

EI VAC - Groupe 1

Lola COURTY, Antoine DELEGUE, Kevin HUANG, Yann LOCHET, Evans PERRET,
Aude VOCANSON

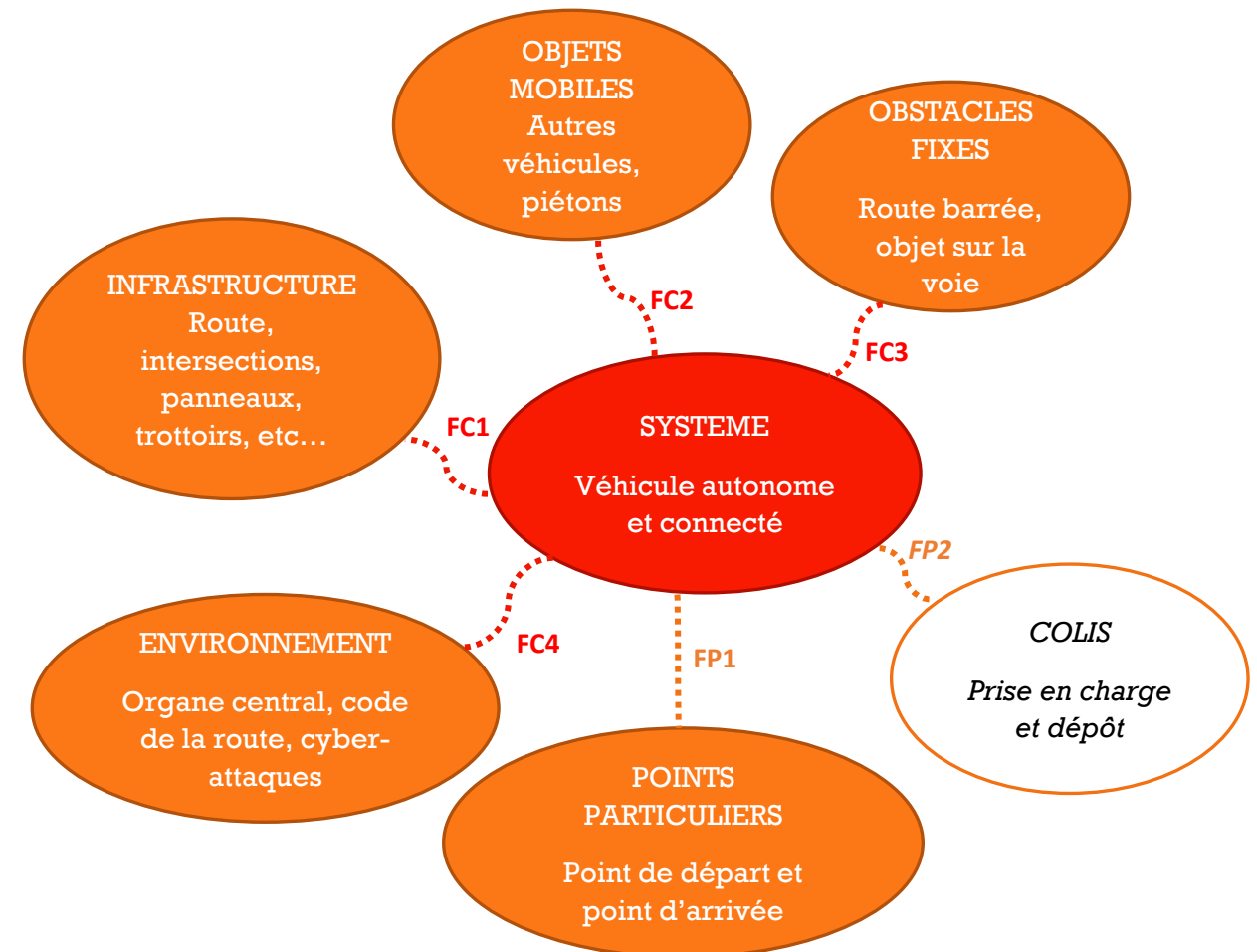
Analyse fonctionnelle du besoin

Fonctions principales

- FP1 - Arriver à la bonne destination en toute sécurité
- FP2 - Colis livré

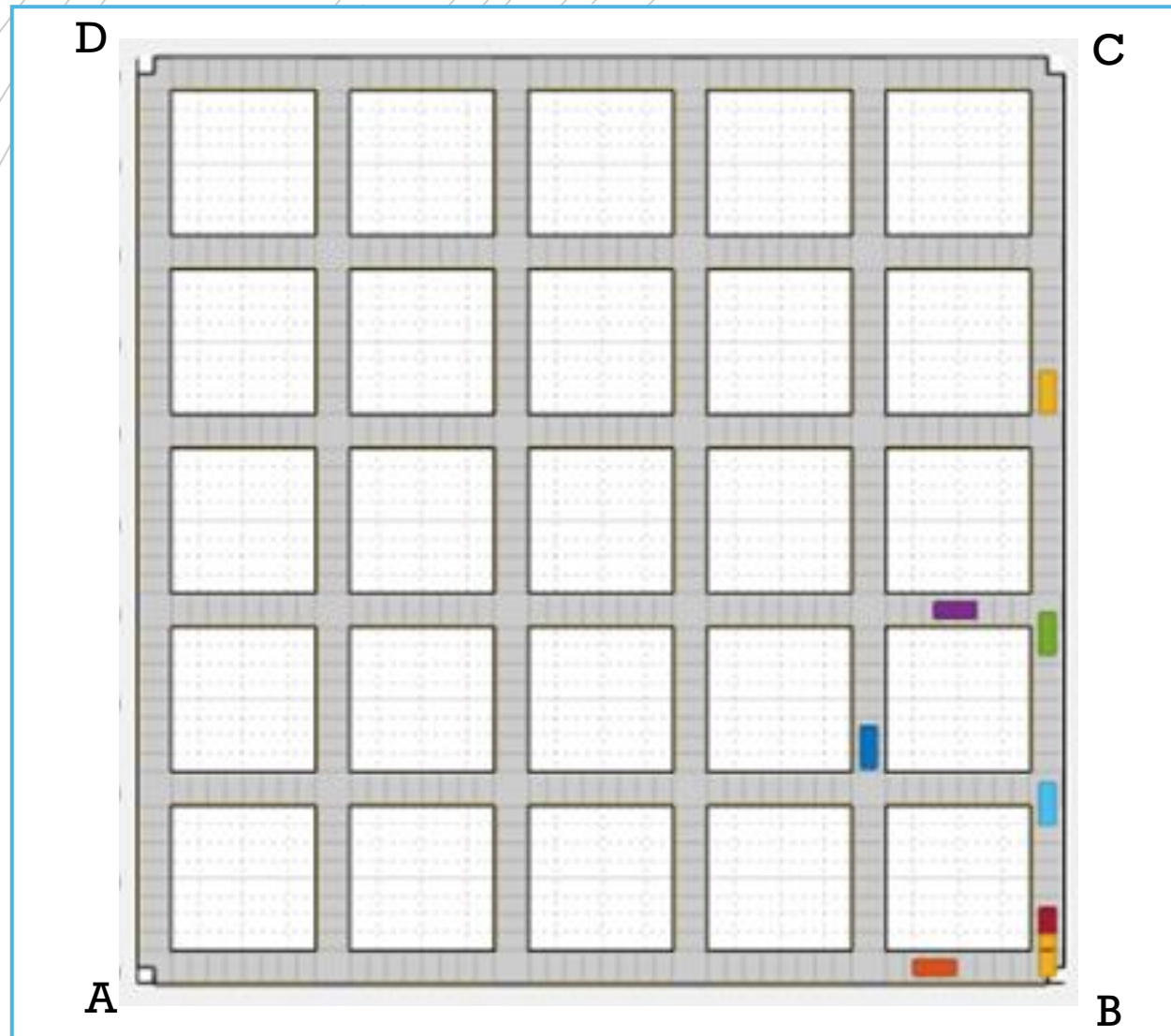
Fonctions de contrainte

- FC1 - Respecter les contraintes imposées par l'environnement (suivre la route)
- FC2 - Gérer les objets mobiles
- FC3 - Gérer les obstacles fixes
- FC4 - Respecter le code de la route



SIMULATION D'UN SCÉNARIO DE LIVRAISON URBAINE

- **Objectif** : conception d'un système de livraison multi-robots dans un environnement complexe
- **Scénario** : chemins A -> C -> A et B-> D -> B
- **3 paramètres** : vitesse V, nombre de voitures N, T temps d'échantillonnage
- **Deux solutions possibles** : centralisée et distribuée

