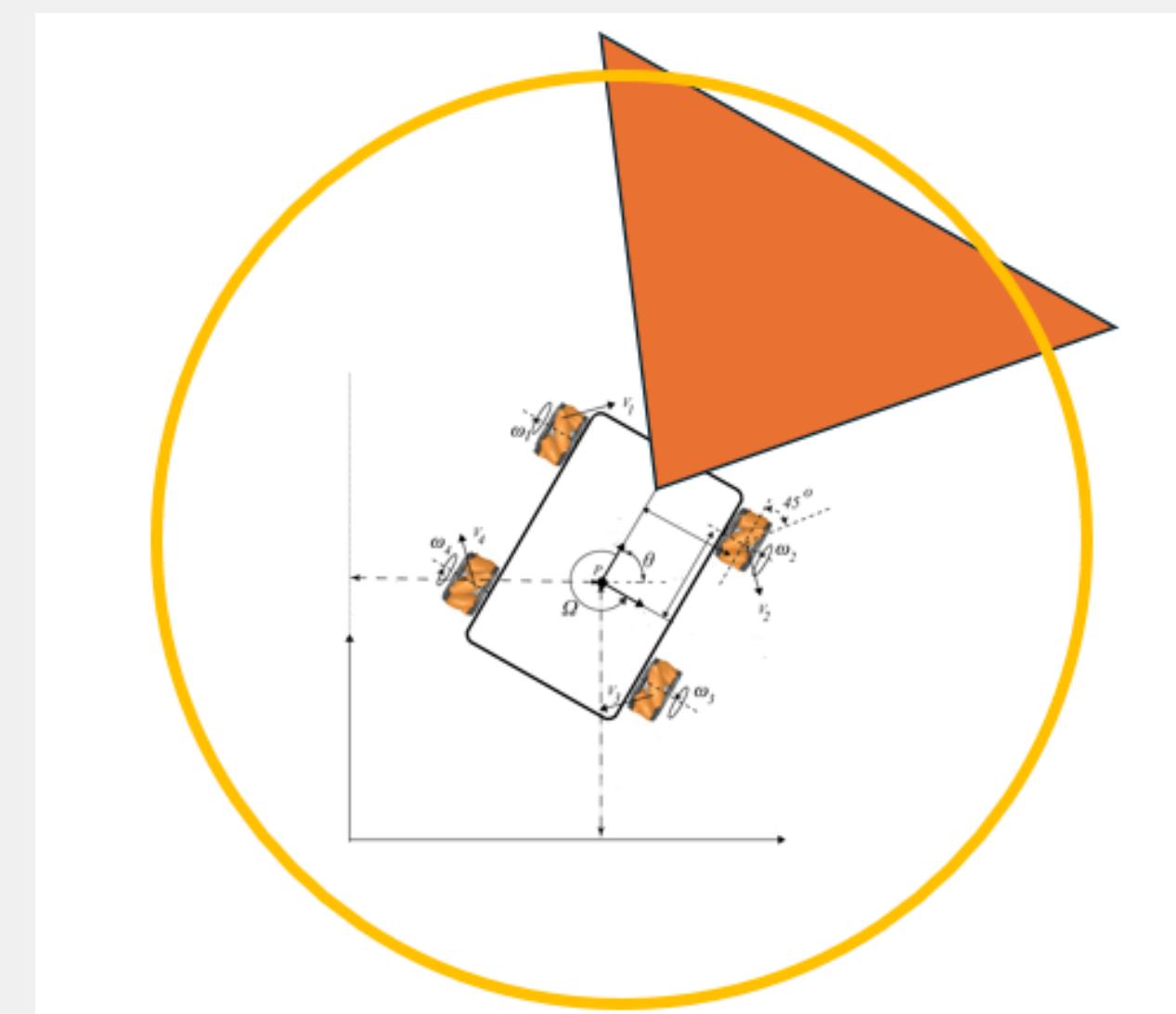
