
Docking autonome pour USV

Théo Massa, Kevin Ren, Guillaume Garde, Hugo Hofmann

ENSTA Bretagne-Lab STICC-Université Bretagne Sud

8 Mars 2024



1 Introduction

2 Conception du dock

- Base RTK
- Mise en place du dock
- Communication

3 Stratégie d'approche de docking

- Filtre de Kalman
- Guidage par champ de potentiel artificiel
- Algorithme

4 Architecture logicielle

- ROS
- Architecture du projet

5 Résultats

Plan

- 1 Introduction
- 2 Conception du dock
 - Base RTK
 - Mise en place du dock
 - Communication
- 3 Stratégie d'approche de docking
 - Filtre de Kalman
 - Guidage par champ de potentiel artificiel
 - Algorithme
- 4 Architecture logicielle
 - ROS
 - Architecture du projet
- 5 Résultats

Introduction



- Créer une solution de docking autonome
- Véhicule de surface
- Utilisation de capteurs basiques
- Crédit d'un système physique pour le dock

Plan

1 Introduction

2 Conception du dock

- Base RTK
- Mise en place du dock
- Communication

3 Stratégie d'approche de docking

- Filtre de Kalman
- Guidage par champ de potentiel artificiel
- Algorithme

4 Architecture logicielle

- ROS
- Architecture du projet

5 Résultats

Test

Mise en place du dock

- Boîte étanche
- IMU en dehors (perturbations électromagnétiques)



Figure – Mise en place du dock

Dock entier



Communication avec le reste du système

`${latitude}, {longitude}; {roll}, {pitch}, {yaw}`

Avantages

- Format léger
- Compatible peu importe les versions de ROS

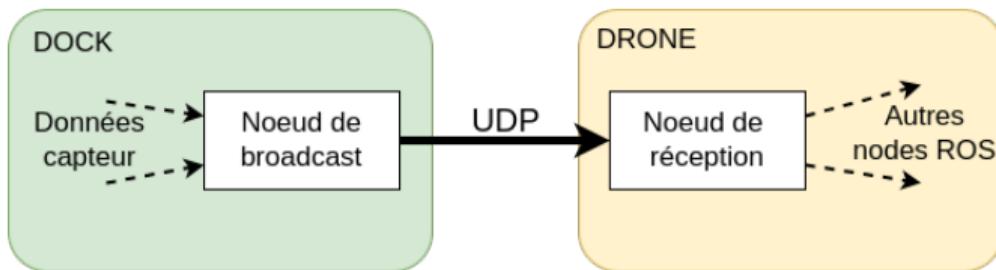


Figure – La communication dock → drone

Plan

1 Introduction

2 Conception du dock

- Base RTK
- Mise en place du dock
- Communication

3 Stratégie d'approche de docking

- Filtre de Kalman
- Guidage par champ de potentiel artificiel
- Algorithme

4 Architecture logicielle

- ROS
- Architecture du projet

5 Résultats

Filtre de Kalman

Modèle de Dubins :

- Vecteur d'état $\mathbf{x} = (x, y, z, \psi)^T$
- Mesure $\mathbf{y} = (x_{GPS}, y_{GPS}, \psi_{GPS/IMU})^T$