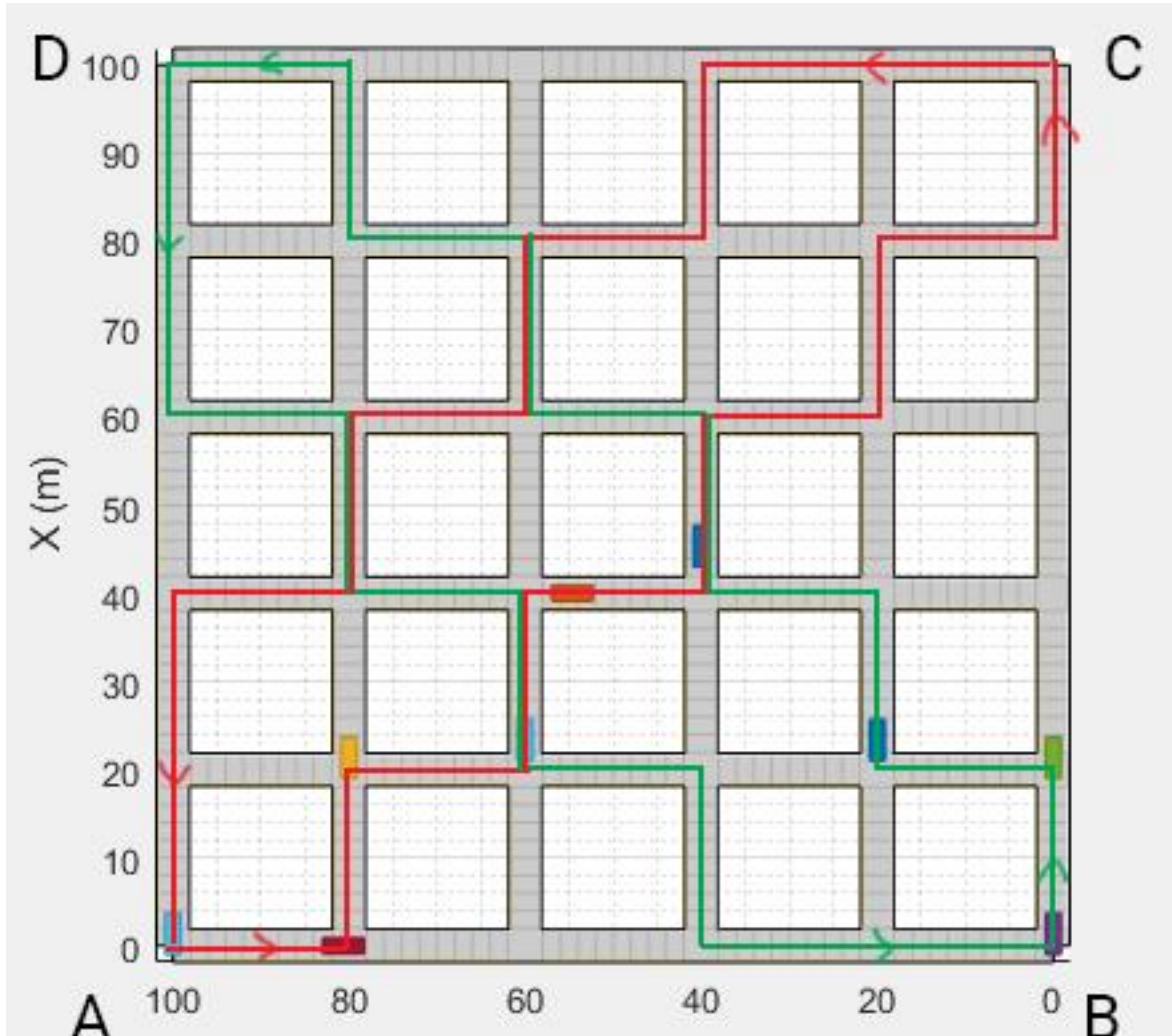


1ère piste : parcours de graphe (centralisée)

- Quadrillage en tant que graphe, les intersections sont les nœuds
- Parcours testés : Dijkstra, A*, DFS
- Avantages :
 - chemin optimal calculé donc gain de temps
 - applicable pour n'importe quel point de départ ou d'arrivée
- Inconvénient : si densité de robot trop élevée ($N > 5$), l'algorithme ne marche plus.

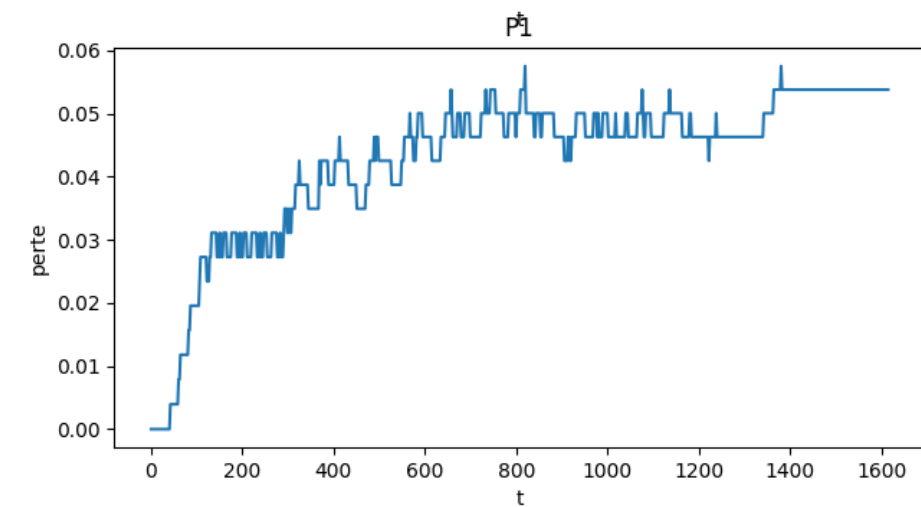
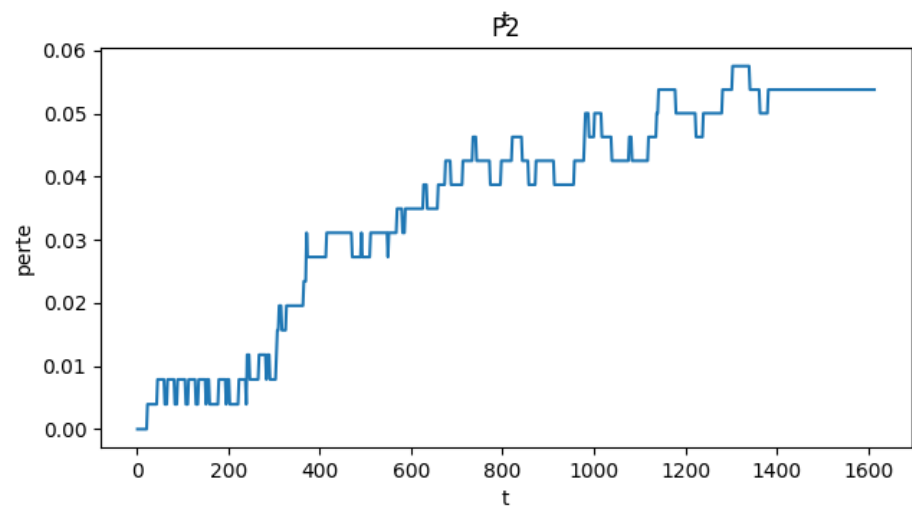
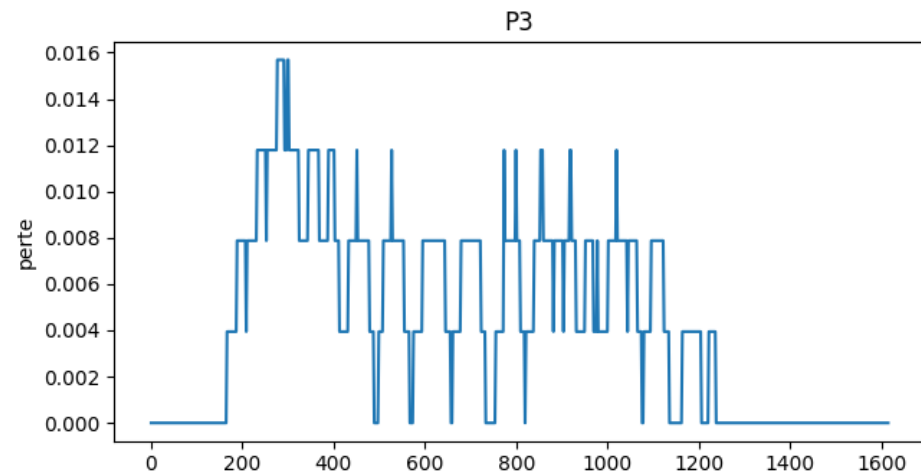
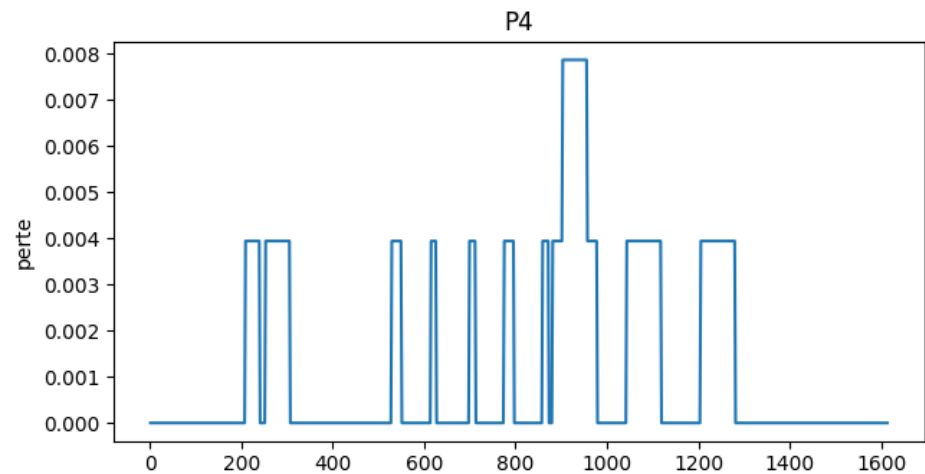
DEMONSTRATION