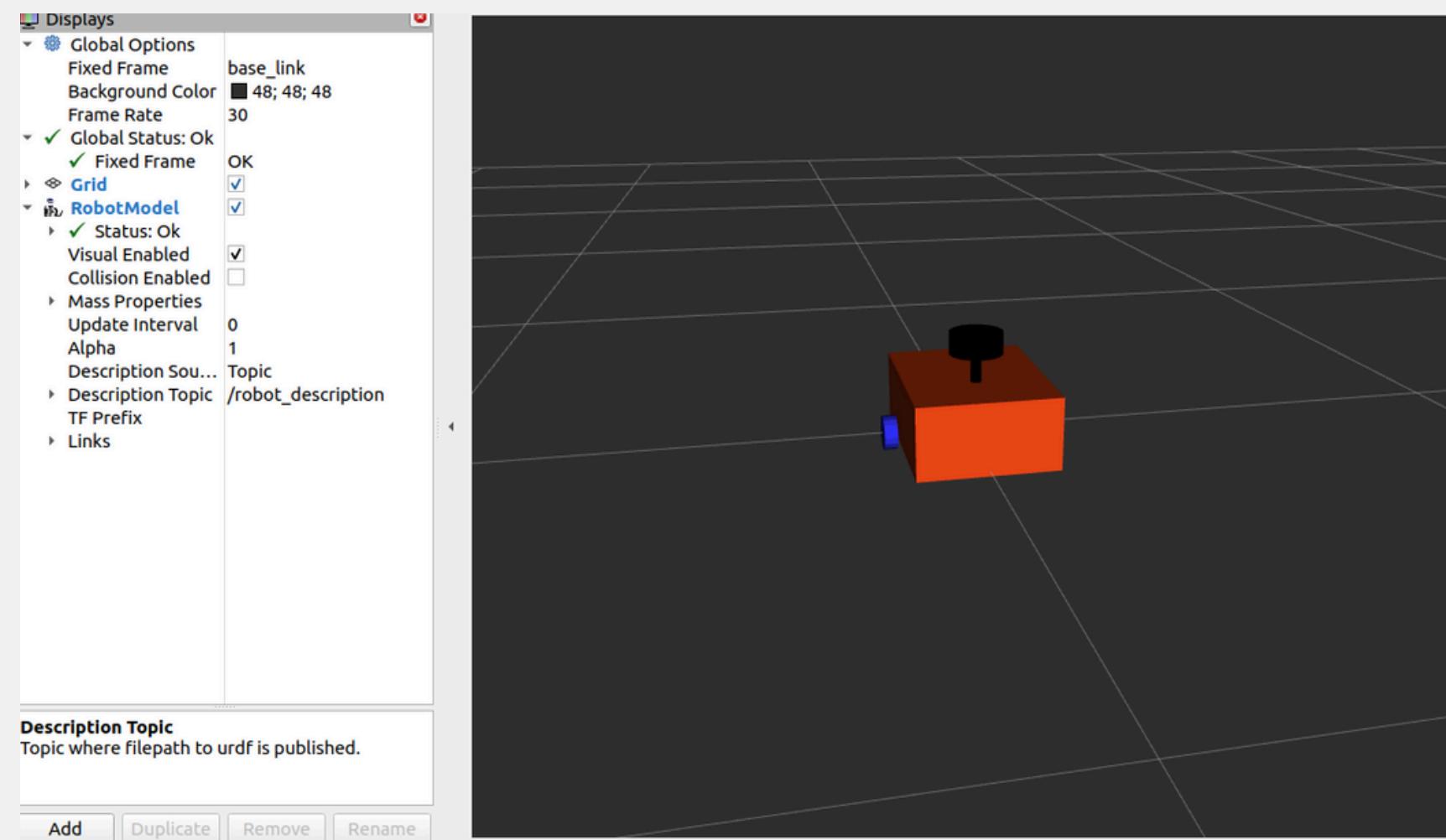


