Mixed Reality Autonomous Vehicle Simulation: Implementation of a Hardware-In-the-Loop

Robin Baruffa
Univ. de Technologie de Belfort-Montbéliard
Belfort, France
email: robin.baruffa@utbm.fr

Jacques Pereira
Univ. de Technologie de Belfort-Montbéliard
Belfort, France
email: jacques.pereira@utbm.fr

Pierre Romet
CIAD (UMR 7533)
Univ. Bourgogne Franche-Comte, UTBM
Belfort, France
email: pierre.romet@utbm.fr

Franck Gechter
CIAD (UMR 7533)
Univ. Bourgogne Franche-Comte, UTBM
Belfort, France
email: franck.gechter@utbm.fr
Mosel Loria (UMR CNRS 7503)
Université de Lorraine
Vandoeuvre-lès-Nancy-54506, France

Tobias Weiss
CIAD (UMR 7533)
Univ. Bourgogne Franche-Comte, UTBM
Belfort, France
Institute of Energy Efficient Mobility
Univ. of Applied Science
Hochschule, Karlsruhe
email: tobias.weiss@hs-karlsruhe.de

IARIA

CIAD
Connaissance et Intelligence Artificielle Distribuées

utbm
université de technologie
Belfort-Montbéliard

Project SURATRAM

New freight transportation system consisting in a fleet of autonomous vehicles navigating across cities and rural areas following existing human-operated vehicles, such as tramways or buses.

The network is based on a multi agent architecture, allowing for optimized delivery paths and capable of coping with static and dynamic customers.

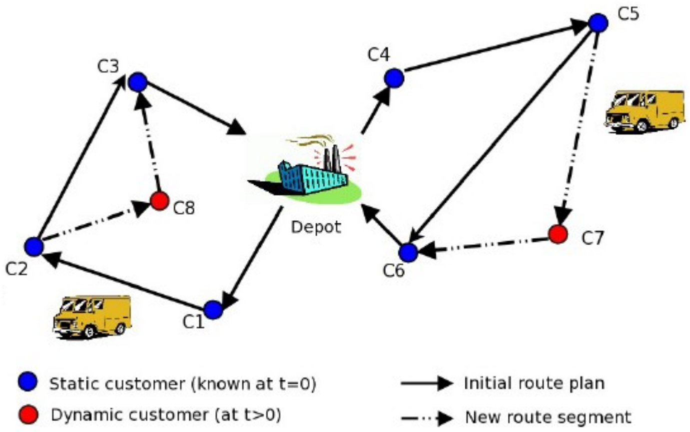

Figure 1 : Illustration of a dynamic vehicle routing optimization problem

Objective of the paper

- Give a presentation of our miniature Hardware-In-the-Loop (HIL) platform
- Investigate the difference in execution time depending on the nature of the sensor data (real or simulated)

Our miniature HIL car

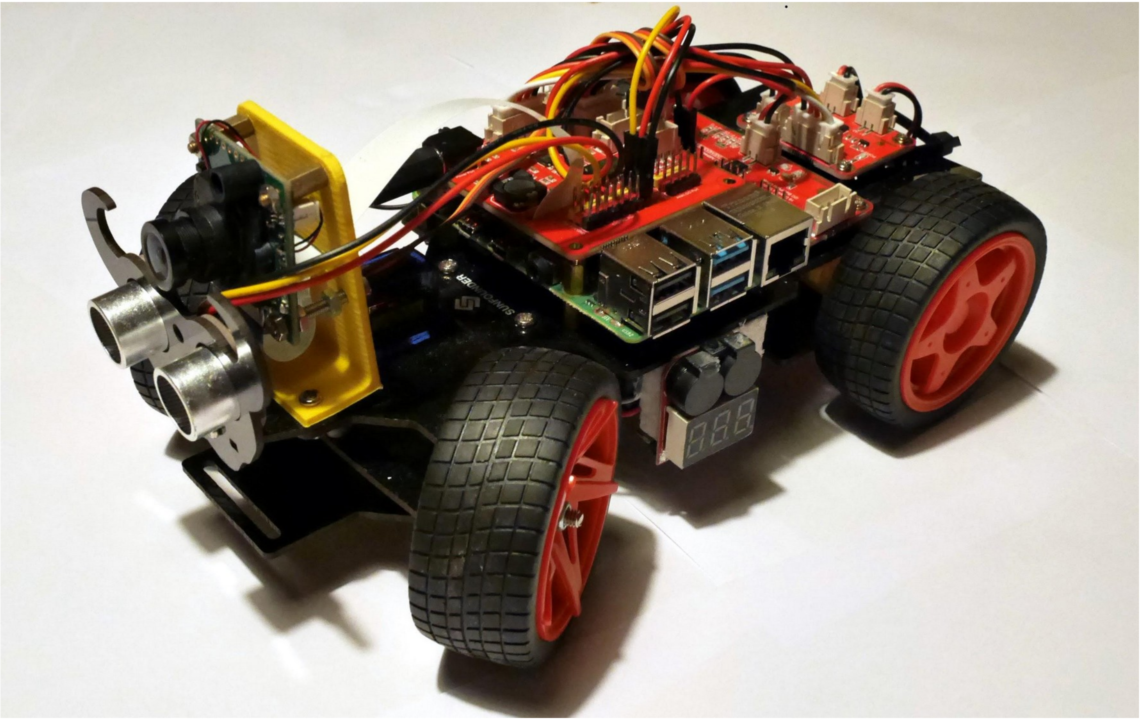

Figure 2 : Miniature car

Hardware :

