

# Conception d'une plateforme robotique communicante pour un convoi autonome

Joseph LOUVILLE, Naila BOUCHEMAL, Jae Yun JUN KIM  
ECE Paris

## CONTEXTE

- De nombreux simulateurs de robotique et de communications ont été développés ces dernières années pour le développement des véhicules autonomes : ROS (Robot Operating System), OMNeT++, SUMO.
- Des plateformes polyvalentes, robotiques communicantes, créées avec ces simulateurs, permettent d'avoir une visualisation des véhicules et des communications (5G NR (New Radio), ITS-G5 (Intelligent Transport System - G5)).
- Les convois comportent de nombreux avantages, en réduisant la consommation ou facilitent la distribution du trafic routier, pouvant faciliter des livraisons en milieu urbain, par exemple en organisant la répartition des véhicules dans l'ensemble du trafic.

## ÉTAT DE L'ART

### Robot Operating System (ROS) The Complete Reference [1]

Ce livre vient à présenter le fonctionnement de ROS et présenter des projets importants qui l'utilisent dans la modélisation.

### OMNeT++

Un simulateur Open-Source de réseaux de communication, mais permet aussi de simuler des systèmes informatiques, des réseaux de file d'attente et des architectures hardware.

### Simu5G-An OMNeT++ Library for End-to-End Performance Evaluation of 5G Networks [2]

Il s'agit d'un article visant à introduire la bibliothèque SIMU5G, qui constitue des modèles de simulation pour OMNeT++. Il permet aussi de lier SUMO avec la 5G New Radio.

### COPADrive - A Realistic Simulation Framework for Cooperative Autonomous Driving Applications [4]

Ce projet met en place une plateforme communicante entre ROS et OMNeT++, par le biais de la bibliothèque Artery.

## PROBLÉMATIQUE

Comment concevoir une plateforme robotique communicante (PRC) pour un convoi autonome?

## OBJECTIF

- Déterminer quelles architectures sont appropriées pour le développement de la PRC pour un convoi autonome.
- Implémenter un convoi autonome à partir de ces plateformes
- Mesurer les performances des véhicules si on apporte ce concept de PRC.
- Comparer les taux d'erreurs entre la plateforme de référence sans simulateur communicant et une PRC.

## MÉTHODOLOGIE

### Configuration :

- ROS Melodic
- Ubuntu 18.04 64bits
- GPU Nvidia GeForce RTX 3070
- CPU Intel® Core™ i7-10700 @ 2.90GHzx16
- OMNeT++ 5.6.2

