Retour d'expérience sur l'utilisation de ROS en robotique mobile (Flotte de Turtlebot 3 et véhicule routier)

Gérald Dherbomez - CRIStAL UMR CNRS 9189
gerald.dherbomez@univ-lille.fr

Journée technique ²RM Robotique Mobile à roues
Clermont Ferrand - 4 juillet 2018

Mon historique avec ROS

- 2011 : encadrement d'un projet d'étudiant de l'UTC sur l'interfaçage entre ROS (Diamondback/Electric) et PACPUS (framework local)
- 2015/2016 : quelques projets d'étudiants sur la plateforme véhicules autonomes à Heudiasyc
- 2017 : utilisateur sporadique de ROS à CRIStAL
- Depuis janvier 2018 : utilisation de ROS pour les projets en cours de la plateforme ***PRETIL*** (***P****lateforme de recherche* ***R****obotique* ***E****t* ***T****ransports* ***I****ntelligents de* ***L****ille*) :
  - Mise en place d'une flotte de Turtlebot 3
  - Acquisition de données capteurs sur véhicule automobile à l'aide de ROS
  - Utilisation de rosjava pour le pilotage de robots à partir du Cloud avec le middleware OCCIware

Contexte

FTF (Fault Tolerant Fusion)
FTC (Fault Tolerant Control)

- Flotte de robots mobiles Indoor
  - Projet PEPS **LOCABOTS** (“LOcalisation CollAborative tolérante aux fautes pour un système multi-roBOTS aéroterrestre avec des capteurs bioinspirés”) avec l’ISM à Marseille.
  - Travaux de thèse de Boussad Abci, encadré par Maan El Badaoui El Najjar et Vincent Cocquempot.
    - Fusion et commande tolérantes aux fautes pour la localisation collaborative d'un système multi-robots.

- Acquisitions de données capteurs sur véhicule automobile
  - Projet **SMARTIES** (“Systèmes de communications et de localisations fiables et intelligents”) du CPER ELSAT2020 (“Écomobilité, Logistique, Sécurité et Adaptabilité dans les Transports à l’Horizon 2020”).
  - Travaux en localisation de Nourdine Ait Tmazirte (IR)
  - Travaux de thèse de Khoder Makkawi, encadré par Maan El Badaoui El Najjar
    - Fusion tolérante aux fautes pour le couplage serré tolérant aux fautes des données GNSS/INS/vision/carto3D