Figure – Rover

Filtre de Kalman

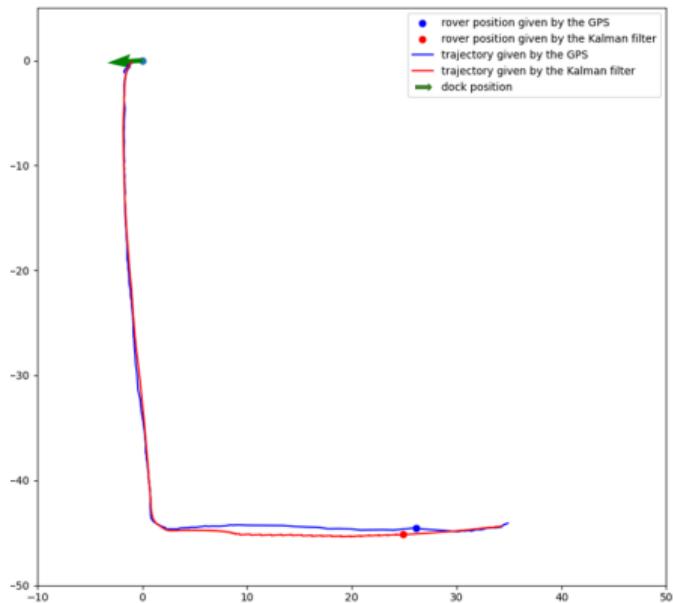


Figure – Comparatif les données du filtre de Kalman et les données brutes

Guidage par champ de potentiel artificiel

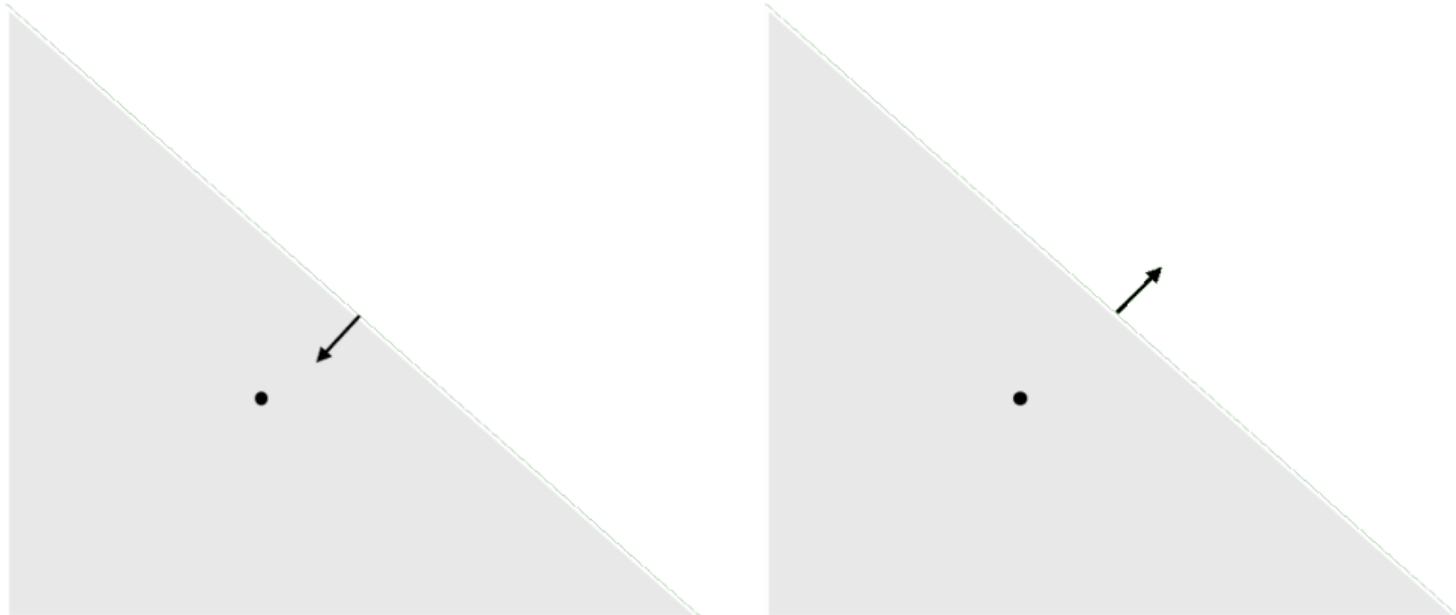


Figure – (a) À gauche le robot se trouve dans le demi-plan en face du robot (b) À droite le robot se trouve dans le demi-plan derrière le dock