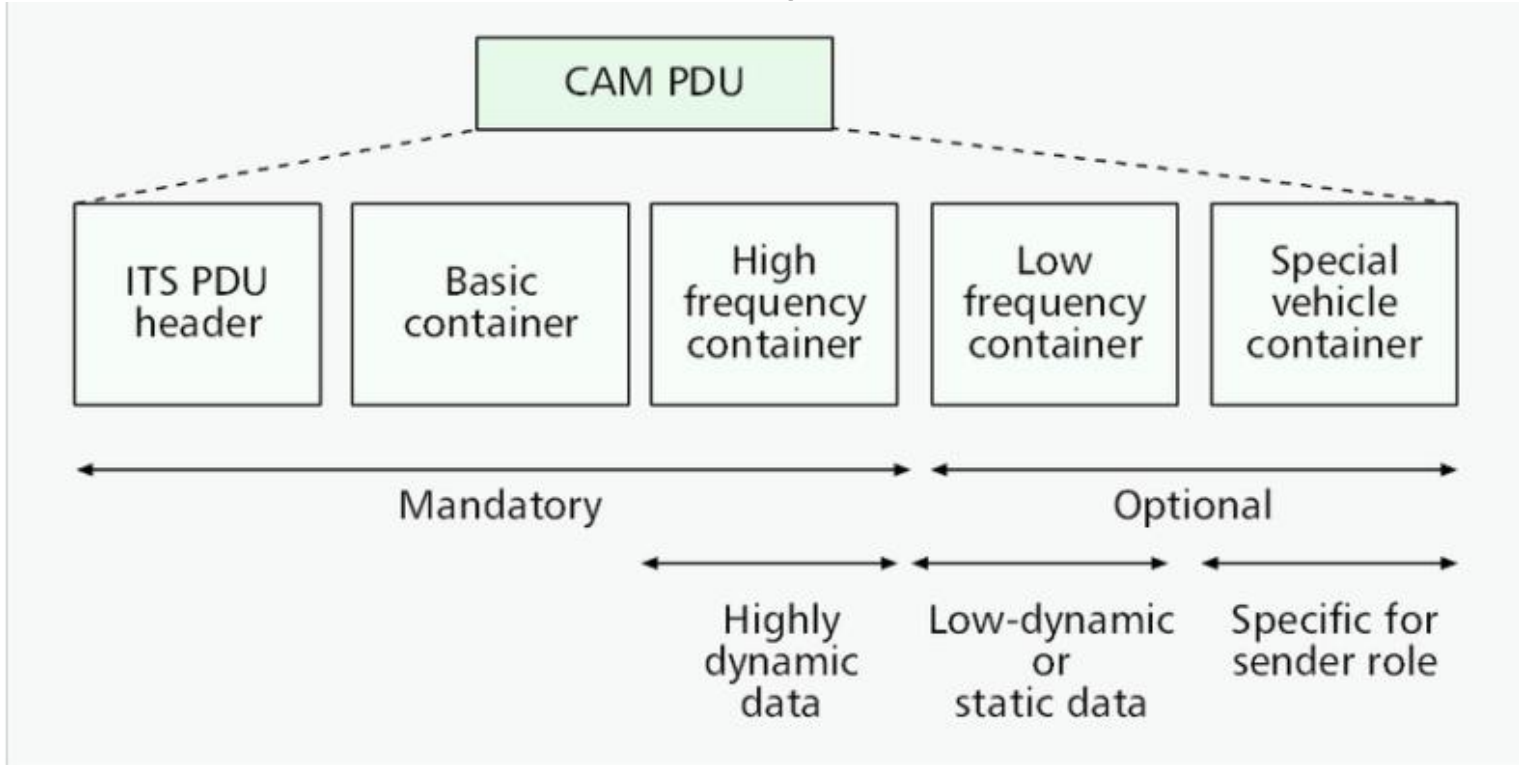