5



2ème piste : un seul chemin (centralisée)

- Chemin invariable prédéterminé
- Priorité par ordre de départ
- Distance de sécurité : $2,5 \cdot v \cdot T$
- Si risque de collisions, arrêt des deux voitures immédiat : la moins prioritaire recule
- Pour éviter les bouchons, si une voiture est arrêtée depuis longtemps, elle recule petit à petit pour débloquer la situation



$V=2, T=1, N=30$

Taux de perte



Robot

Architecture de contrôle

**Serveur
central**

Raspberry

Arduino

Architecture de contrôle

Serveur central

- **Pathfinding**
- **Gestion des obstacles**
- **Tooling de debug**

Raspberry

- **Gestion de la position**
- **Gestion des collisions**
- **Tooling de debug**

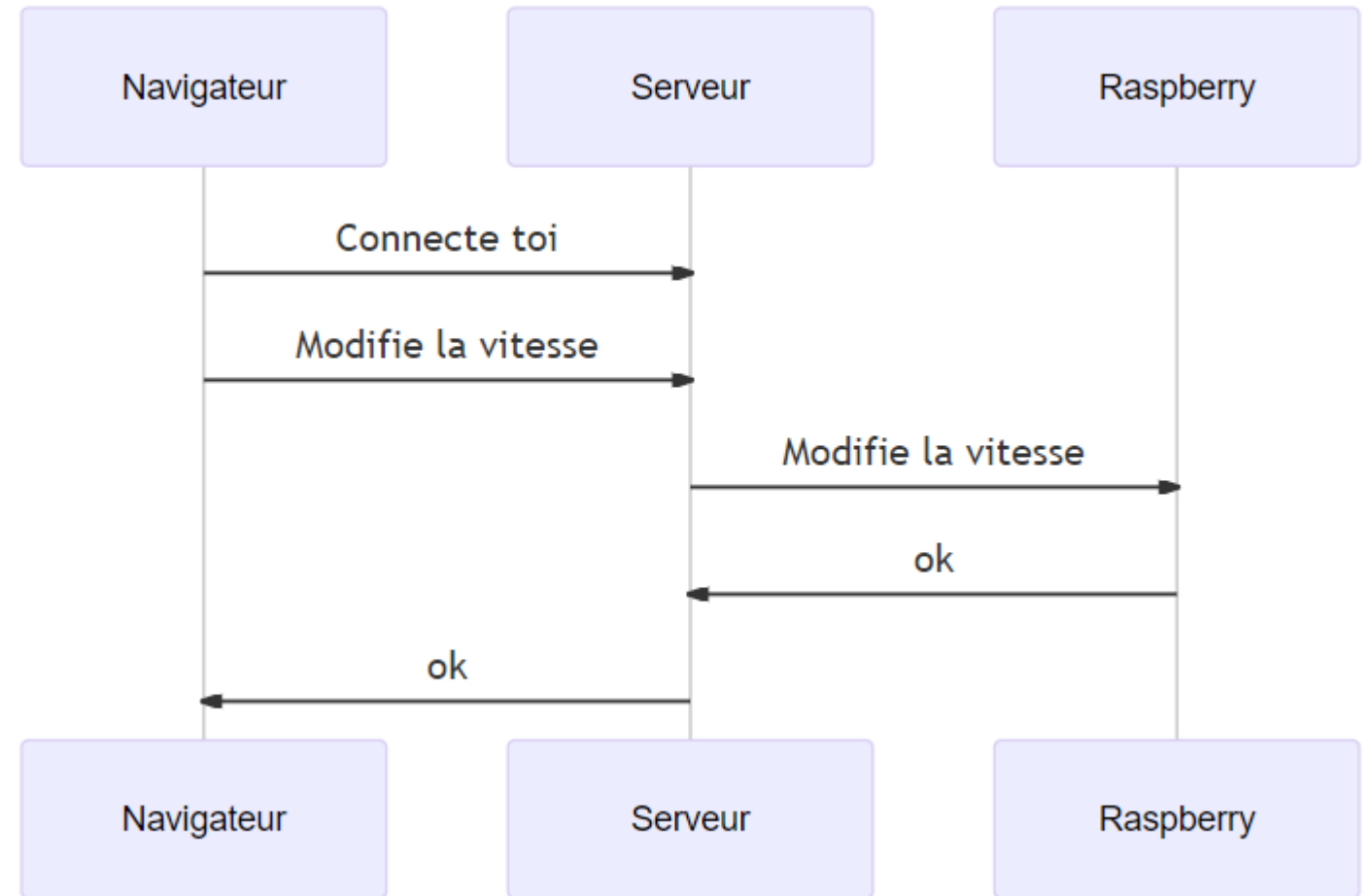
Arduino

- **Feedback des mouvements**
- **Feedback des capteurs**

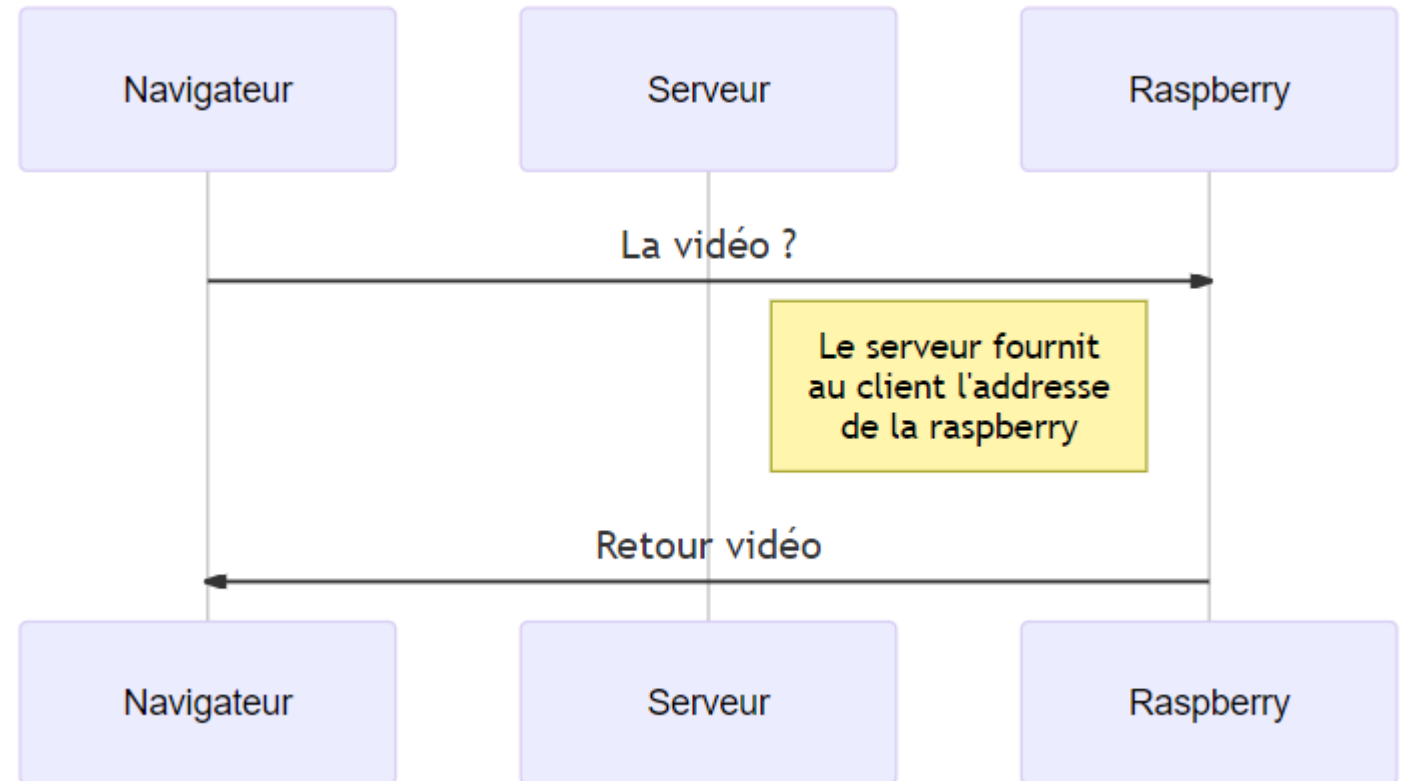
Recherche de chemin

- Reprise du travail de simulation
- Recherche de chemin optimal par Dijkstra
- Quand un obstacle est détecté, calcul d'un nouveau chemin avec suppression du point de l'obstacle dans le graphe
- Par défaut, tous les obstacles ont un temps de vie (obstacles mobiles)

