

# Etude et développement d'outils mathématiques pour estimer, en temps réel, le tassage et le volume d'un silo de maïs à partir de capteurs embarqués

Point Mi-stage  
Sébastien Hervieu  
12 Juillet 2018



**Tellus**  
ENVIRONMENT  
Cartographier l'invisible pour agir

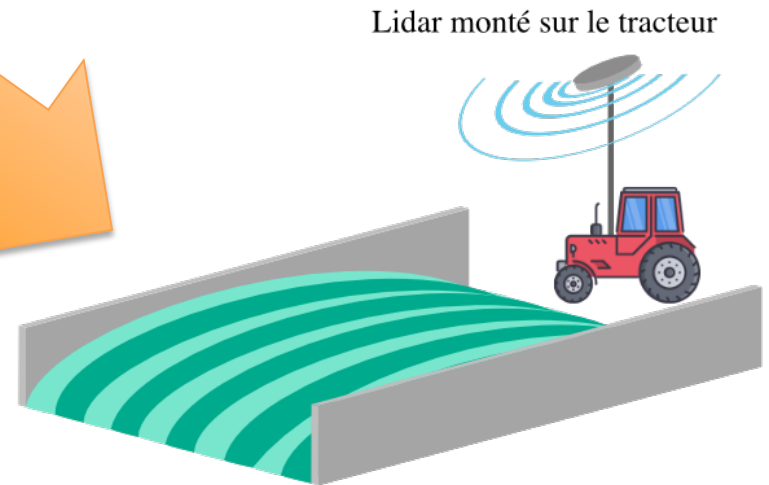
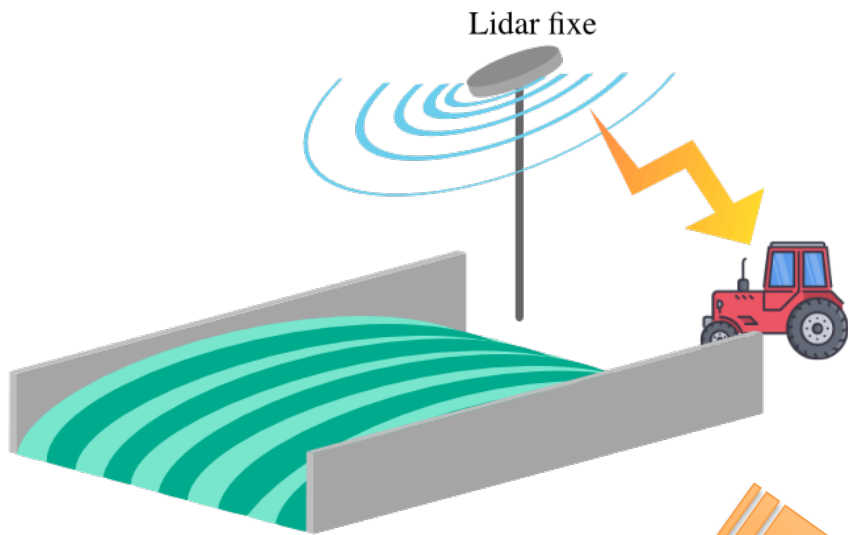
# Plan

- Tellus Environnement
- Progression Stage
- Démonstration
- Questions

# Tellus Environment

- Startup spécialisée dans le relevé et la cartographie de sous-sols et milieux sous-marins et aquatiques
- Petite activité R&D:
  - Nouveaux outils de relevé (Drones, Relevés Multicanaux)
  - **Agriculture de précision**

# Symeter 2



# Environnement de développement

- Prototypage d'un système de mesure largement autonome
  - Logiciel sous ROS
  - Plateforme Hardware Simulée sous Gazebo
- Le HW simulé permet:
  - Initier la chaine de traitement
  - Simuler des configurations HW variées
  - Identifier les axes d'améliorations pour augmenter la fiabilité du procédé.
  - Evite Moyens « lourds ».