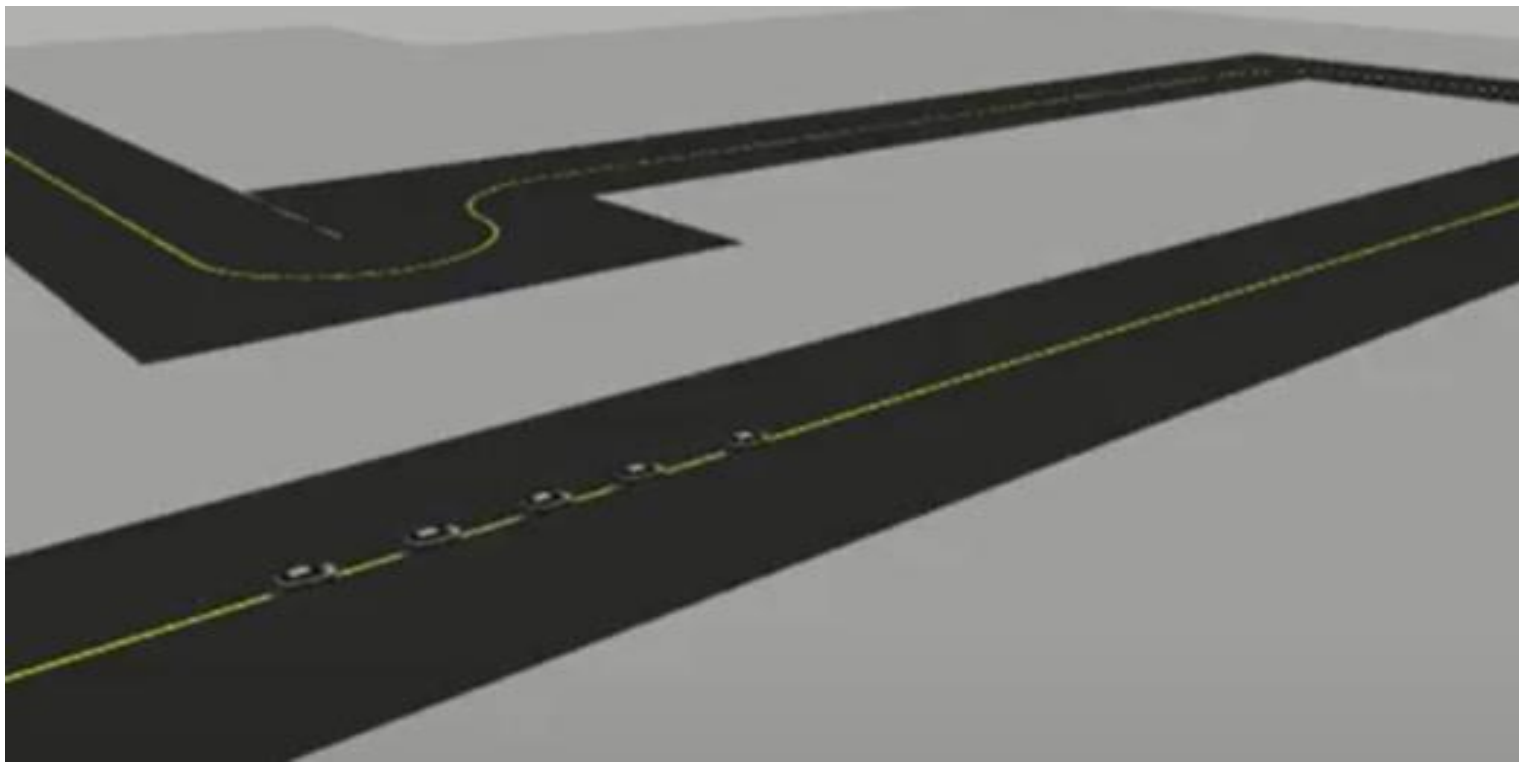
### Protocole :