CRIStAL
Centre de Recherche en Informatique, Signal et Automatique de Lille

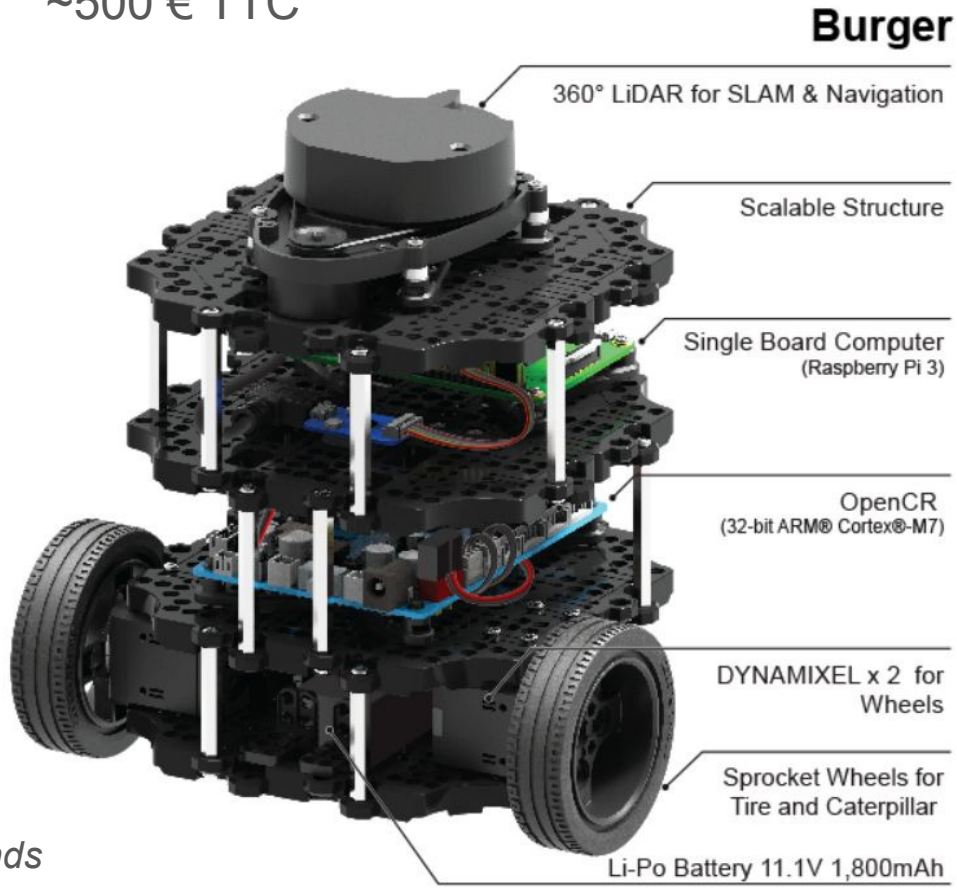
Flotte de robots mobiles Turtlebot 3

Le Turtlebot 3

~500 € TTC

- Full ROS
- Ouvert :
  - Codes PC, Raspberry et Arduino (rosserial) open source
  - Open hardware (électronique de la carte OpenCR et la CAO mécanique)
- 2 à disposition de ²RM (CRIStAL et LAAS)

Turtlebot3 friends



Flotte de Turtlebot 3 : Nos besoins

Besoins scientifiques :

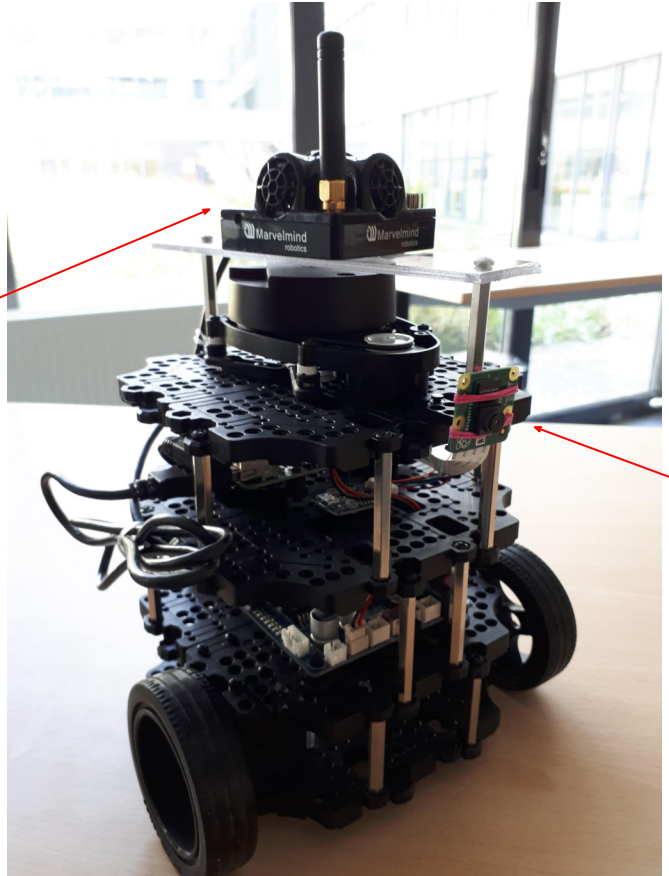
- Source de données redondantes
- Système Multi-robots présentant des défaillances

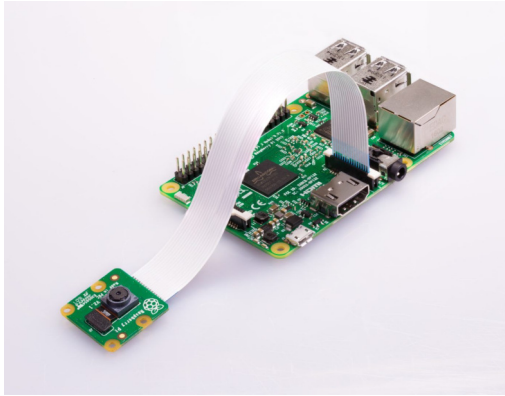
Besoins technologiques :

- Multiples sources de localisation
- Possibilité de générer des fautes capteurs et actionneurs
- Perception et localisation des robots entre eux