# Démarche du Stage

- Apprentissage / Faisabilité ROS – Gazebo
- Implémentation « Tracteur Virtuel »
- Montage des modules ROS pour Symeter 2
  - Localisation (fusion IMU + GPS)
  - Acquisition LIDAR
  - Traitement des données
  - Pilotage logique prestation (GUI, Stockage, coordination)

# ROS – Plateforme Robotique

- Robot OS – Open Source
- Basé sur un ensemble de process
- Interfaces standardisées, simples à déployer
  - Tf: Transformations – positionnement des éléments internes du robots, et du robot dans le monde
  - Com interprocess continue (topics) ou interactionnelle (services)
- Extrêmement modulaire
  - De très nombreux capteurs intégrés
  - De nombreux modules existants pour assurer localisation, traitement des données, etc

# Gazebo

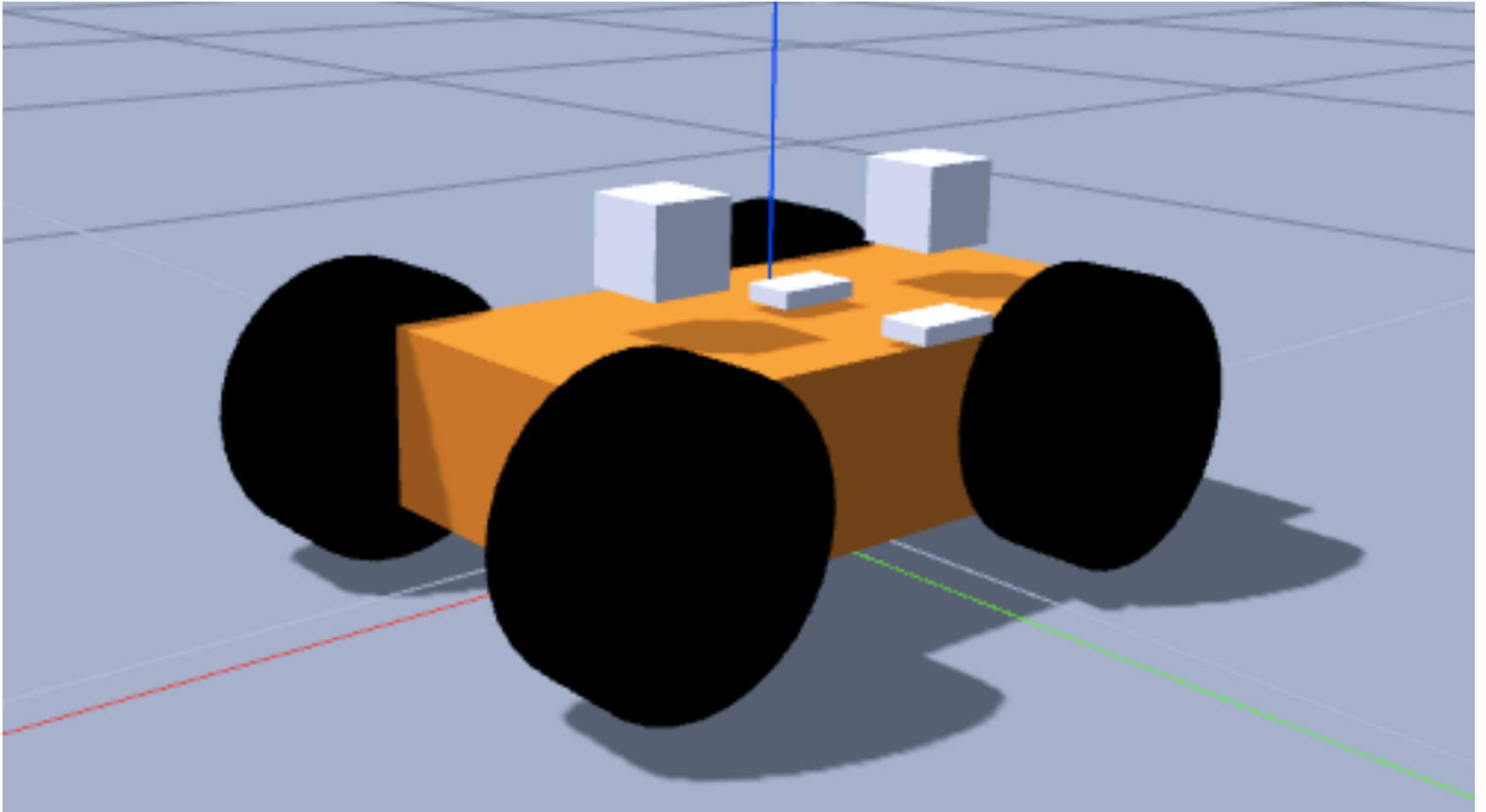
- Simuler un environnement « physique » dans lequel évolue un robot virtuel
  - Spécification d'un « monde » virtuel
  - Interaction du robot avec son environnement
    - Capteur (Lidar, GPS, IMU, Caméra, etc)
    - Actuateurs (moteurs, actionneurs,...)
    - Physique (Gravité, friction, ...)
- Interaction fine avec ROS