COPADrive (PRC) et PFE ECE - Challenge UTAC CERAM comparés, tous les deux basés sur ROS pour des projets avec convoi, COPADrive est un PRC apportant la communication sur OMNeT++.

Par l'utilisation des topics sur ROS, récupération des commande de vitesse par CAMs (Cooperative Awareness Messages) et les mesures de vitesses réelles.

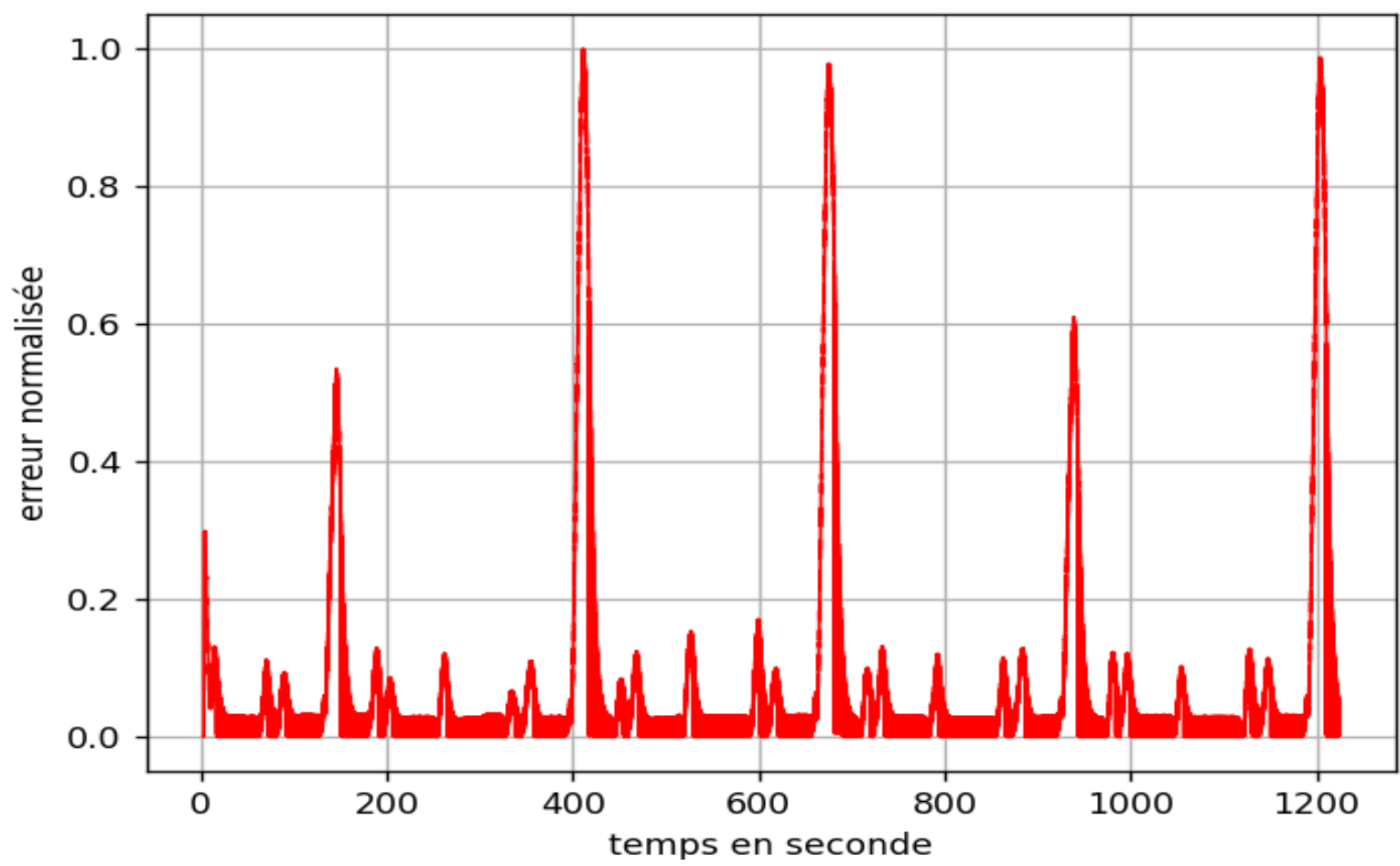
La mesure de la vitesse réelle des véhicules et de la vitesse commandée seront intégrées par une formule d'erreur  $e(t) = r(t) - y(t)$  (avec  $r(t)$  la vitesse commandée et  $y(t)$  la vitesse réelle qui) sera normalisée pour ensuite en obtenir l'erreur quadratique moyenne normalisée (NMSE) afin de déterminer quelle plateforme à un meilleure impact sur la trajectoire des véhicules. Une valeur faible de NMSE correspond à une plus grande précision.

$$NMSE(\text{mean square error}) = 10 \log_{10} \left( \frac{\sum_{t=1}^T |e(t)|^2}{\sum_{t=1}^T |r(t)|^2} \right)$$