Adaptation du Turtlebot 3

Ajout d'un module de localisation Marvelmind





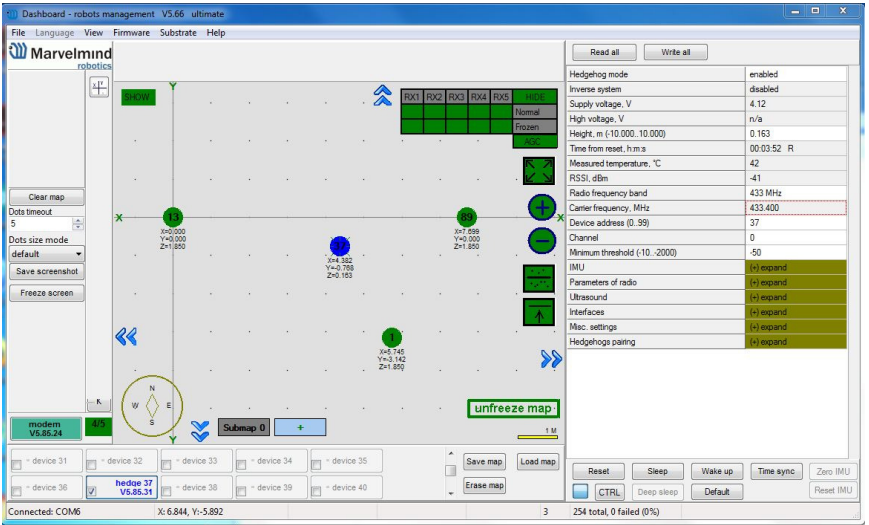
Ajout d'une caméra de détection des autres robots Camera Module V2 (Sony IMX219 8-megapixel sensor)

Focus sur le marvelmind

Principe : **“GPS Indoor”**

HIDE	13	65	79	89	91
13		4.092	10.148	9.160	
65	4.092		9.097	10.335	
79	10.148	9.097		5.271	
89	9.160	10.335	5.271		
91	6.583	6.372	4.035	4.649	

- Triangulation d’émetteurs ultrasonores (5 par balise)
- Échanges des données brutes par radio (433 ou 900MHz)
- Précision : **+/- 2cm** (différentiel) ou en absolu, 1-3% de la distance à la balise
- Fréquence : **25/n Hz** où n = nombre de noeuds mobiles
- Création d’une carte de balises fixes (verts) et positionnement des mobiles (bleu) à l’intérieur



Attention :

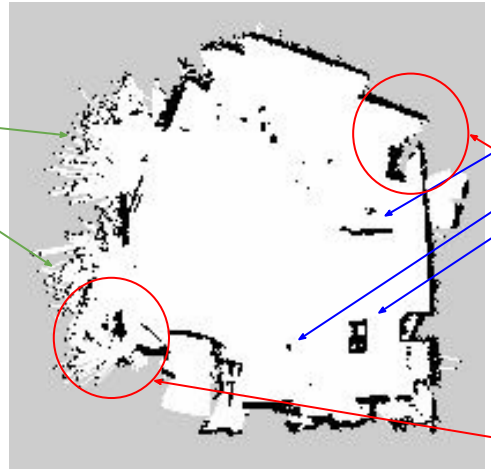
- La précision sur le paramétrage des hauteurs des balises fixes est essentielle pour la localisation en 3D.

- Sensibilité aux perturbations sonores de l’environnement et aux interférences radio (4 canaux disponibles)

SLAM avec gmapping

Bureau, planche correctement modélisés

Problème avec les fenêtres



Problème de fermeture de boucle dû à des perturbations Lidar (câbles électriques verticaux, pieds de table, …)

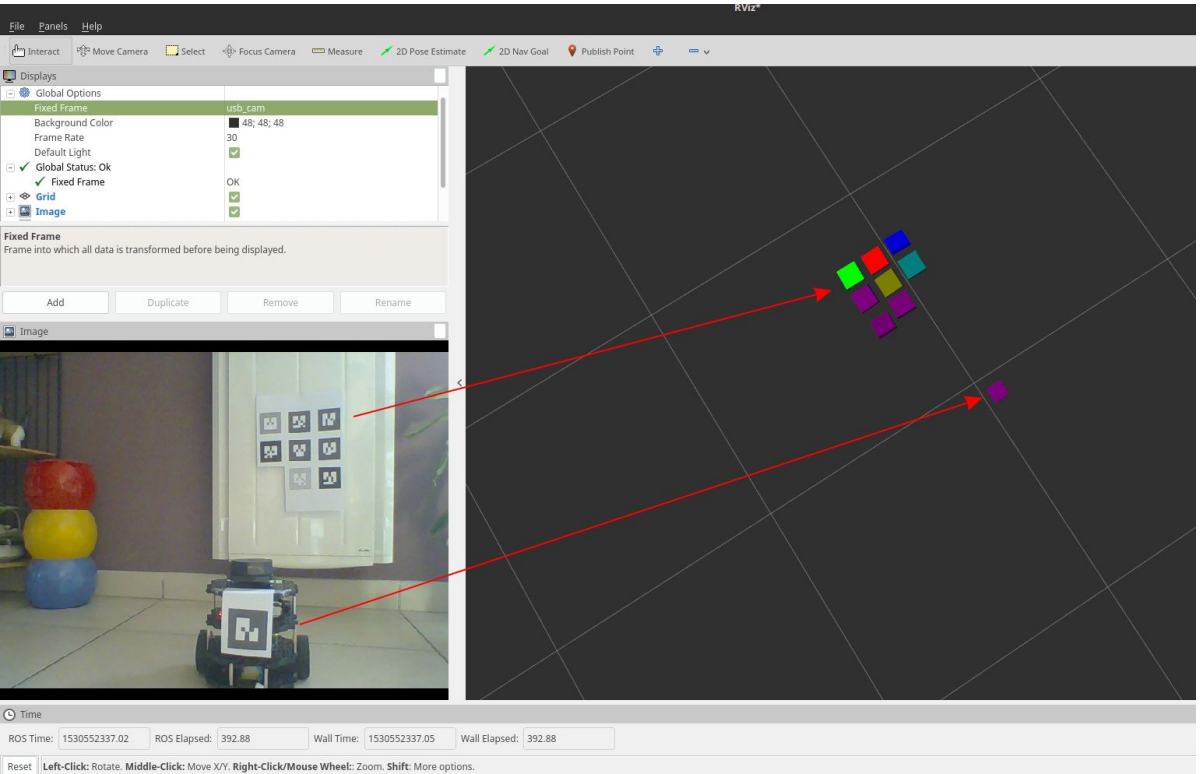
Remarques :