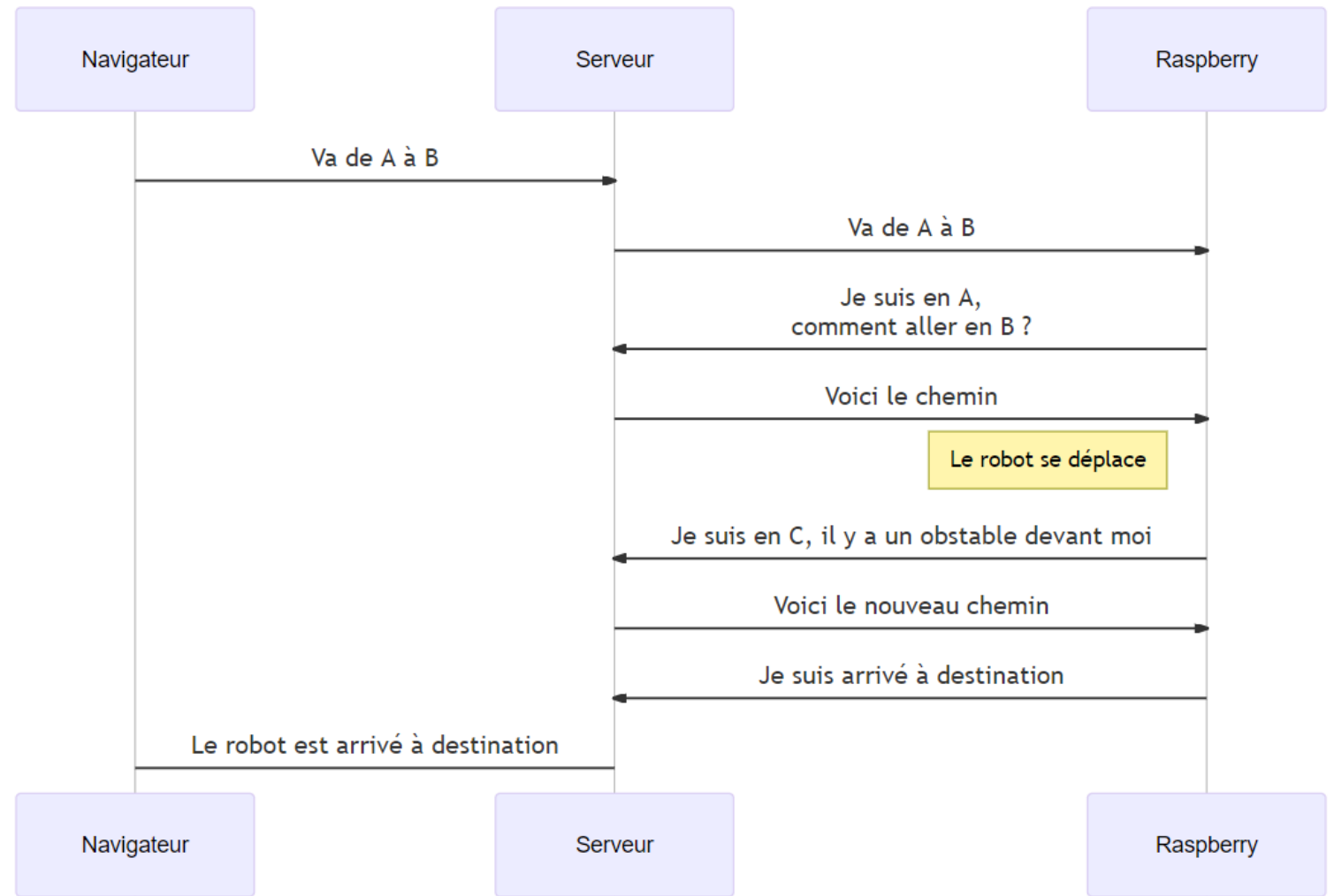
Interface de contrôle



Retour vidéo



Contrôle automatique



IP: 192.168.137.127:5000

192.168.137.127:5000

CONNECT

Status: Idle

Behavior mode: 0

Follow grid (0) ▾

UPDATE

Going to: -1, -1

South ▾

Way Forward ▾

LET'S GO

Control parameters

Speed: 100

100