Circuit Commande
5V

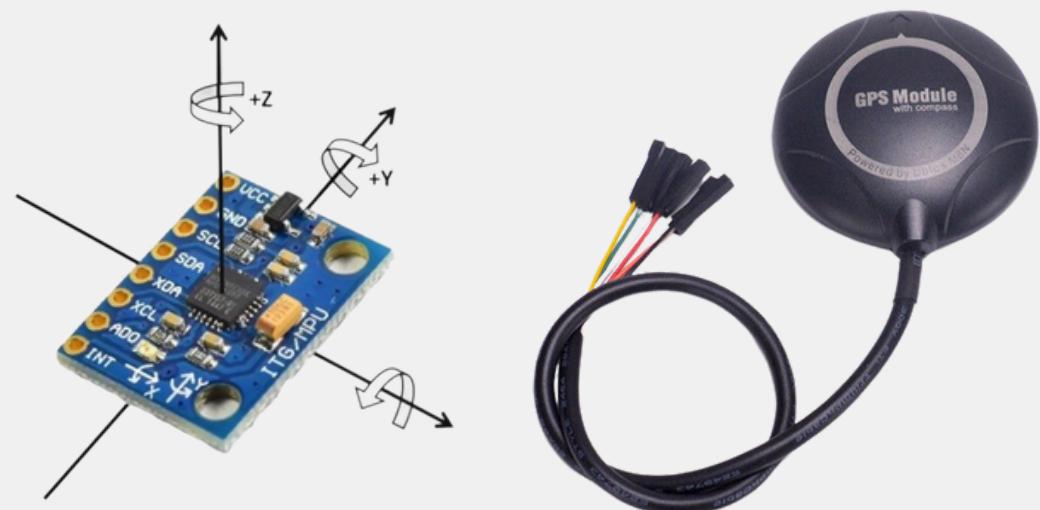


Circuit Puissance
12V

Le Robot



GPS - Filtre de Kalman



VECTEUR D'ÉTAT DU FILTRE
DE KALMAN

GPS UBLOX 7M pour une localisation du robot en temps réel
pour améliorer l'autonomie et la navigation du robot
(incertitude = 2m)

$$X_k = \begin{bmatrix} x \\ v \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

x position statique à un instant
 v vitesse dynamique à un instant

$$v_k = v_{k-1} + a_k \Delta t$$

LIEN ENTRE POSITION ET VITESSE :