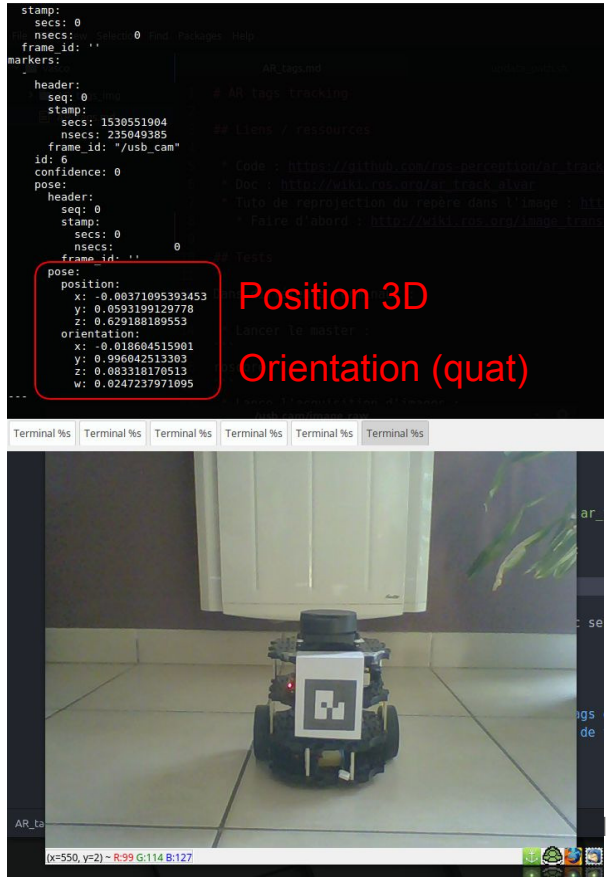
- Privilégier des zones bien nettes pour le LIDAR (murs, plans, …)
- Robustesse de l'algo de navigation qui fonctionne néanmoins même avec une carte déformée
- Faciliter de mise en oeuvre d'applications de SLAM grâce à ROS !

Perception des robots : AR_TRACK_ALVAR

http://wiki.ros.org/ar_track_alvar

Détection de tags AR par vision.



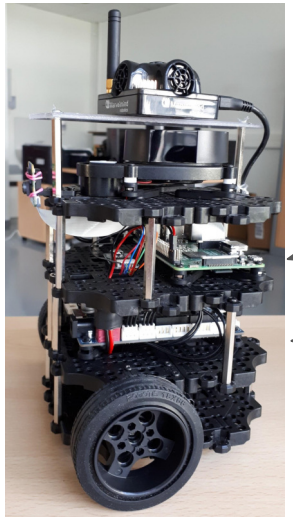


Architecture informatique du système multi-robots

PC Linux (Applis ROS)

```
1.  roscore
2.  roslaunch turtlebot3_navigation turtlebot3_navigation.launch map_file:=$HOME/map_iut.yaml
3.  roslaunch ar_track_alvar usb_cam.launch
```

192.168.1.100



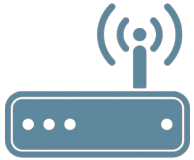
192.168.1.101
192.168.1.102
192.168.1.103