UPDATE

Camera status

Enabled ▾

Manual precision: (in seconds, higher = less precise) 0.5

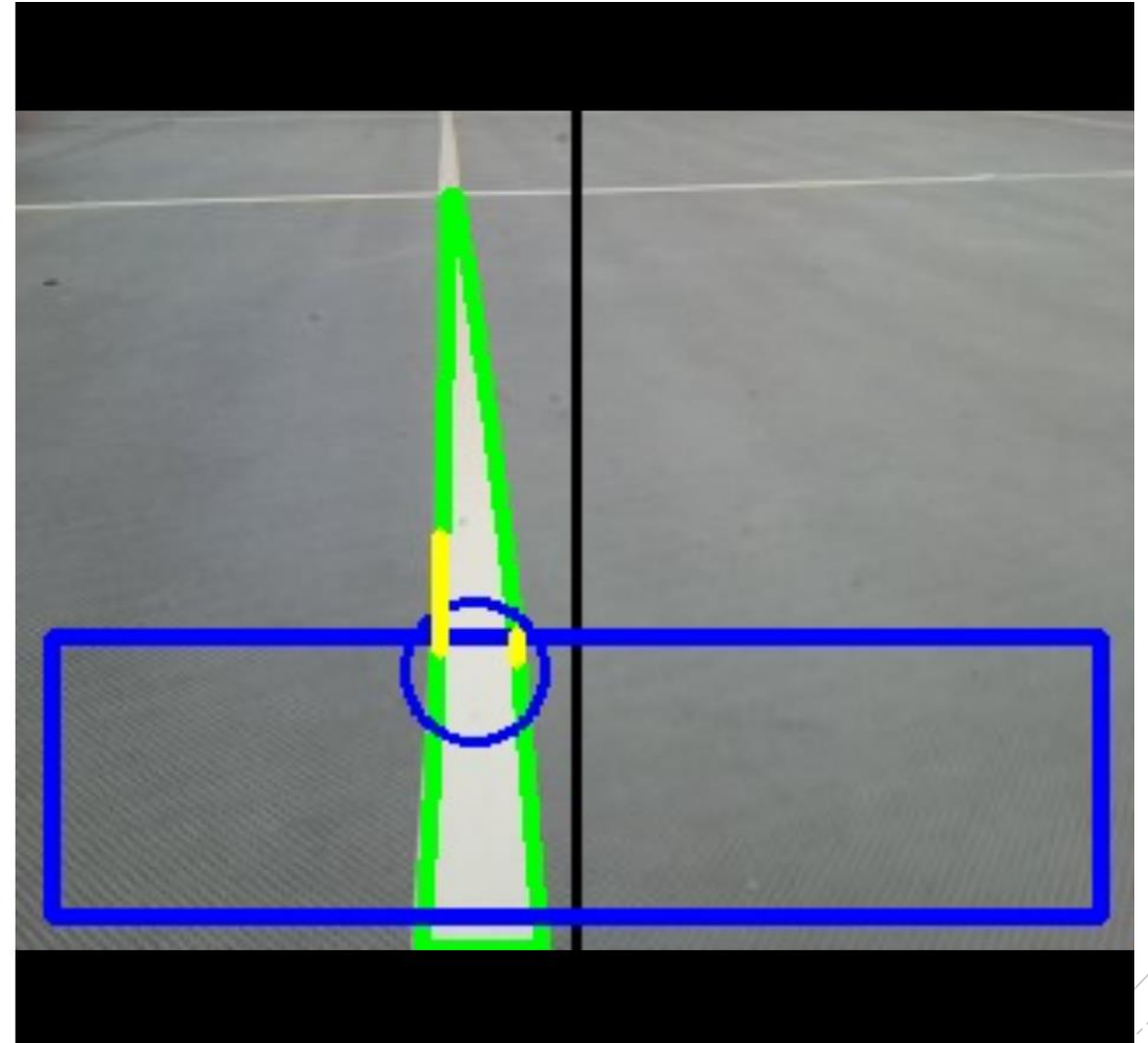
0.5

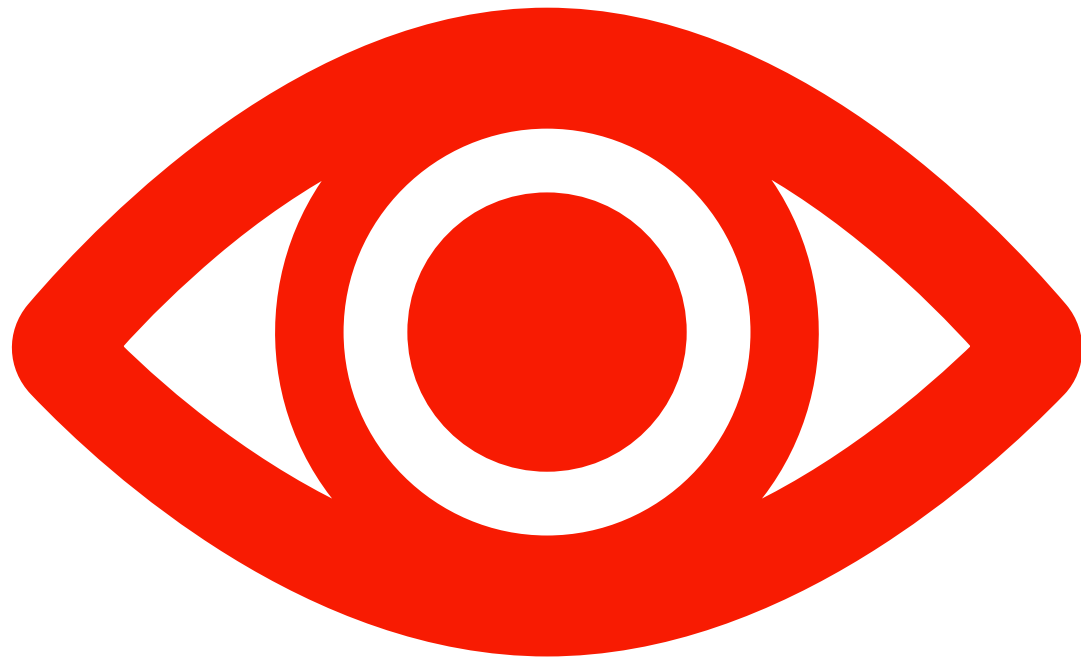
UPDATE

Leave circuit: (road mode only) False

False ▾

UPDATE

Z
Q S D

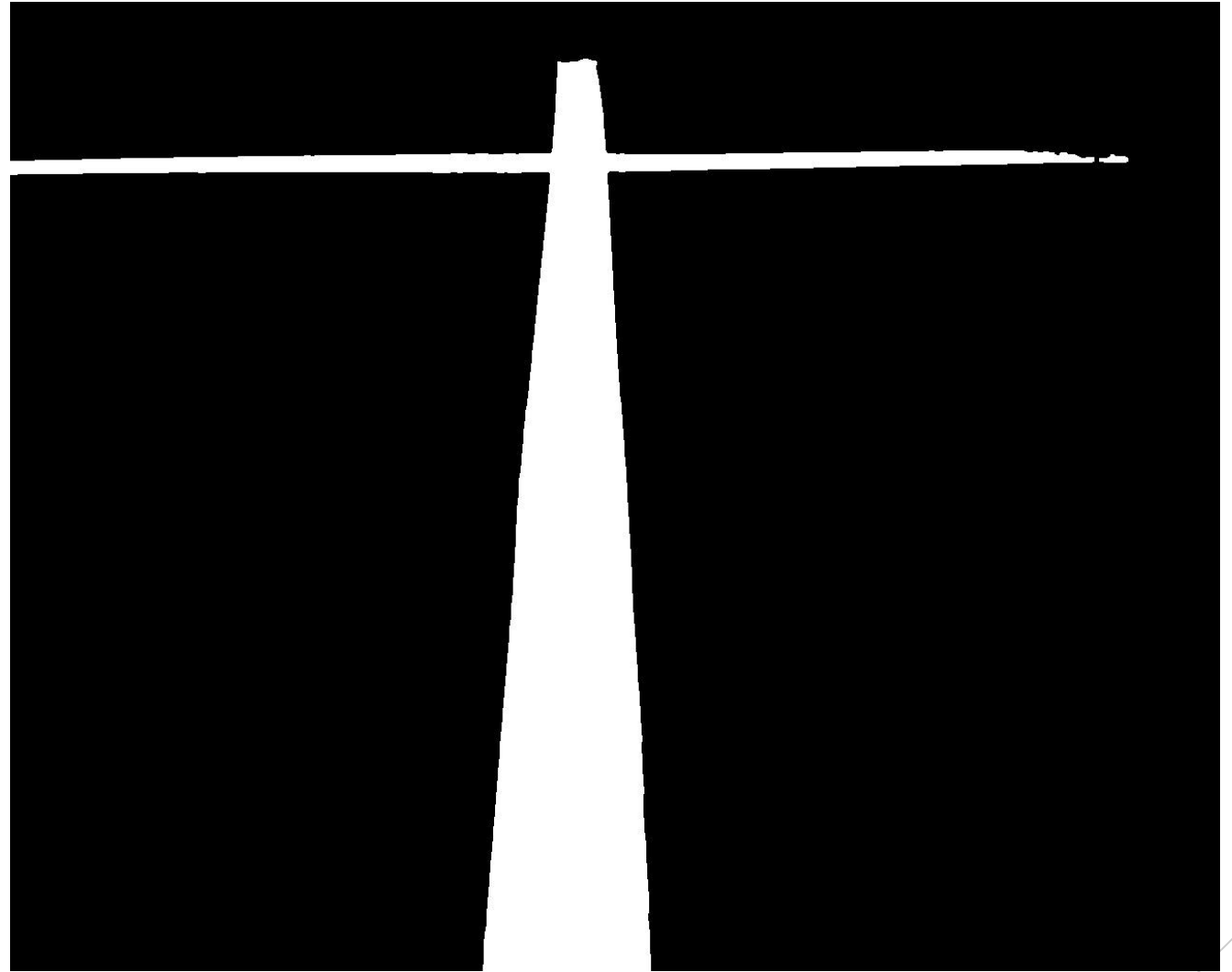


Vision par