$$x_k = x_{k-1} + v_k \Delta t$$

GPS - Filtre de Kalman



Etape d'extrapolation = anticipation de l'état futur

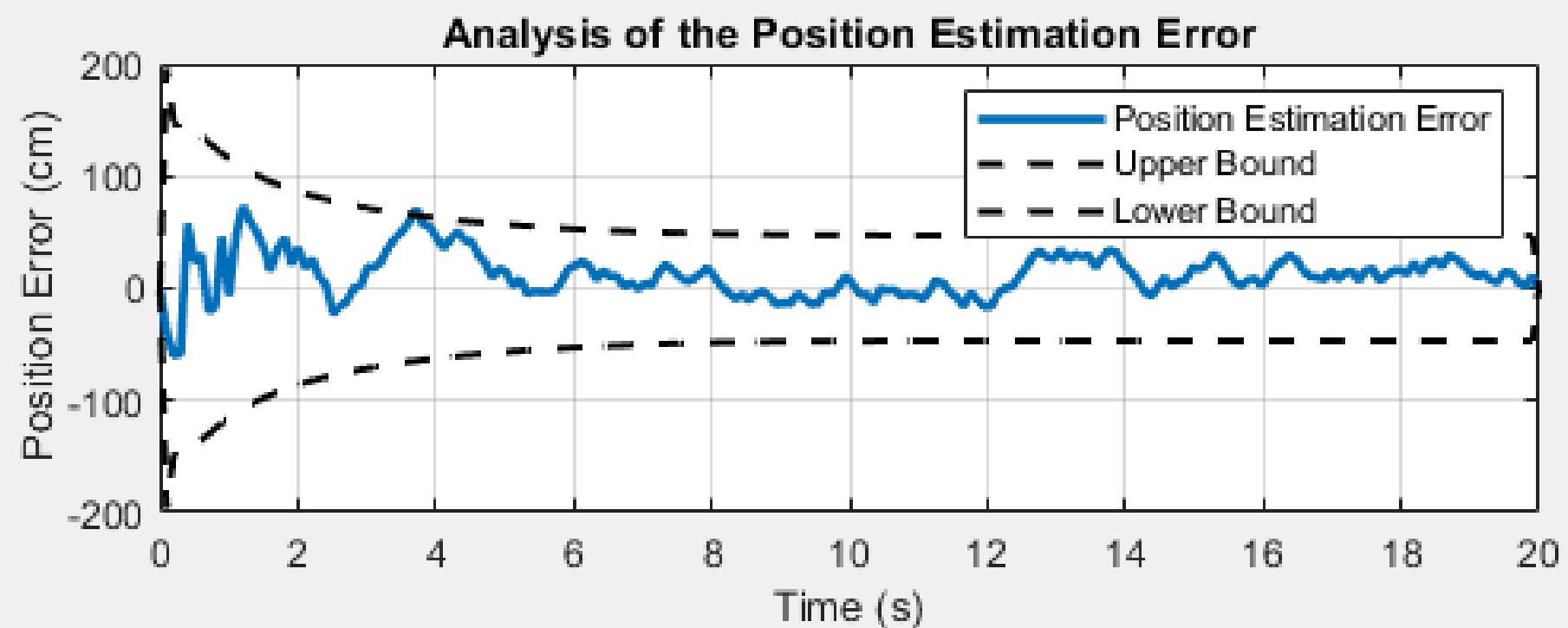
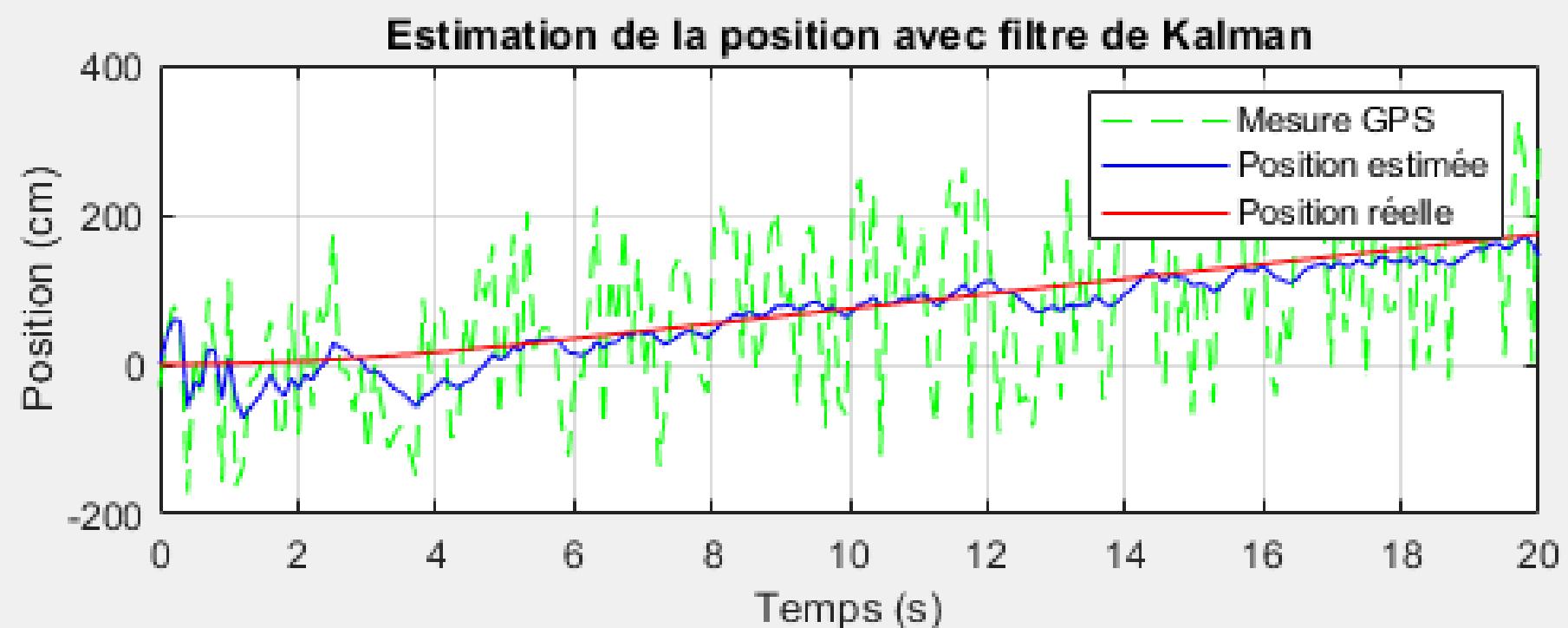
$$X_{k+1} = \Phi_d X_k + G_d u_k$$