Rpi3

```
roslaunch turtlebot3_bringup
  turtlebot3_robot.launch
```

Pilote l'Arduino avec rosserial

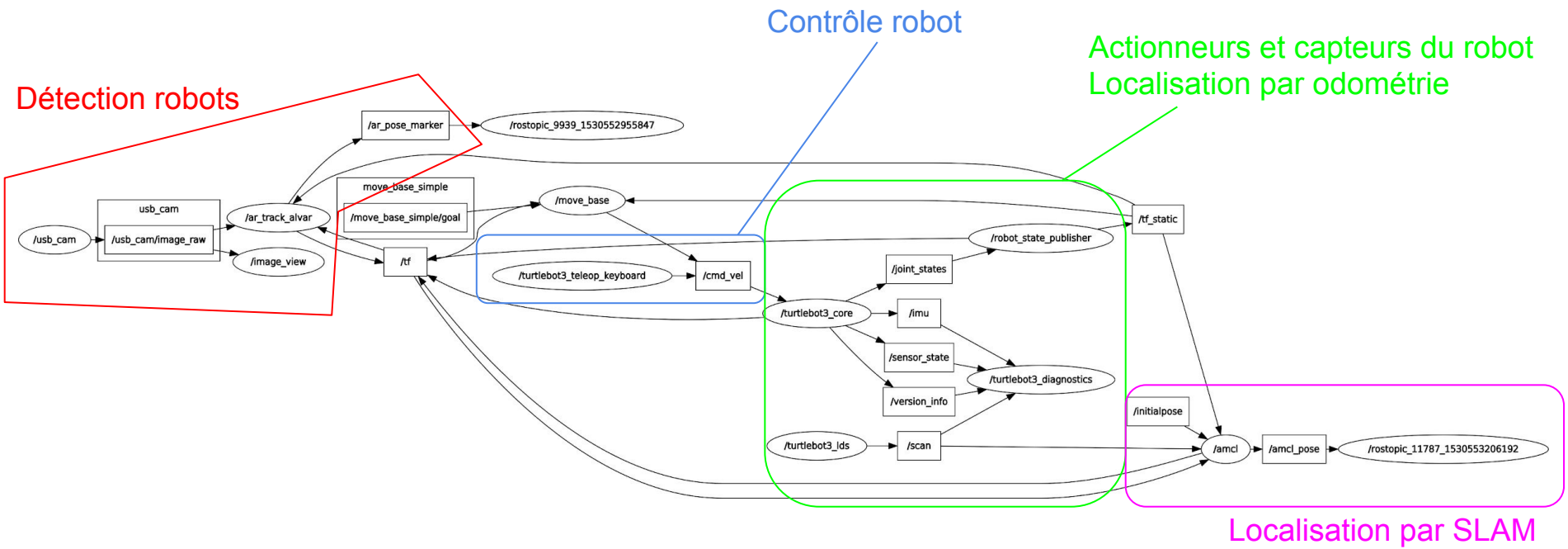
X 3



192.168.1.130

PC Matlab
(commande et fusion)
Algos FTC et FTF

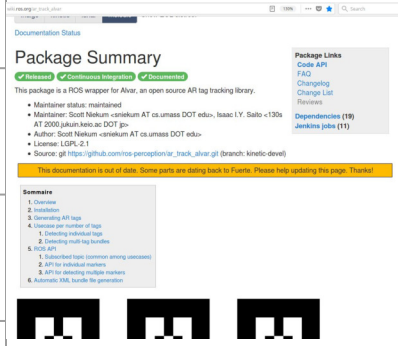
Rqt_graph mono-robot



\+ *Localisation par Marvelmind*

Synthèse des problèmes rencontrés et pistes

Problème	Piste(s) à étudier
Imprécision de la création de la carte pour la navigation par SLAM	Changer l'algorithme de SLAM (gmapping → hector ?) Utiliser un capteur de meilleur précision (RPLIDAR A2)
L'algo de localisation SLAM non utilisable sur le plan scientifique (car source de données communes /odom et /amcl_pose)	Identifier une méthode de localisation basée LIDAR uniquement
Documentation parfois obsolète, exemple ar_track_alvar	Capitaliser au sein de la communauté Remonter l'information aux développeurs
Quid des performances réseau en mode multi-robots + contrôle déporté sur Matlab	Passer sur un fonctionnement multi-masters
Problème d'horloge non à jour sur la Rpi3 du Turtlebot ⇒ timeout pour le SLAM	`sudo timedatectl set-time "YYYY-MM-DD HH:MM:SS"` Ou synchro NTP Ou connexion Internet