ordinateur

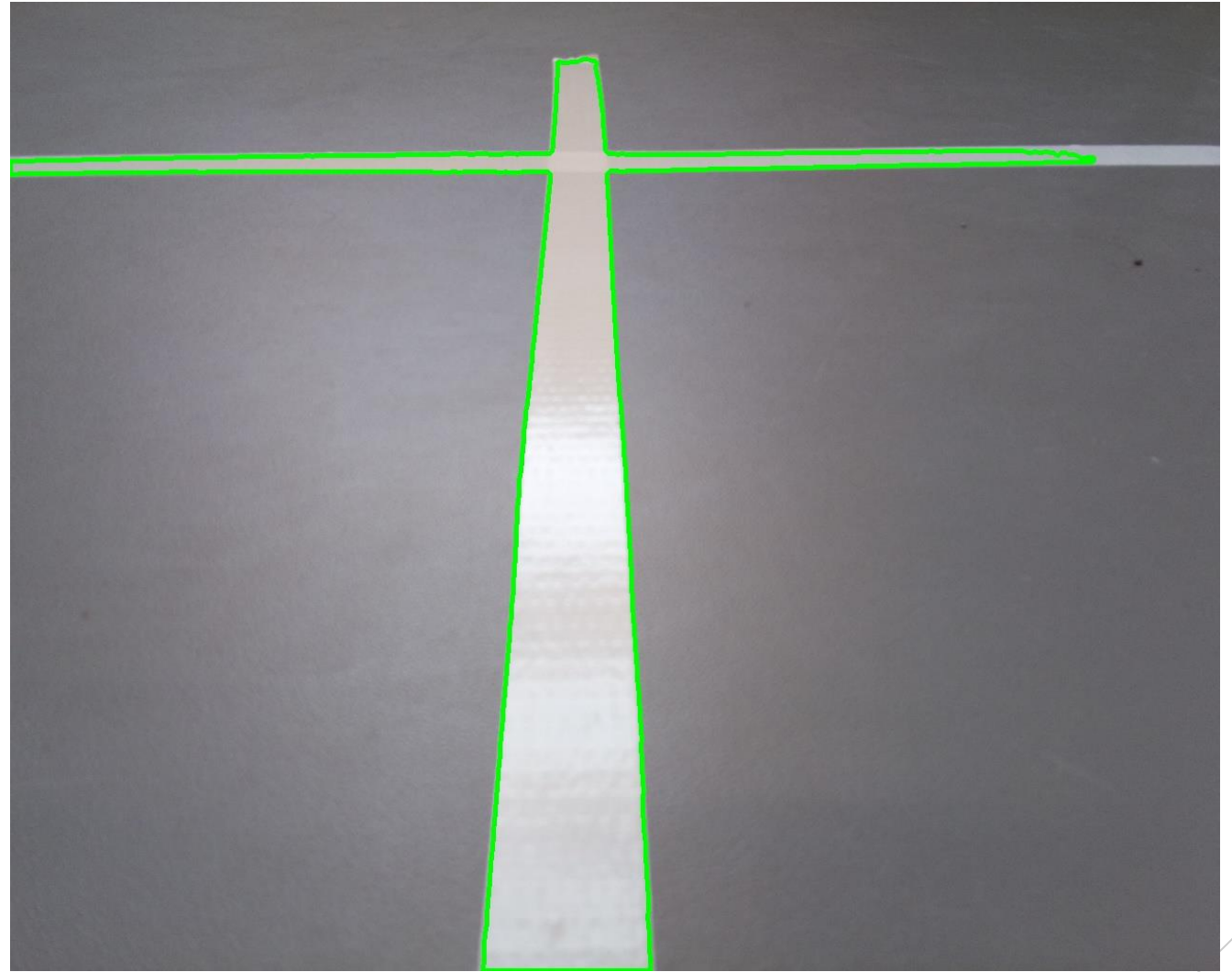
Image d'origine



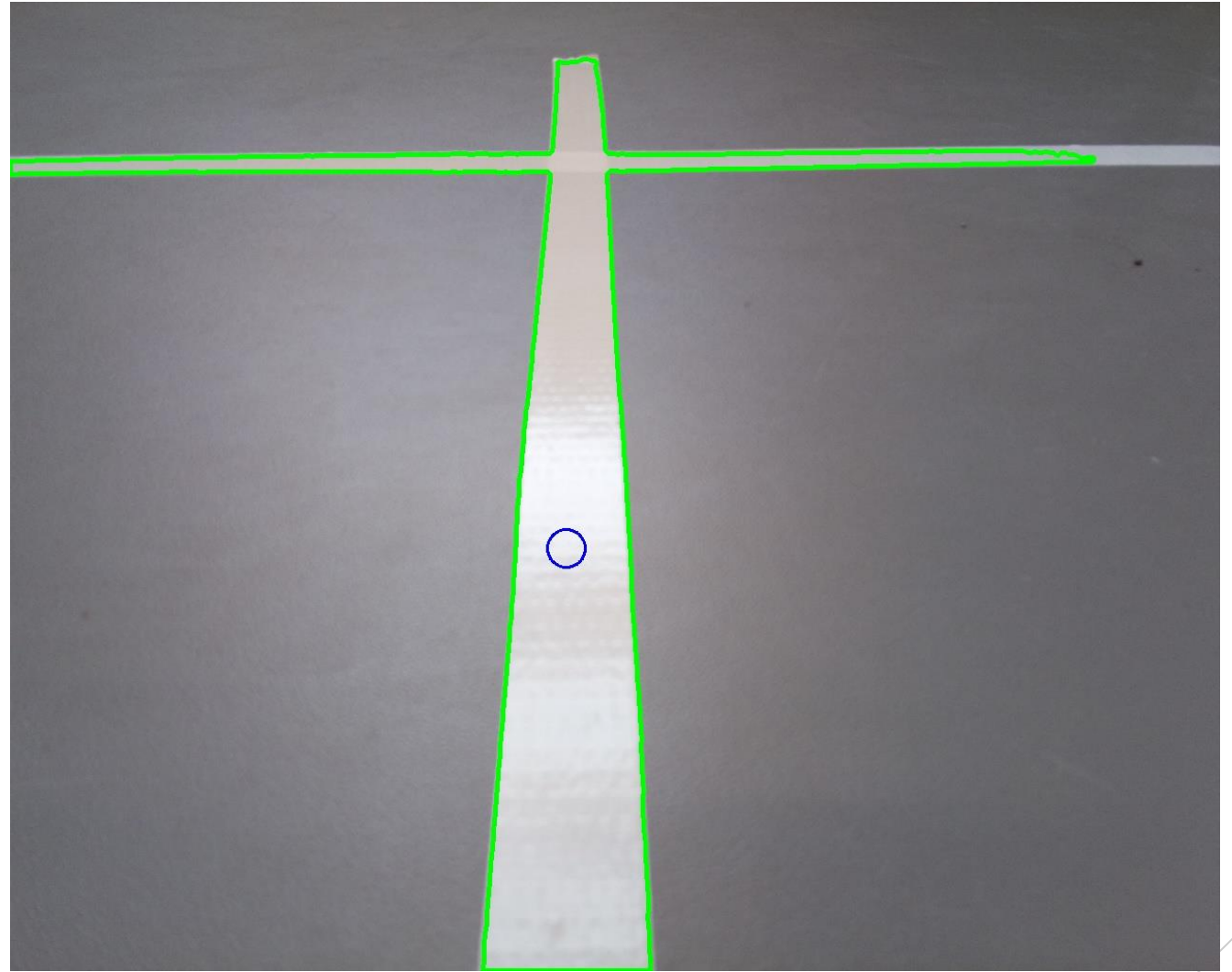
Pré-traitement :
seuillages,
ouverture



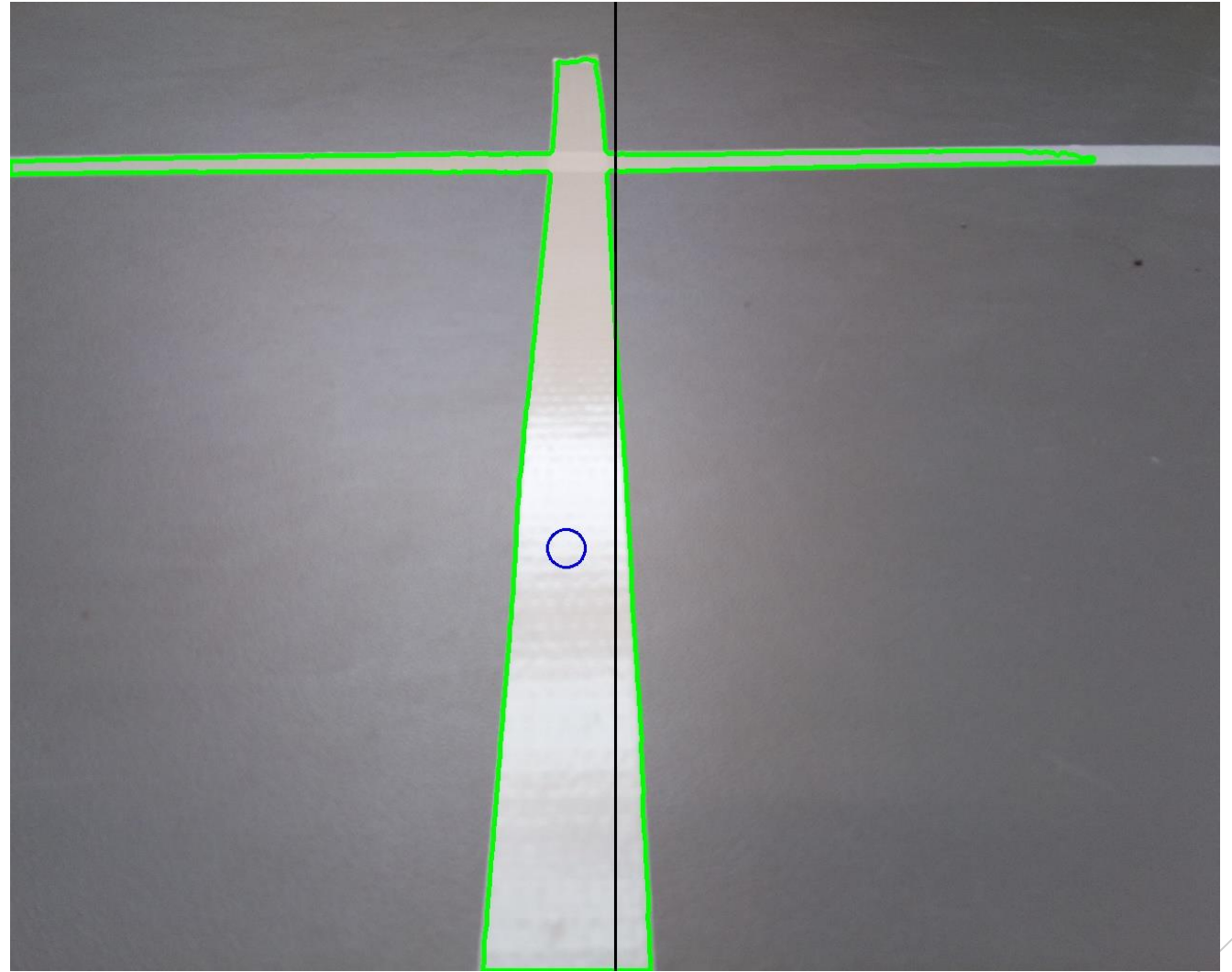
Détection des contours



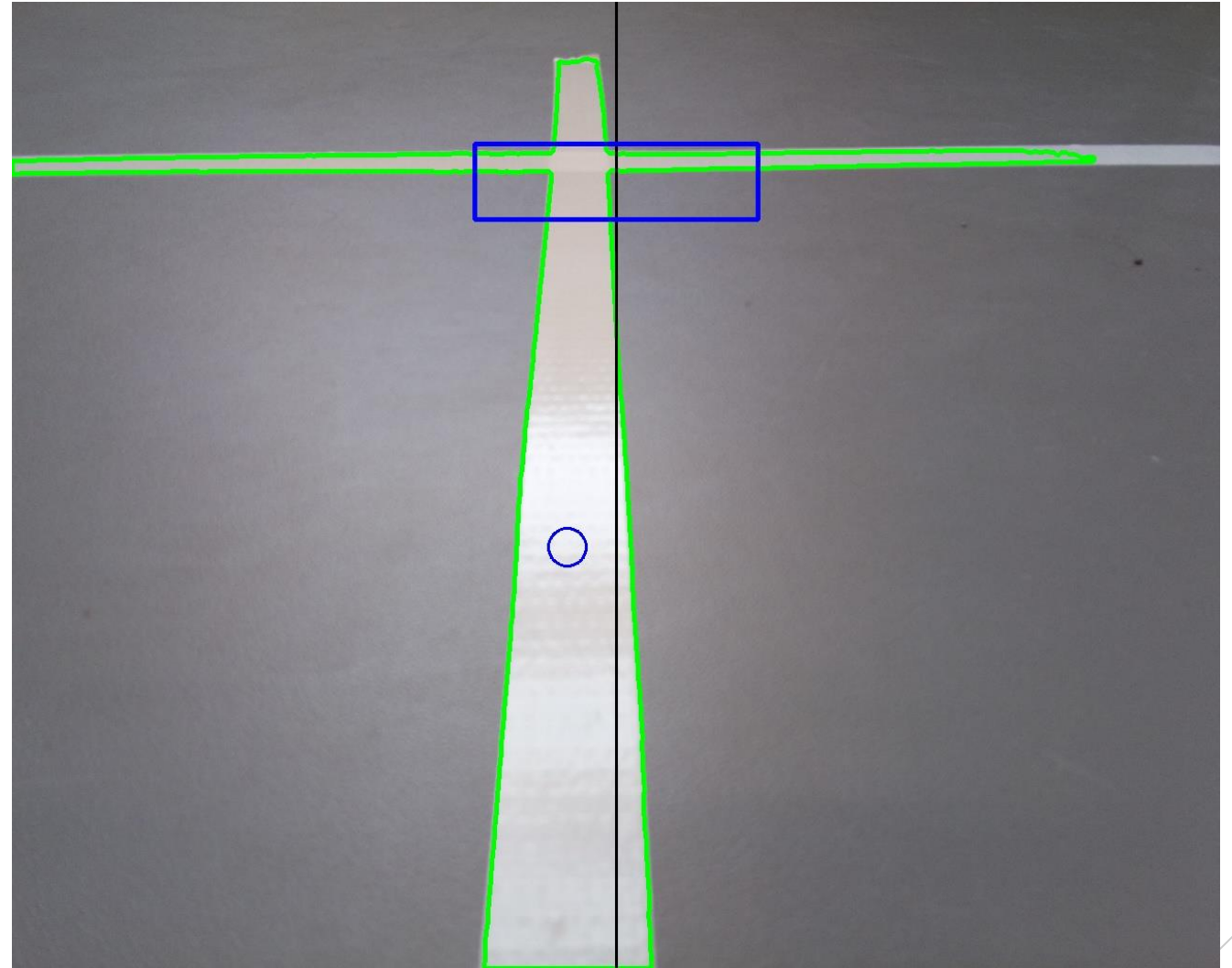
Calcul du
centroïde des
surfaces des
lignes