# Apprentissage / Faisabilité ROS - Gazebo

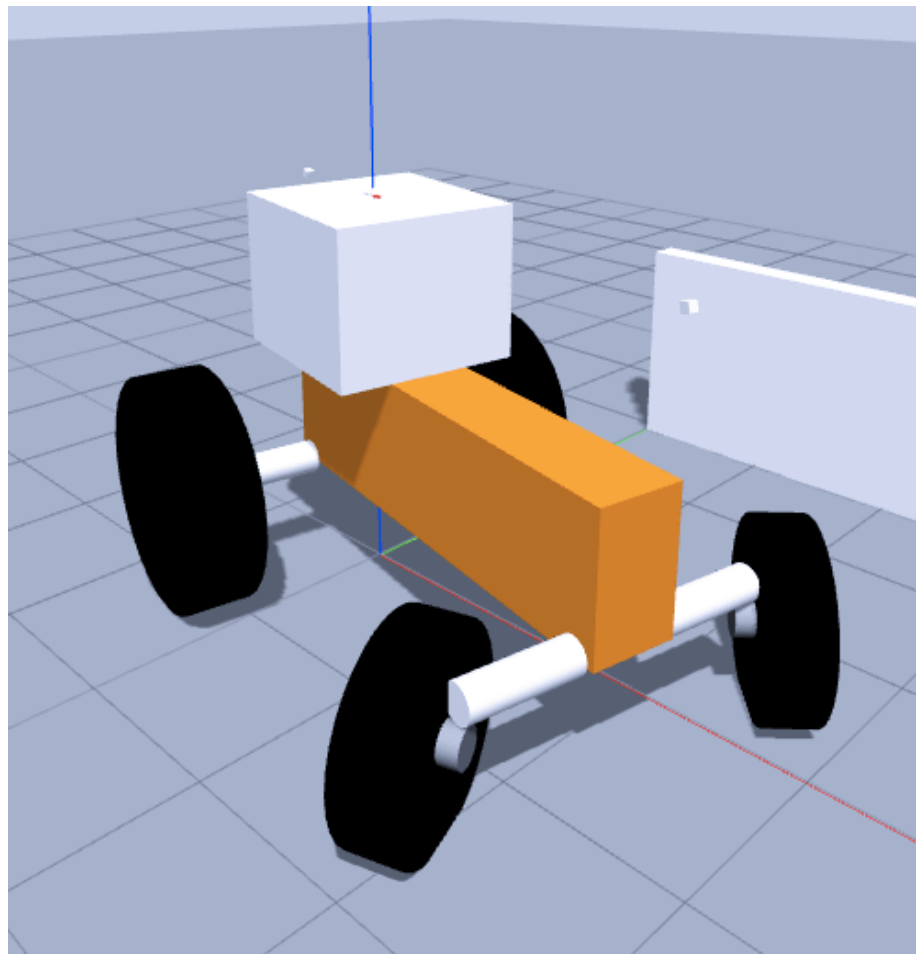
- Monter une simulation d'un robot simple
  - Acquérir l'expertise nécessaire
  - Vérifier la disponibilité des modules nécessaires à simulation Symeter 2
    - Véhicule téléguidable
    - Mise en œuvre Lidar simulé
    - Mise en œuvre IMU simulé
    - Mise en œuvre GPS simulé
- Contournement de bugs et limitations de Gazebo
  - Contrôleur PID, gestion de la friction capricieuse