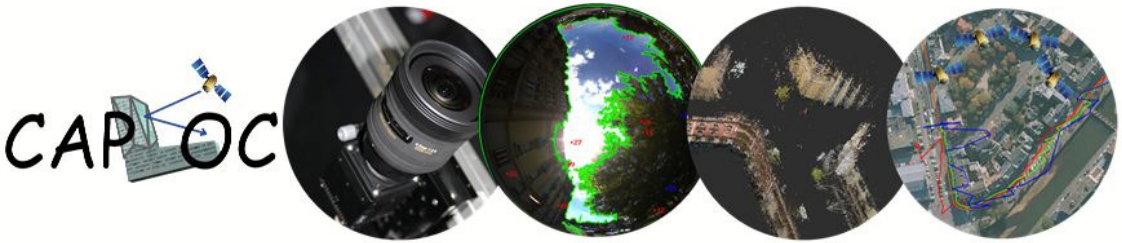


Acquisition de données multi-capteurs sur un véhicule automobile en environnement urbain

Problématiques du projet SMARTIES

Le concept CAPLOC

Une solution bas coût, sans infrastructure au sol, reposant sur les signaux satellitaires et une connaissance de l'environnement local de propagation de ces signaux



La connaissance de l'environnement donne une information sur la qualité de la mesure et peut être offerte par le traitement d'images/vidéo

A partir du Concept PREDISSAT (2002) breveté en 2007

www.ifsttar.fr

Pour les transports terrestres (ferroviaire et automobile) :

- Approche de **positionnement par filtrage informationnel** (variante du filtre de Kalman) et **couplage serré GNSS/IMU/Odométrie/Image** pour fournir une localisation tolérante aux fautes (multi-trajets, satellites en défaut, dérives inertielles)

- Mise en oeuvre de CAPLOC

Source : Juliette Marais, Ifsttar/Cosys/Leost

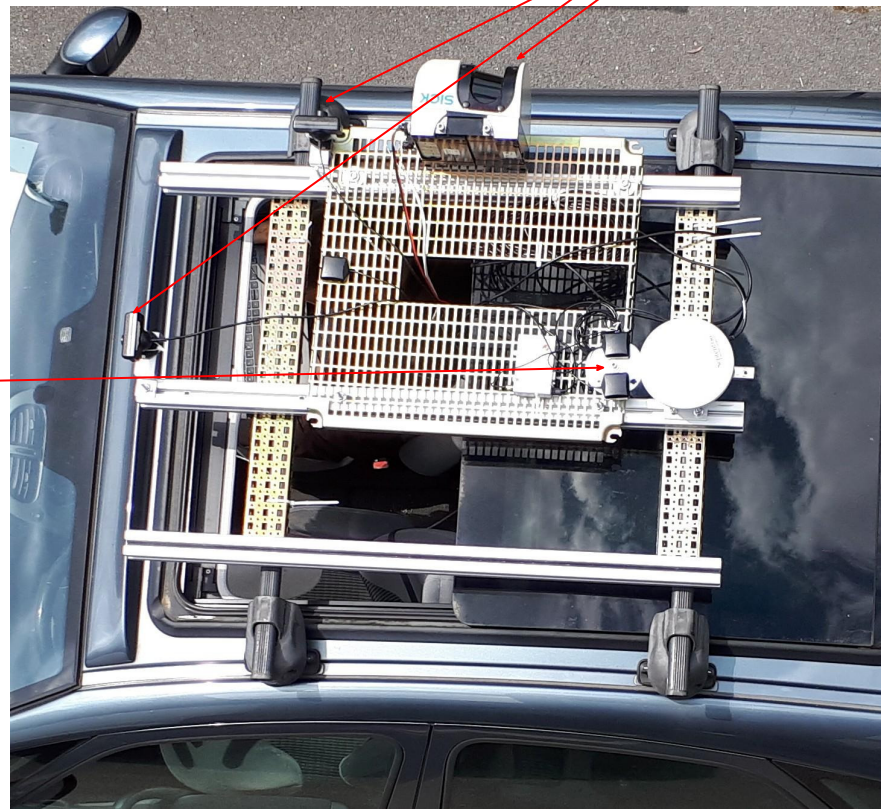
Instrumentation et acquisition de données