Guidage par champ de potentiel artificiel

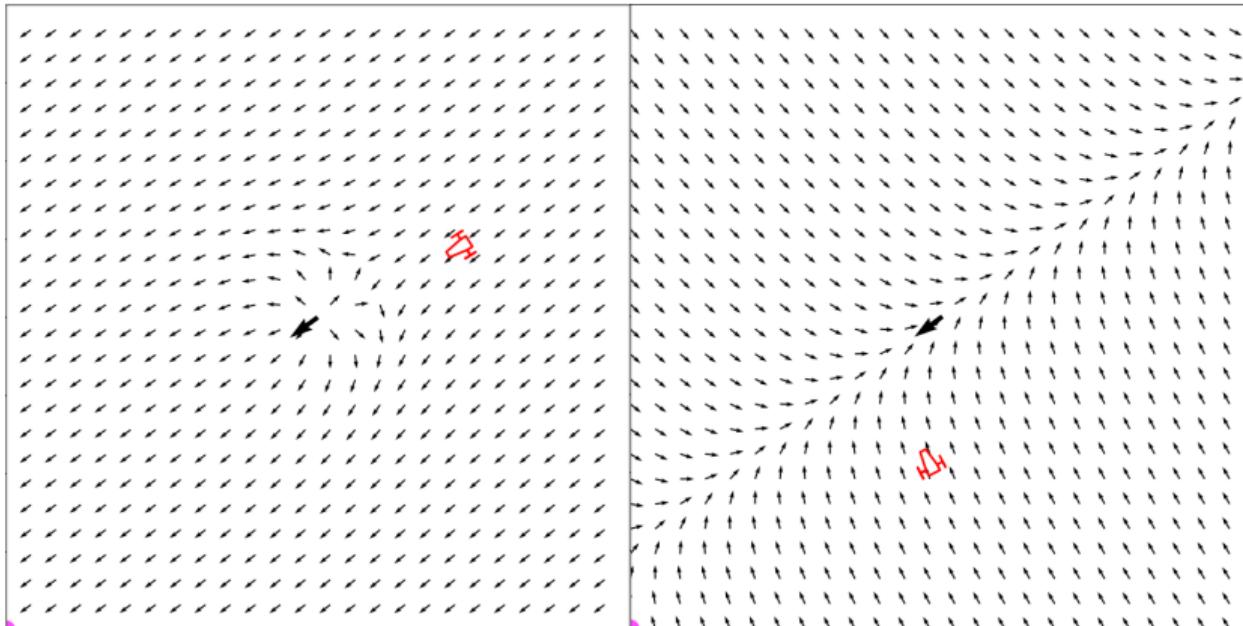


Figure – (a) Champ de vecteurs dans le premier cas. (b) Champ de vecteurs dans le second cas.

Algorithme

Régulation en vitesse fait via cmd_vel.

Régulation en cap via un PI.

Instabilité au niveau de la ligne séparatrice, les conditions de transition ont été adaptées :

if $\overrightarrow{M'M} \cdot \vec{t} < -L$ and cas1 \Rightarrow cas2 (1)

if $\overrightarrow{M'M} \cdot \vec{t} > L$ and cas2 \Rightarrow cas1 (2)

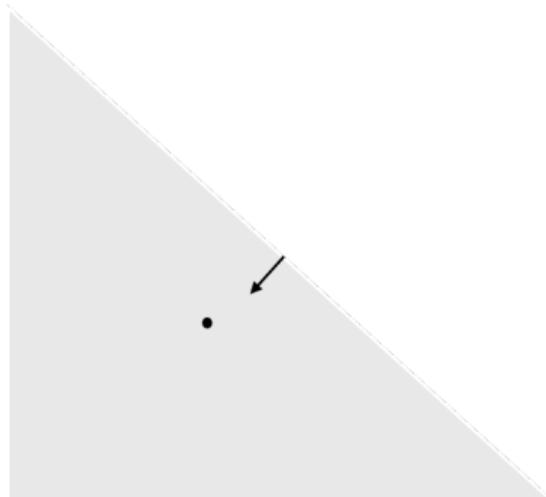


Figure – Ligne séparatrice

Plan

- 1 Introduction
- 2 Conception du dock
 - Base RTK
 - Mise en place du dock
 - Communication
- 3 Stratégie d'approche de docking
 - Filtre de Kalman
 - Guidage par champ de potentiel artificiel
 - Algorithme
- 4 Architecture logicielle
 - ROS
 - Architecture du projet
- 5 Résultats