# Robot Préliminaire



# Tracteur Simulé

- Plateforme ayant taille et vitesse d'un Tracteur Réel
  - Propulsion et direction (différentiel, Ackerman)
  - Implantation des capteurs
- Description par fichier XML (URDF)
  - De nombreux contournements à mettre en place

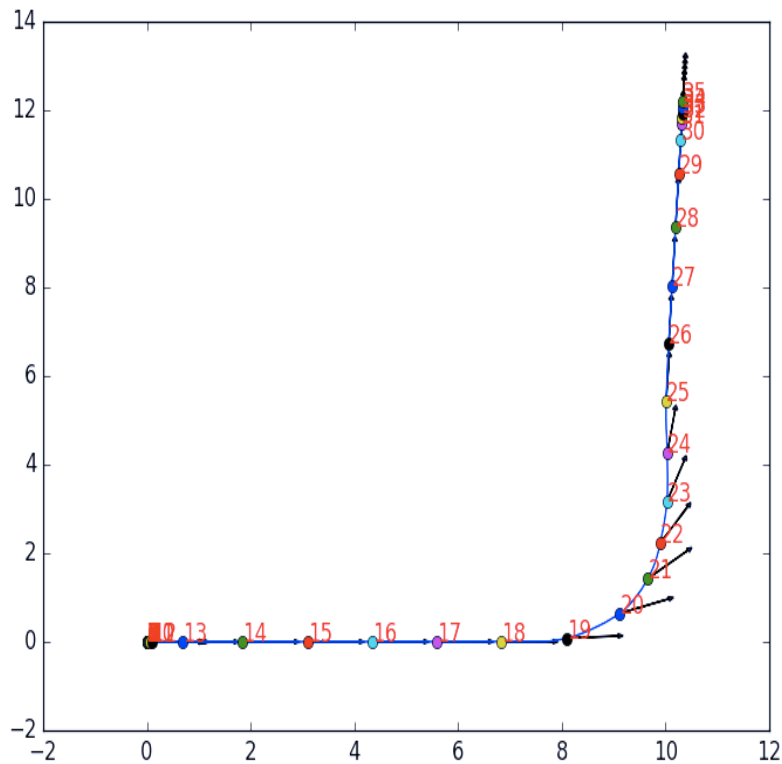


# Localisation

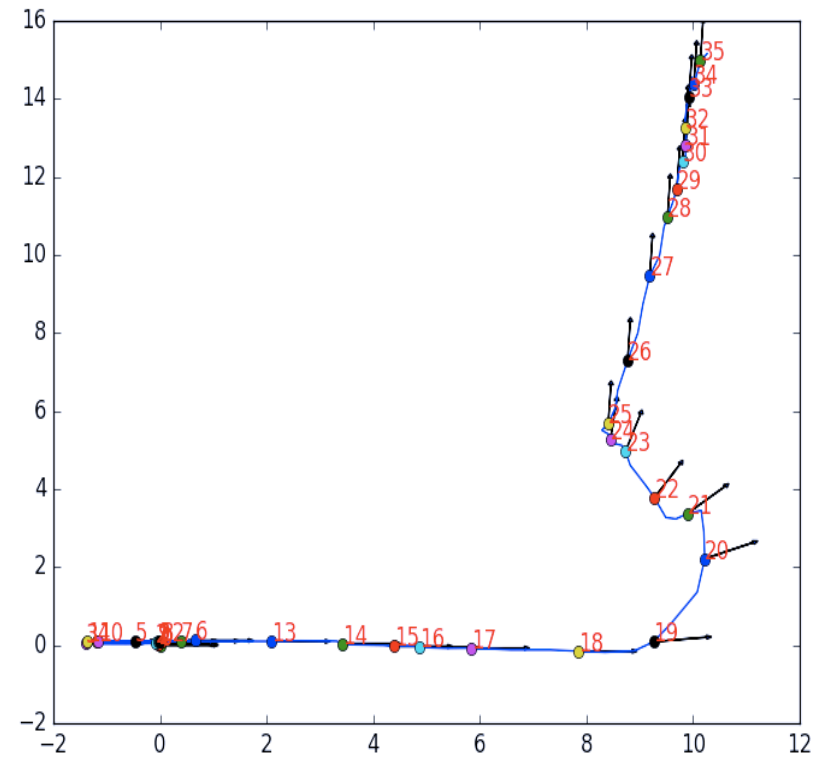
- Fusion de Données:
  - IMU (assiette, accélération linéaire)
  - GPS (coordonnées terrestres, précision  $\sim 10\text{cm}$ )
  - Obtenir estimation précise Pose du tracteur
- Mise en œuvre d'un node ROS:  
`ekf_localisation_node`
- Développement scripts Python pour analyse des tests