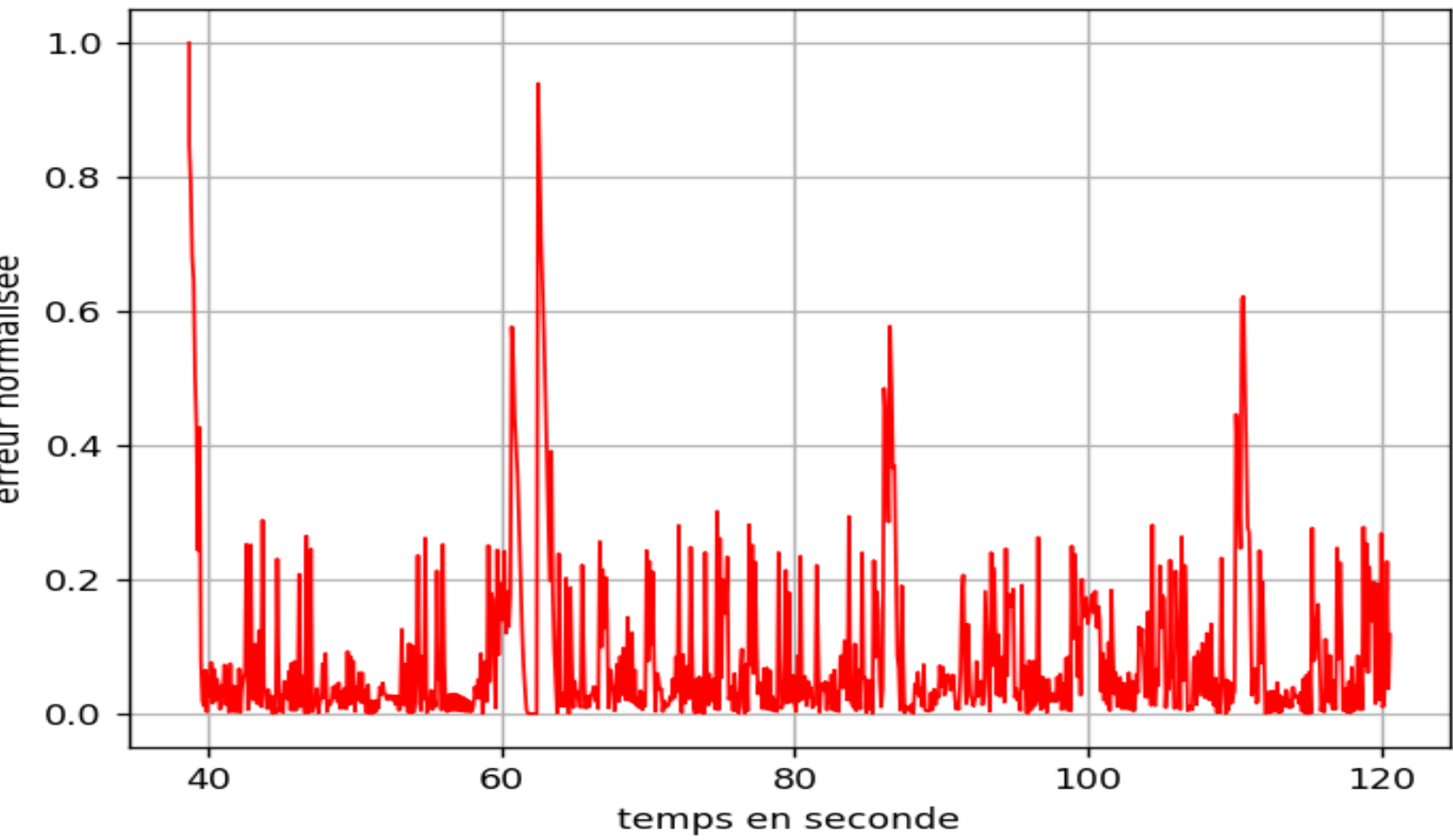
## RÉSULTATS

### COPADrive (PRC)



$$NMSE_{COPA} = 19,2999$$

### PFE ECE



$$NMSE_{PFE} = 34,5413$$

Ces graphiques montrent l'erreur normalisée au cours du temps pour les deux projets. On peut remarquer que l'erreur est plus régulière pour le projet PFE, mais il arrive que des pics importants d'erreurs arrivent sur COPADrive.

## DISCUSSIONS & PERSPECTIVE

- Dans la simulation, on obtient des erreurs de vitesse moins importants avec COPADrive.

- Les méthodes pour récupérer les données ne sont pas facilement accessibles, ce qui pourrait être intéressant dans le futur pour le développement d'une PRC de générer facilement les données pour analyse.

- Possiblement pour un projet PFE futur, il serait bienvenu de s'inspirer de COPADrive ou d'un projet récent inspiré de celui-ci, AuNa [3], pour développer une analyse de communication avec les DENMs (Decentralized Environmental Notification Messages) afin de regarder les interactions avec l'environnement.

## RÉFÉRENCES & BIBLIOGRAPHIE

- [1] A. Koubaa. Robot Operating System (ROS) : The Complete Reference (Volume 6). Studies in Computational Intelligence. Springer International Publishing, 2021.
- [2] Giovanni Nardini, Dario Sabella, Giovanni Stea, Purvi Thakkar, and Antonio Virdis. Simu5g—an omnet++ library for end-to-end performance evaluation of 5g networks, 2020.
- [3] Harun Teper, Anggera Bayuwindra, Raphael Riebl, Ricardo Severino, Jian-Jia Chen, and Kuan-Hsun Chen. Auna : Modularly integrated simulation framework for cooperative autonomous navigation. 07 2022.
- [4] Bruno Vieira, Ricardo Severino, Enio Vasconcelos Filho, Anis Koubaa, and Eduardo Tovar. Copadride - a realistic simulation framework for cooperative autonomous driving applications. In 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE), pages 1–6, Nov 2019.