Figure – ROS Melodic

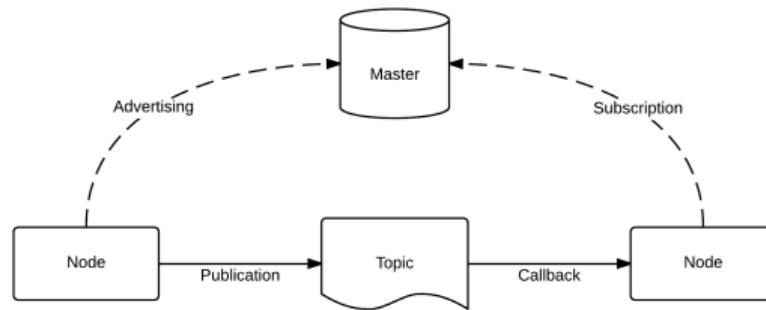


Figure – Schéma de fonctionnement

Principe de ROS

- Middleware commun en robotique
- Principe de nodes/topics

Architecture du projet

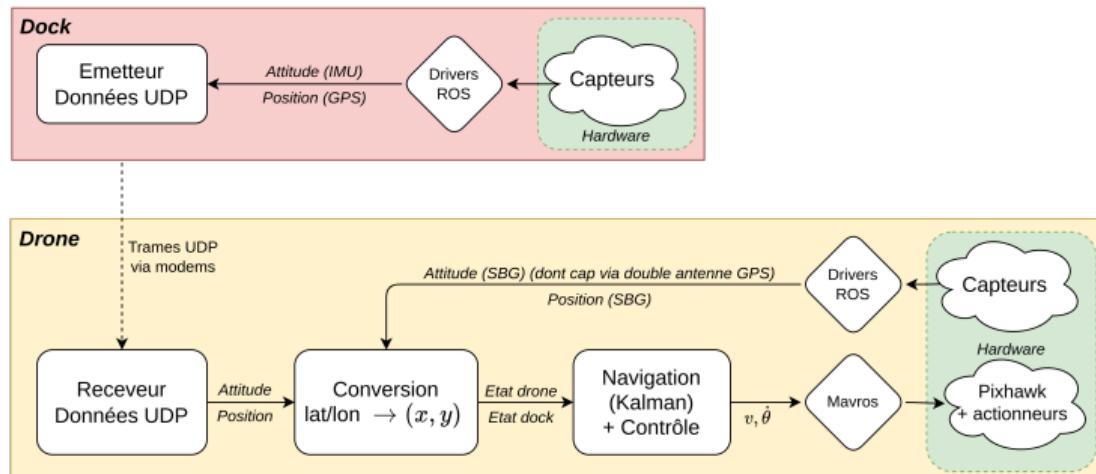


Figure – Schéma général de l'architecture du projet

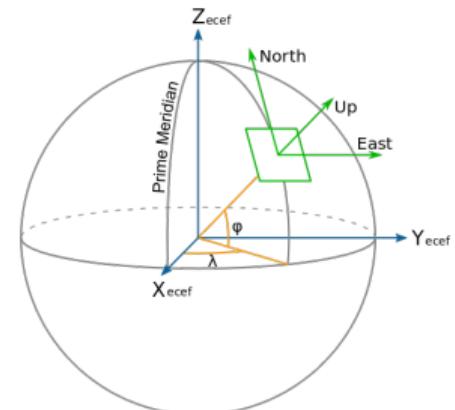


Figure – Repère ENU

Plan

1 Introduction

2 Conception du dock

- Base RTK
- Mise en place du dock
- Communication

3 Stratégie d'approche de docking

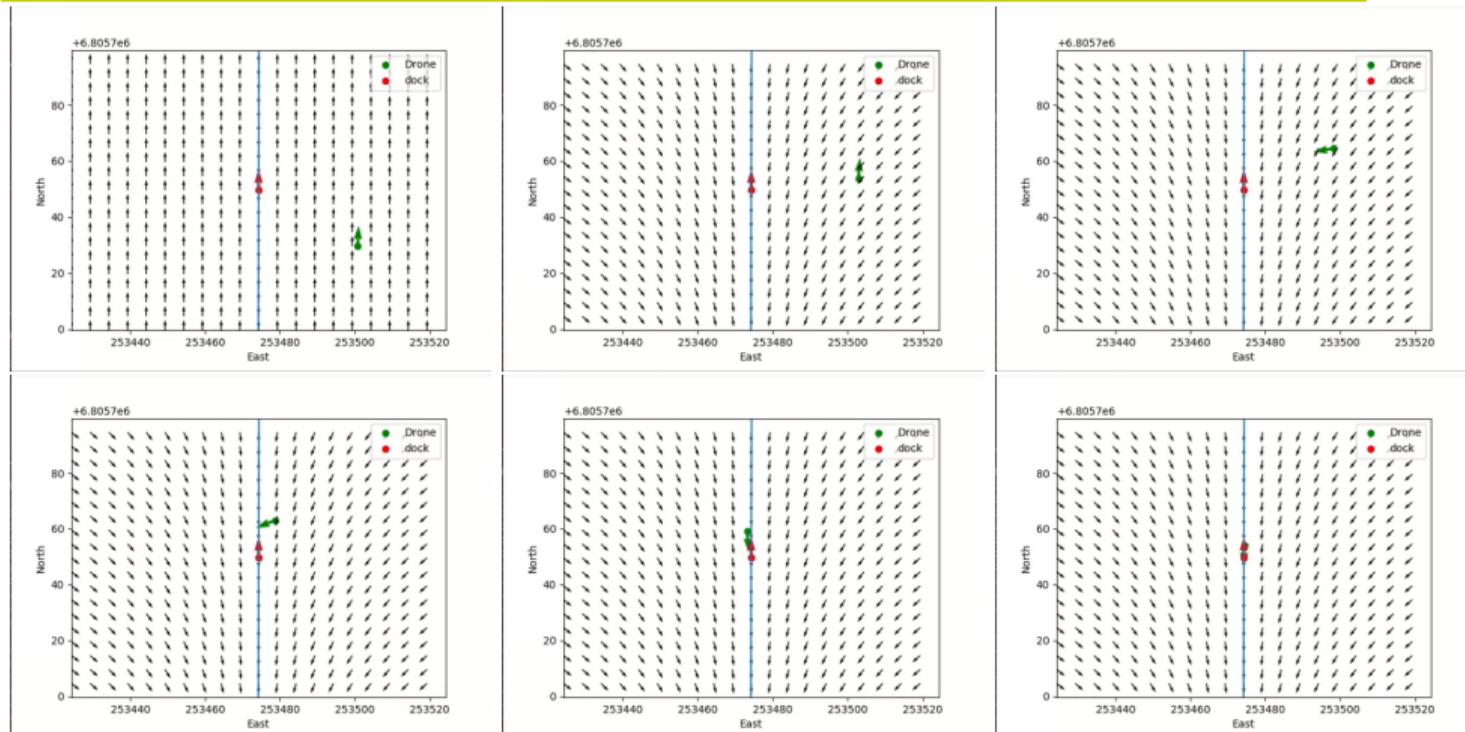
- Filtre de Kalman