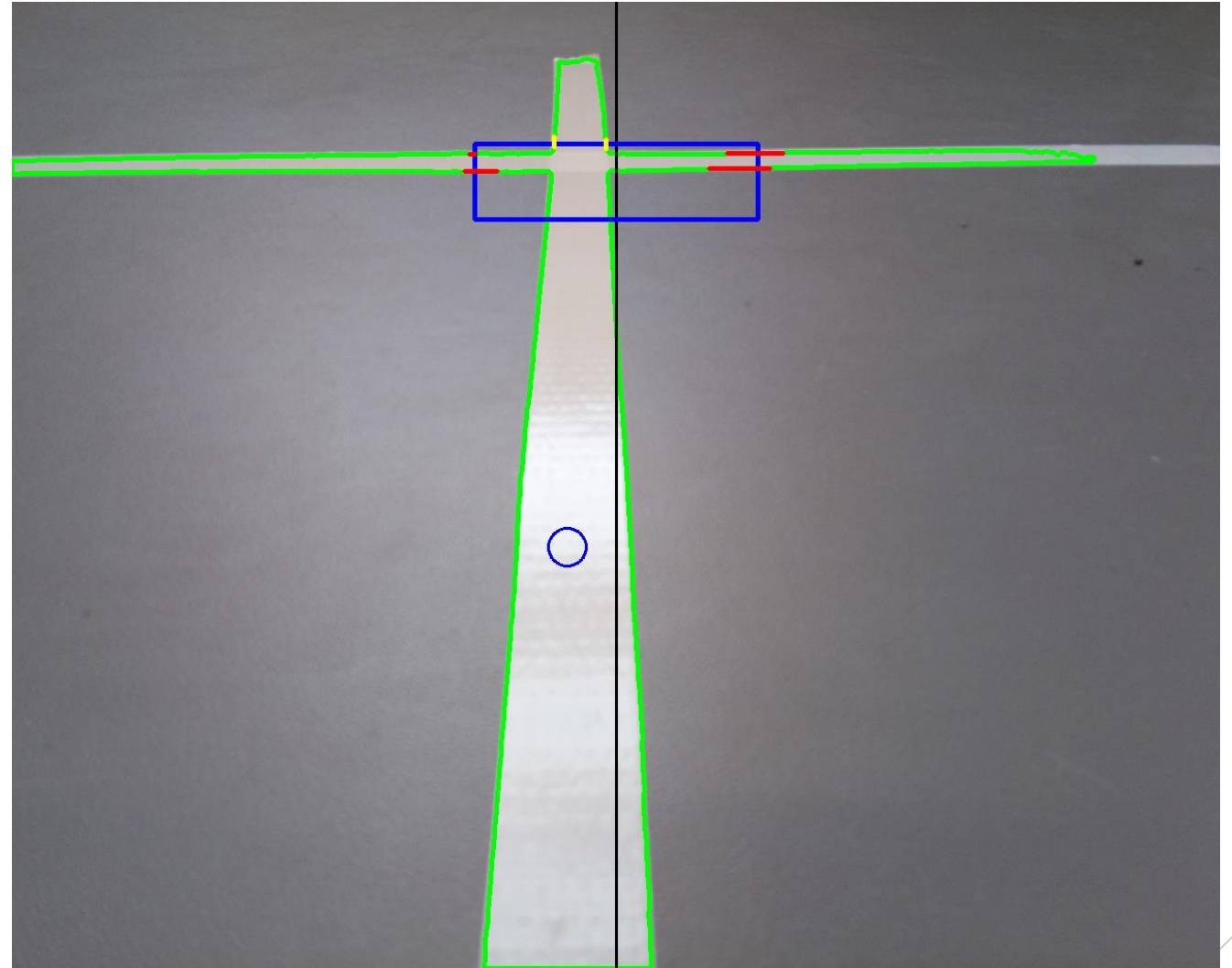
Suivi de ligne



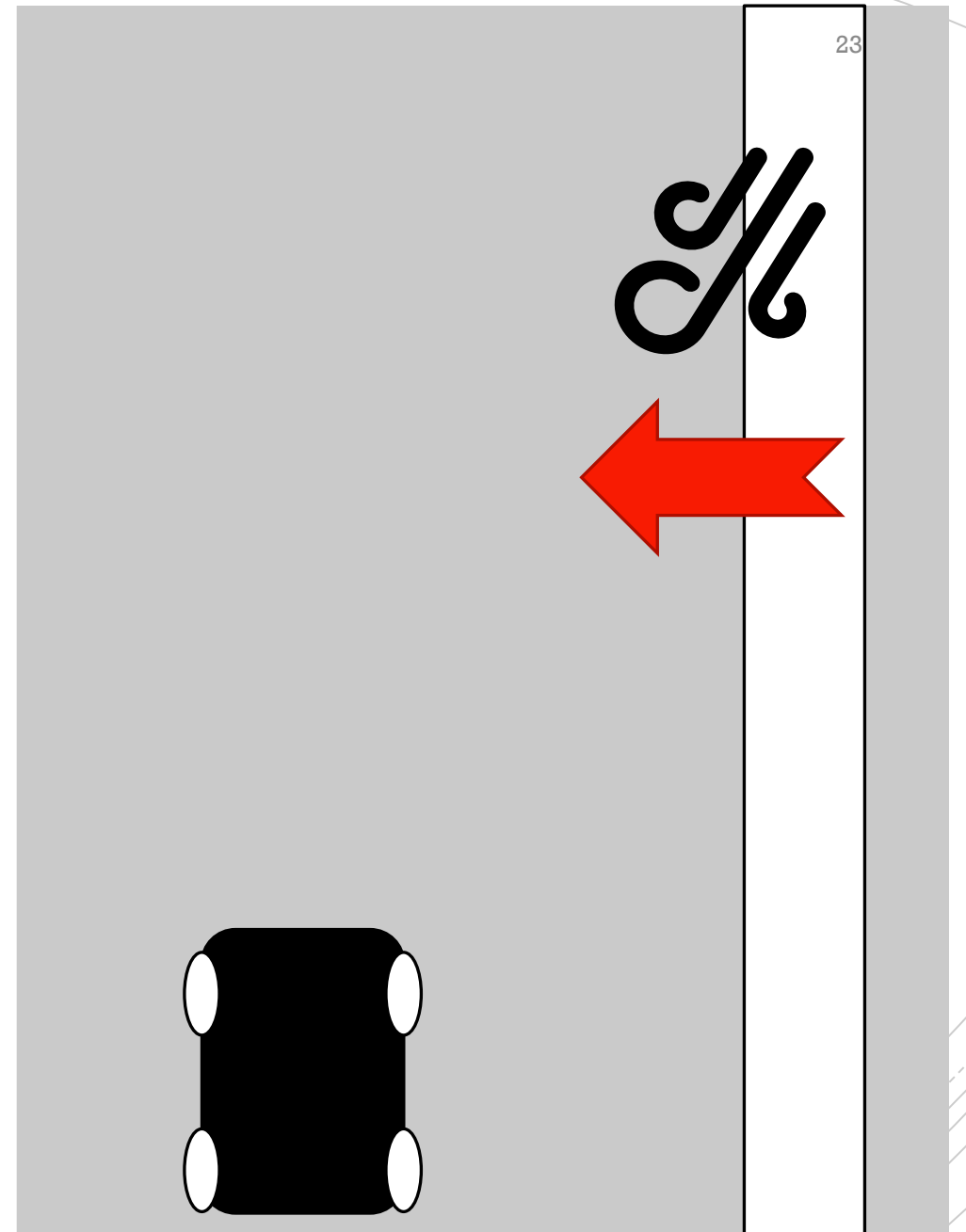
Détection d'intersection



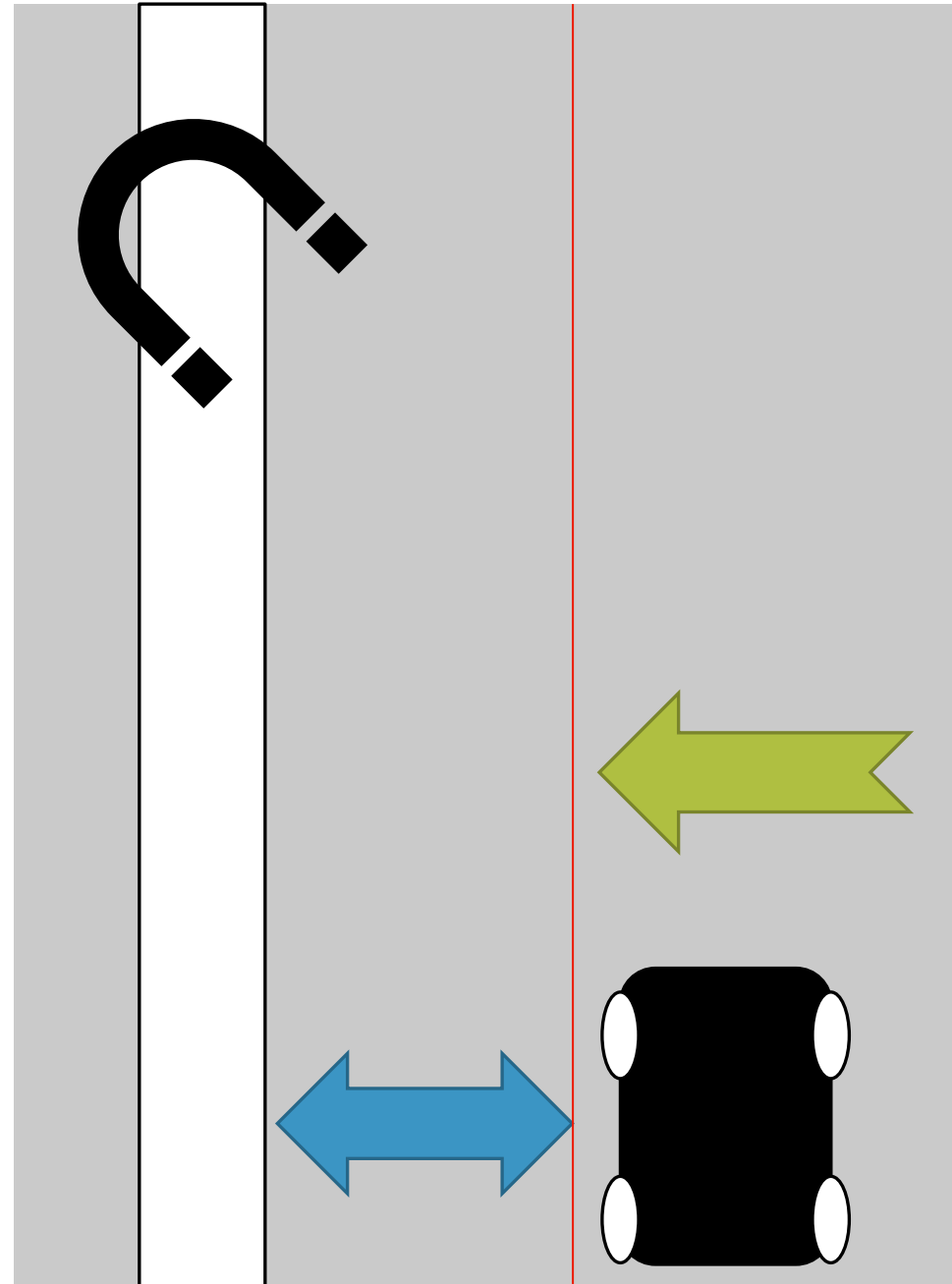
Détection d'intersection