- Véhicule en attente de réception (Renault Zoé robotisée par FH Electronics)
- Capteurs :
  - Septentrio AsterX4 en RTK : **nmea_topic_serial_reader** de https://github.com/ros-drivers/nmea_navsat_driver + logiciel RxControl de Septentrio (**NTRIP Clien**t et enregistrement RINEX)
  - 2x Ublox M8T : un récepteur pour la synchro + logiciel Ucenter de Ublox (RINEX)
  - 1x Ublox M8U : logiciel Ucenter de Ublox (RINEX)
  - Lidar Sick LMS 511 : https://github.com/clearpathrobotics/LMS1xx
  - 2 caméras USB (Sony PS3) : https://github.com/ros-drivers/usb_cam
  - IMU OS3D de Inertial Labs : https://github.com/gdherbom/imu_os3d
  - CAN KVASER : https://github.com/astuff/kvaser_interface
- Moyens informatiques :
  - PC embarqué ADLINK MXC-6401 sous Linux Mint 18.04, ROS Kinetic
  - Portable Windows pour les logiciels Septentrio et Ublox
  - Partage de connexion Internet du smartphone ⇒ Récupération du **flux RTCM** issu de la **station de base IGN** LIL2 (http://rgp.ign.fr/STATIONS/#LIL2)

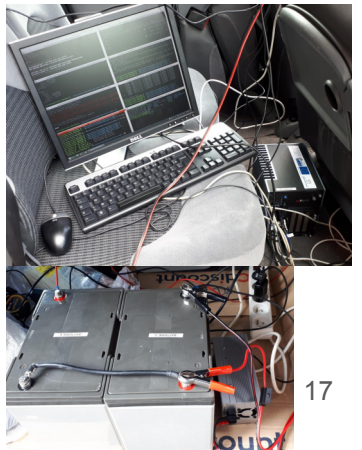
Campagne du 13 juin 2018

LIDAR + caméras
(scène + bâtiments)

IMU + Antennes GPS



Système d'acquisition ROS
(DC24V)



Synchronisation temporelle

Problématique : base de temps commune entre les bagfiles ROS et les données RINEX des récepteurs.

Solution : synchroniser l'horloge PC sur le temps GPS **(intérêt : fonctionne sans Internet)**

- gpsd + chronyd ⇒ date le PPS au niveau des interruptions du noyau linux
- RINEX : datation en temps GPS
- Mise en oeuvre :
  - Récepteur Ublox M8T générant un front PPS et une trame ZDA, connecté en port série (pas d'USB) sur le PC.
  - Configuration et lancement des démons gpsd et chronyd qui communiquent par socket.

source

```
Every 2,0s: chronyc sources                                  Tue Mar 29 10:50:58 2016

210 Number of sources = 3
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
#- PPS                           0   4   177     8    +20us[  +20us] +/-   50ms
#- GPS                           0   4   177     8    -52ms[  -52ms] +/-  200ms
#* PREC                          0   4    77    21  -8211ns[  -59us] +/- 7116ns
```

précision

Analyse des données de positionnement

Synchronisation temporelle :

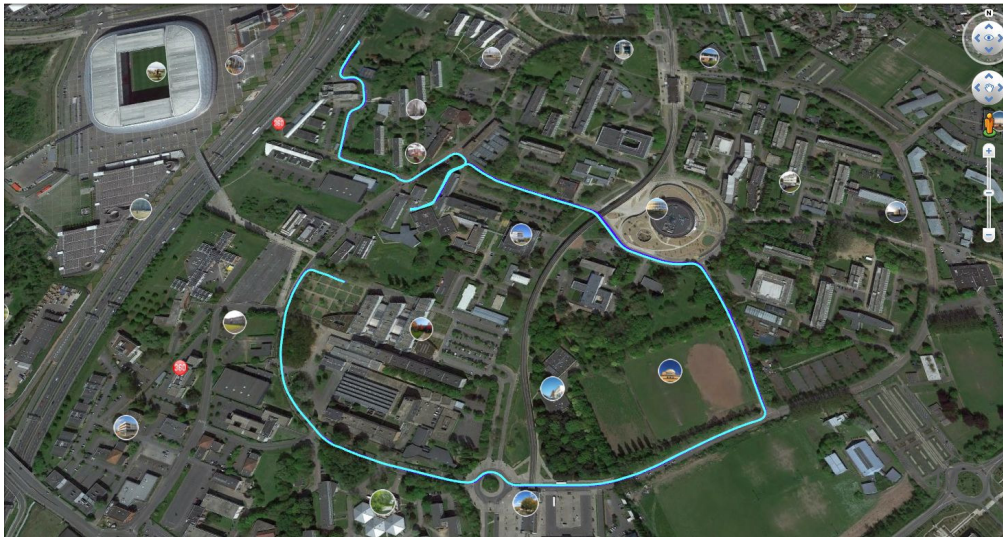
```
>> NMEA.Sentence(mess-13)
ans =
    '$GPZDA,135557.90,13,06,2018,,*60'
>>
datetime(NMEA.Time(mess-13),'ConvertFrom','posixtime','Format','HH:mm:ss[.SSS]Z')
ans =
   13:55:57[.905]*
```