- Guidage par champ de potentiel artificiel
- Algorithme

4 Architecture logicielle

- ROS
- Architecture du projet

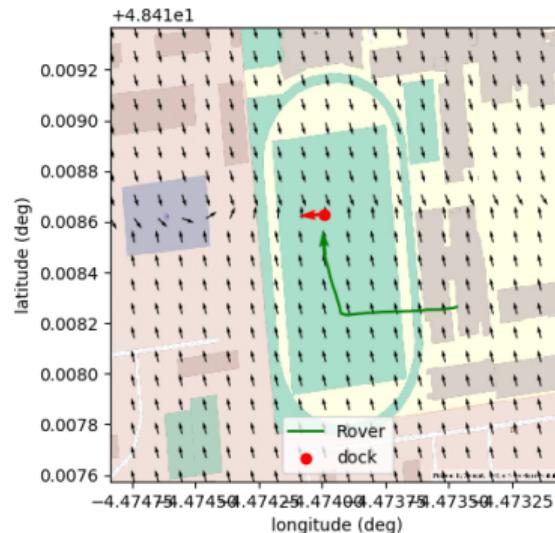
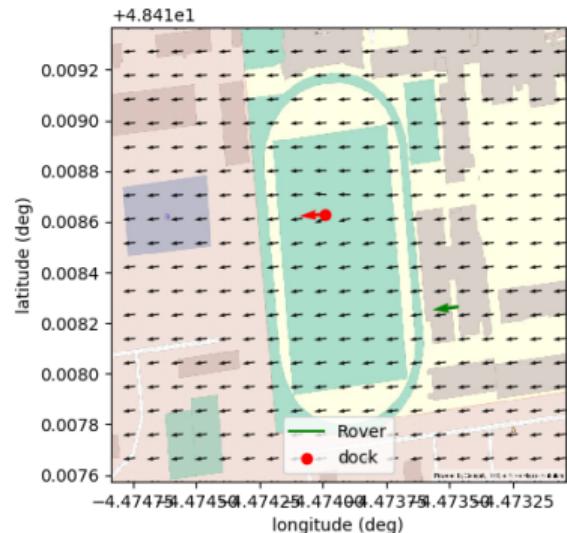
5 Résultats

Résultats



Algorithme mis en place sur le lac

Résultats



Algorithme mis en place sur le stade avec le rover

Conclusion

Différents points de validation

- Objectif globalement accompli
- Navigation et guidage très efficaces
- Architecture efficace

Futur du projet

- Calibration correcte du dock
- Plus d'essais en lac avec vrai dock
- Mise en place de la correction RTK