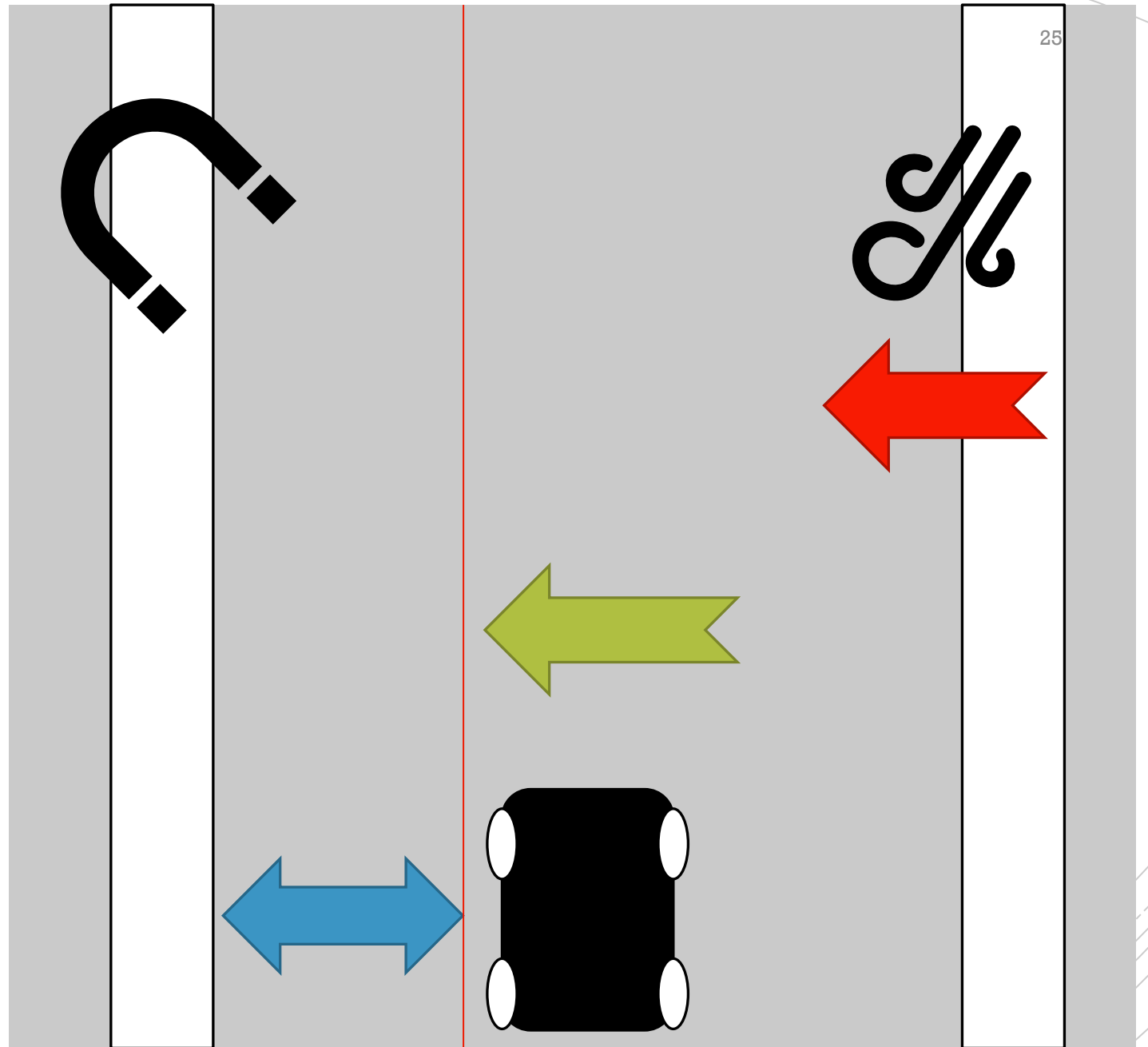
Navigation
entre les lignes



Navigation
entre les lignes



Navigation
entre les lignes



Merci de votre attention !



Avez-vous des
questions ?