$$u_k = a_k = \ddot{x}_{MPU_acc}$$

$$\Phi_d = \begin{bmatrix} 1 & 0,1 \\ 0 & 1 \end{bmatrix} = I + FT_s ; \quad F = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$G_d = \left(\int_0^{T_s} \Phi(t) dt \right) G = \begin{bmatrix} -\frac{T_s^2}{2} \\ -T_s \end{bmatrix} = \begin{bmatrix} -0,005 \\ -0,1 \end{bmatrix} ; \quad G = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Les tests montrent une réduction de l'incertitude de localisation.



LIDAR



Théorie

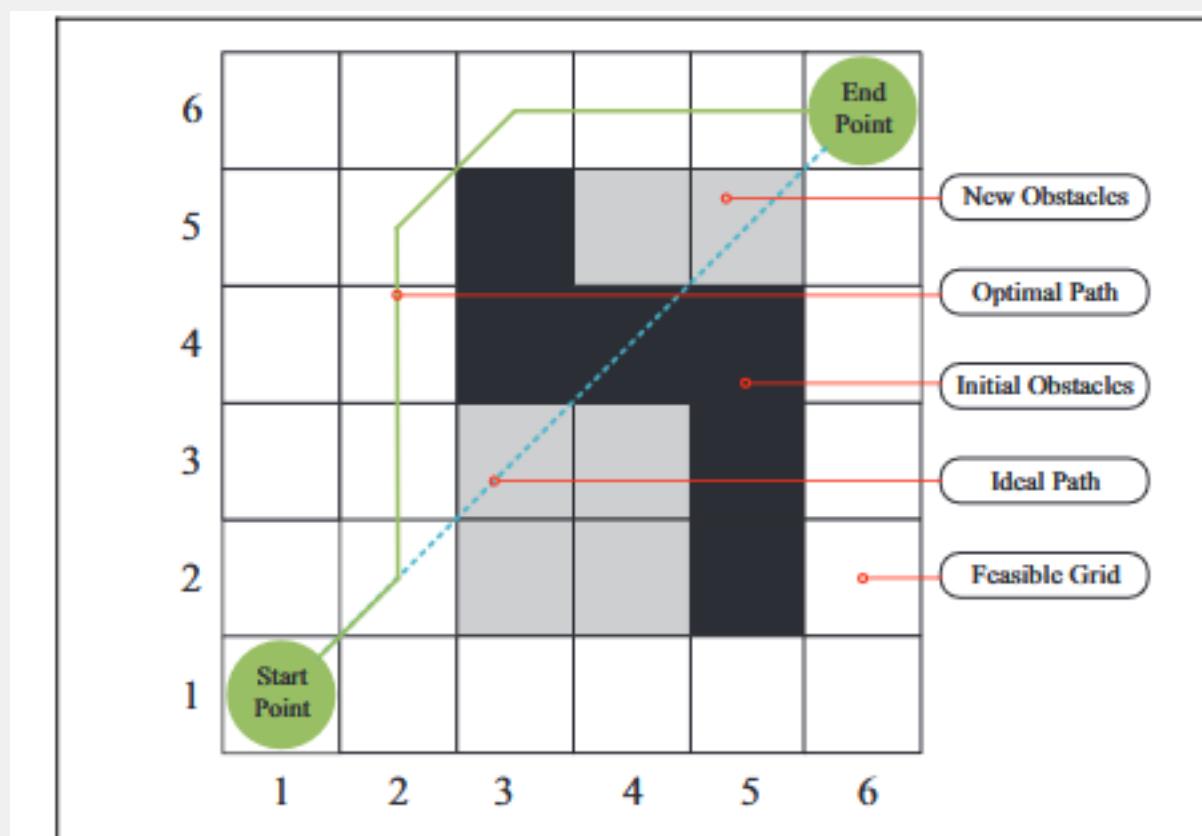
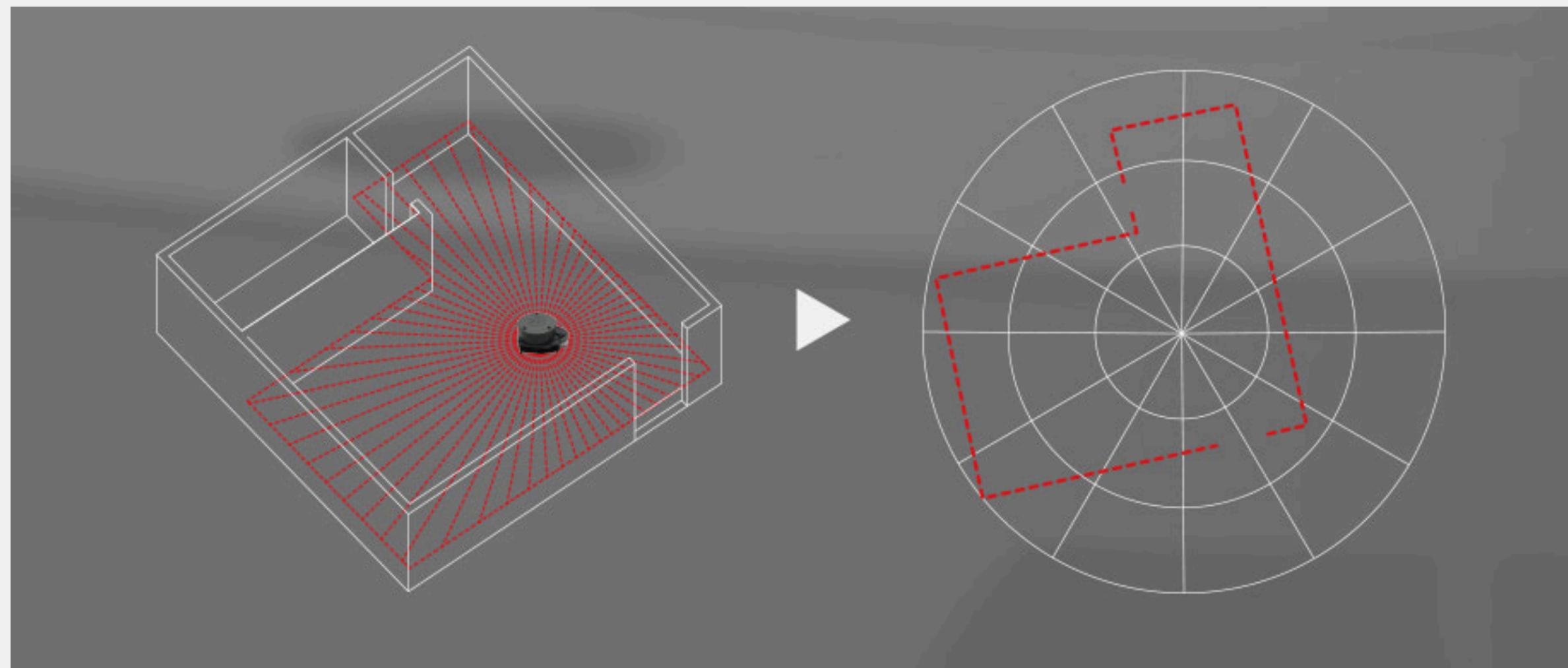
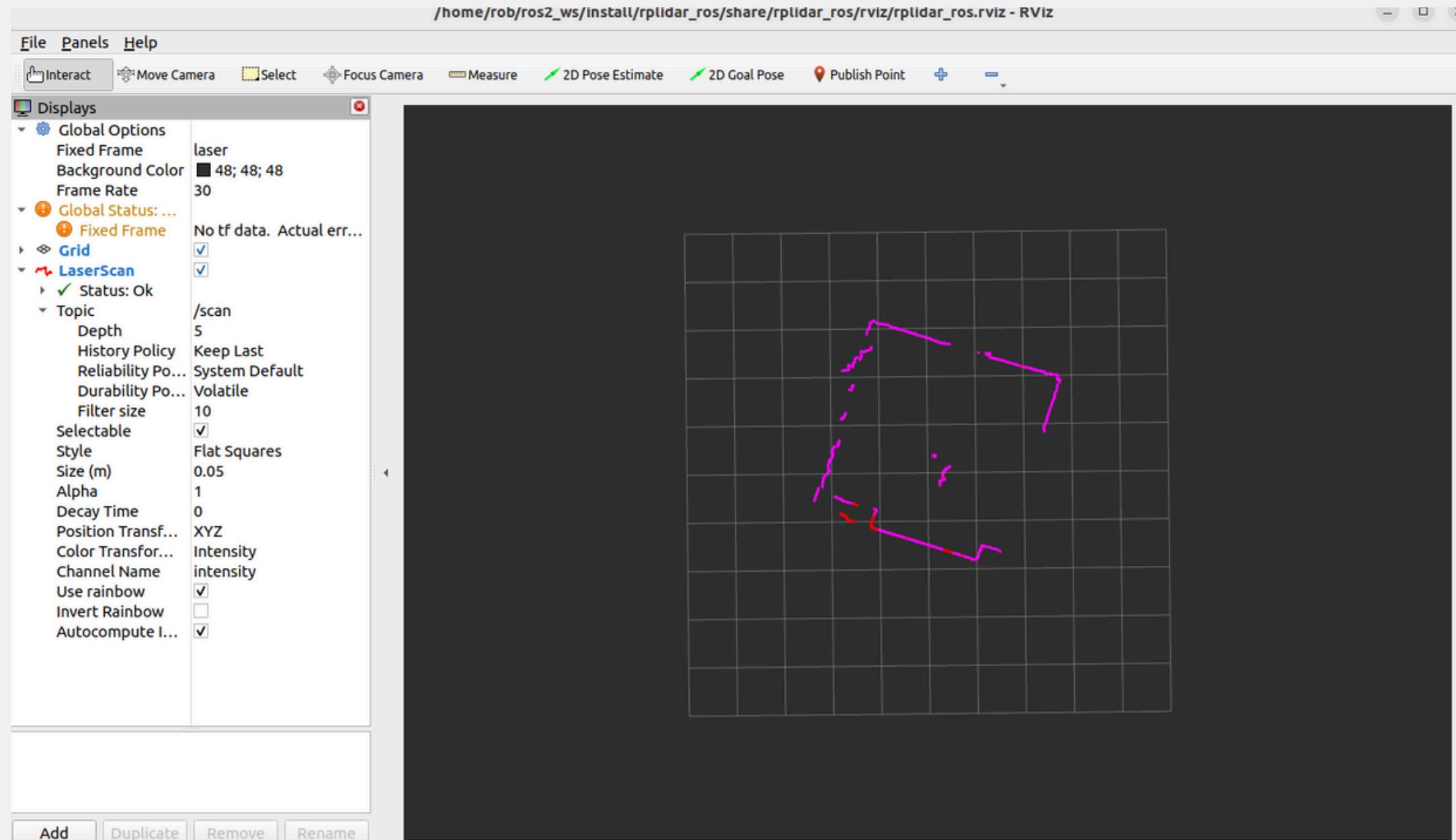