# Localisation

- Parcours Réel



- Parcours Estimé



# Localisation: Axes d'amélioration

- Mieux utiliser l'IMU
- Chercher un IMU simulé moins bruité
- Utiliser des données issues d'un IMU réel.

# Traitement Lidar

- Un Lidar génère beaucoup de données
  - Filtrage nécessaire
- Reconstitution de la scène
  - Carte d'occupation 3D
  - Octree
- Point Cloud Library



RViz\*

Interact

Move Camera

Select

Focus Camera

Measure

2D Pose Estimate

2D Nav Goal

Publish Point

Displays

Global Options

Fixed Framebase\_link

Background Color48; 48; 48

Frame Rate30

Default Light☒

Global Status: Ok

Fixed FrameOK

Grid☒

LaserScan☐

RobotModel☒

TF☒

PointCloud2☒

Status: Ok

Topic/symmeter2/cloud/filter...

Unreliable☐

Selectable☒

StyleFlat Squares

Size (m)0,01

Alpha1

Decay Time0

Position Transfo...XYZ

Color TransformerIntensity

Queue Size10

Channel Nameintensity

Use rainbow☒

Invert Rainbow☒

Min Color0; 0; 0

Max Color255; 255; 255

Autocompute In...☒

Min Intensity999999

Max Intensity999999

PointCloud2☐

Topic/octomap\_point\_clo...

Unreliable☐

Selectable☒

Add

Duplicate

Remove

Rename

Views

Type:Orbit (rviz)Zero

Current ViewOrbit (rviz)

Near Clip ...0,01

Invert Z Axis☐

Target Fra...<Fixed Frame>

Distance35,2276

Focal Shap...0,05

Focal Shap...☒

Yaw0,587223

Pitch0,609797

Focal Point-1.8782; 9.748; -...

Save

Remove

Rename

Time

ROS Time:29.10ROS Elapsed:28.91Wall Time:1531383669.04Wall Elapsed:78.69

Reset

Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel:: Zoom. Shift: More options.

31 fps

# Progression du stage

- Nous partions d'une feuille blanche
- Le procédé de base est maintenant disponible
- Maintenant faire des tests:
  - Positionnement des capteurs
  - Amélioration de la Localisation → précision
  - Améliorer les traitements PCL → précision et performance

# Démonstration

Questions?

# Et les Maths dans tout ça?

- Géométrie (« Pose » )
  - Coordonnées Homogènes, Géométrie Projective
  - Quaternions Unitaires (rotations)
- Estimation – Filtres de Kalman
- Nuages de Points
  - Filtrage (Voxel grid)
  - Interpolation
  - OctoMap (Map d'occupation probabiliste)
  - Extraction de paramètres (RANSAC)

# Etude préliminaire Lidar

- Etudier l'influence de l'inclinaison d'un lidar par rapport à la surface
- Etude bibliographique:
  - Matériaux granulaire répondent bien même si forte inclinaison
- Simulation sous Spyder:
  - Etudier la taille (forme) du faisceau laser suivant l'inclinaison par rapport à la cible
  - Géométrie projective