Précision du RTK :

- En cours d'analyse (sous Matlab), centimétrique à première vue.
- Disponibilité du RTK très bonne malgré l'environnement adverse (végétation, bâtiments) : 85 à 88%
- Récupération d'un "RTK Fixed" très rapide (1 à 2 sec en moyenne)

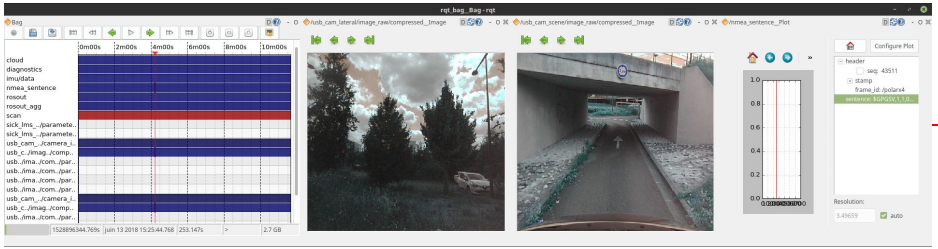
5 msec de décalage entre le temps UTC de la trame ZDA et le timestamp ROS :

Délai de transmission de la ZDA (32 octets) sur le port série :
1000 x 32 x 8 / 115200 = 2.2 msec (+ temps de traitement du noyau et de ROS) ⇒ **~5msec**



Cas d’étude, récupération d’un fix après masquage total

Approche du tunnel : 13h25m44s.800 (ind_qual_GGA = 4)



Entrée dans le tunnel : 13h25m46s.800 (ind_qual_GGA = 5)

Masquage total : 13h25m48s.500
(ind_qual_GGA = 0)

Récupération DGPS : 13h25m50s.100
(ind_qual_GGA = 2)

Récupération RTK Fixed : 13h25m56s.100
(ind_qual_GGA = 4)

Retour sur l'utilisation de Matlab en lien avec ROS

- Pros :
  - Matlab souvent présent dans les laboratoires et universités
  - Répond au besoin des automaticiens fervents adeptes de Matlab
  - Facile à mettre en oeuvre (Robotics System Toolbox) + connexion du PC au réseau ROS (le PC Matlab peut faire office de master) https://fr.mathworks.com/products/robotics.html
  - Disponibilité des toolbox de Matlab
  - Utilisable en post-traitement de données (lecture des bagfiles) et en temps réel pour le contrôle des robots (génération et lecture de messages sur les topics)
- Cons :
  - Toolbox payante
  - Dans certains cas, notamment pour les flottes de robots, charge réseau problématique
  - Favorise les biais de programmation Matlab (manque d'organisation du code, manque de factorisation, code de maquettage non capitalisable)

Synthèse sur les problèmes rencontrés et pistes

Problème	Piste(s) à étudier
Exploitation des données LIDAR. Pas de frame_id dans les bagfiles	https://github.com/srv/srv_tools Attention bag_tools ne s'installe pas par les dépôts Ubuntu
Essais du noeud de décodage ublox (format UBX) infructueux	Etudier la doc plus en détail (http://wiki.ros.org/ublox) ou tester un autre package : https://github.com/tu-darmstadt-ros-pkg/ublox
Pas de mesure odométrique disponible (bus VAN)	Odométrie visuelle ??
Exploitation des données pour des traitements : position des capteurs non modélisé sous forme d'URDF, pas de TF publié.	Recherche d'une solution de calibration extrinsèque

Prochaines étapes

Turtlebot 3 :

- Contrôle simultané de la flotte sous Matlab (utilisation des namespaces ROS)
- Génération de défauts
- Perception des robots les uns par rapport aux autres
- Portage des travaux sur une flotte de robots outdoor (Jaguar)

Véhicule routier :

- Arrivée en 2019 de la Zoé robotisée (prestation FH Electronics) et portage du système actuel.
- Analyse des données de localisation (RINEX + filtre GNSS/IMU sous GoGPS http://www.gogps-project.org/)
- Nouvelle campagne d'acquisition avec installation d'une caméra 360° sur le toit du véhicule pour alimenter CAPLOC (fourniture IFSTTAR COSYS/LEOST : système d'acquisition sur FPGA)

Globalement :

- Généralisation de l'utilisation de ROS sur nos plateformes robotiques
- Interrogation de la compatibilité de ROS pour certains projets (ex : robot médical)


Merci pour vos questions !