Figure 2. Schematic diagram of obstacle handling.

LIDAR



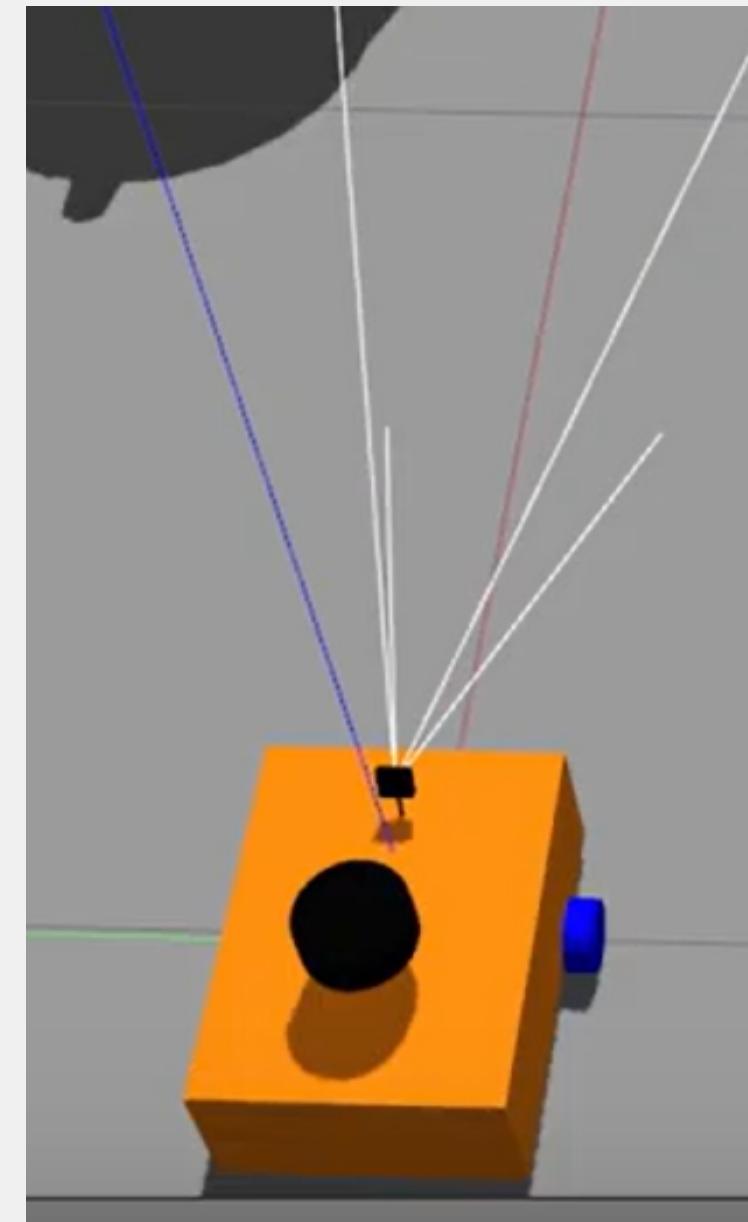
Cartographie de l'environnement du robot



Perspective : Camera - IA embarquée



Installation caméra et
implémentation d'un
algorithme de
reconnaissance d'objets
avec opencv



Conclusions



Ce qui a été fait:

- Robot implémenté dans ROS2
- LiDAR fonctionnel
- GPS fonctionnel

Ce qu'il reste à faire:

- Implémentation caméra
- Fusionner tout les capteurs sur ROS
- Algorithme SLAM