Raspberry PI 4
Camera
Rear wheel drive
2 x 18650 li-ion batteries

Software :

Robot Operating System (ROS) middleware
Ubuntu 16.04

Our miniature HIL car

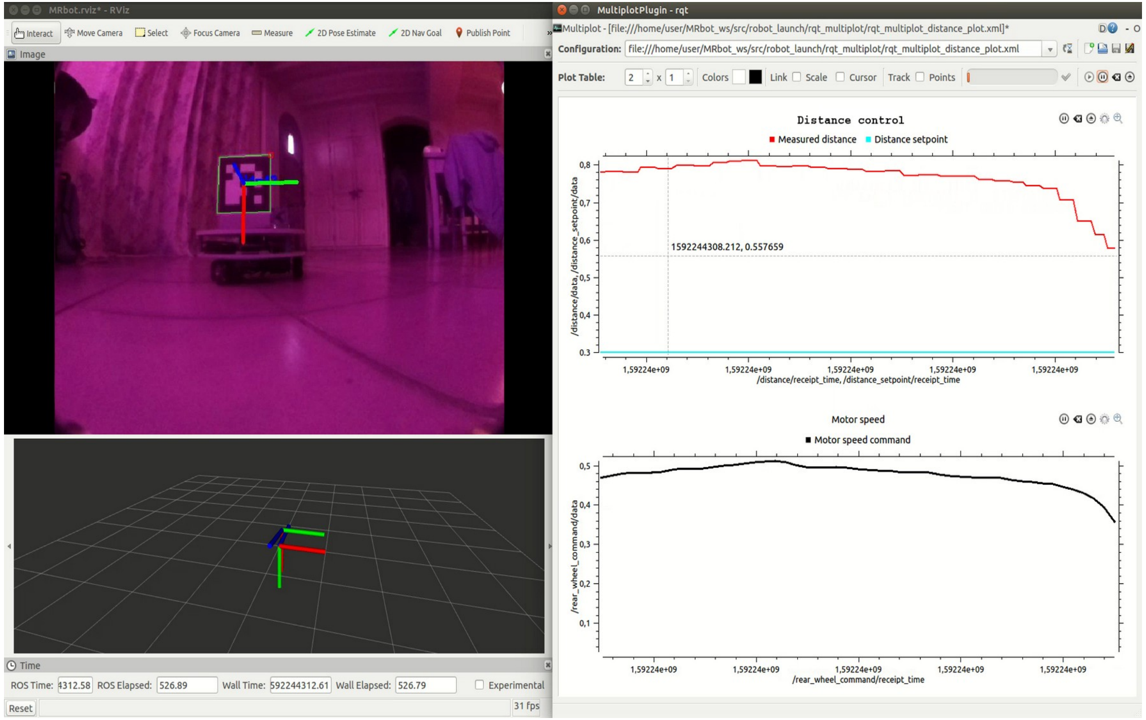

Figure 3 : Debugging interface

Software capabilities

Dynamically change distance setpoint
Dynamically modify PID parameters
Pose estimation of the leading vehicle overlaid on the video stream
Tracks timestamps for each step of the computation (from observation to action)

Execution time experiment

Goal :

Check the differences in behaviour concerning the execution time of each computation step when using real (*T3* in Figure 3) or simulated observations (*T1* and *T2* in Figure 4).

Observations :

When using simulated observation, the processing time is lower.

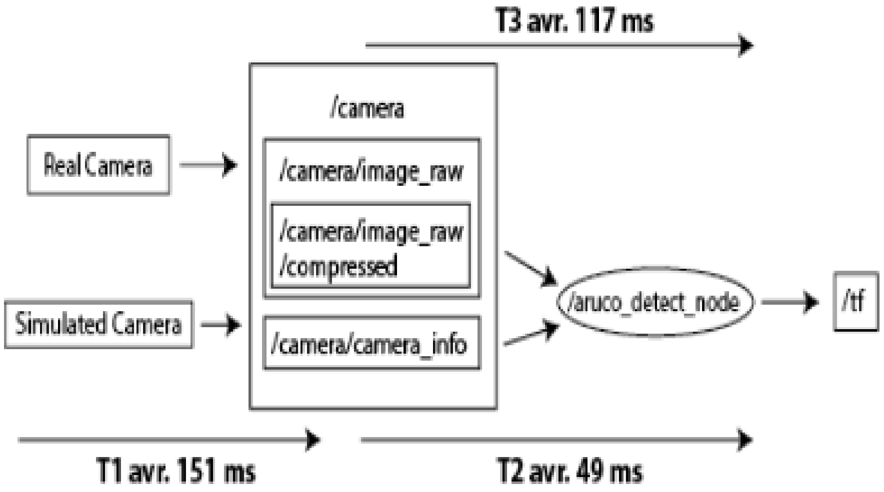

Figure 4 : ROS computation graph along with measured computation time.

Aruco_detect_node is the computer vision algorithm to estimate the leading vehicle pose (here called /tf).

Conclusion and future work

Conclusion :

- Our architecture and miniature HIL platform proved to be an economical yet effective solution for preliminary studies or the architecture.
- We quantified the delay contribution of each computation step.
- We believe the difference in computation time is due to the lower complexity of the simulated camera feed.

Future work :

- Quantify dynamical differences between the simulated car and its real counterpart.
- Adding additional sensors to get “ground truth” measurements to improve the simulation accuracy.
- Investigate using ROS2 to lower latency.
- Investigate the effect of various levels of noise on the computation time.