

# Guidage *Teach-and-Repeat* pour un robot mobile tout-terrain avec LiDAR



***Philippe Giguère***, *ing. jr*

Professeur Associé, Informatique et génie logiciel

***David Landry***

Stagiaire 1<sup>er</sup> cycle, Informatique



UNIVERSITÉ  
**LAVAL**

# Aperçu présentation

---

- Problème
- Guidage autonome *Teach-and-Repeat*
- Plateforme robotique
- Expérimentations



# Problème

- Comment automatiser les déplacements de véhicules en milieu forestier, en l'absence de
  - routes
  - balises visuelles ou radio
  - GPS

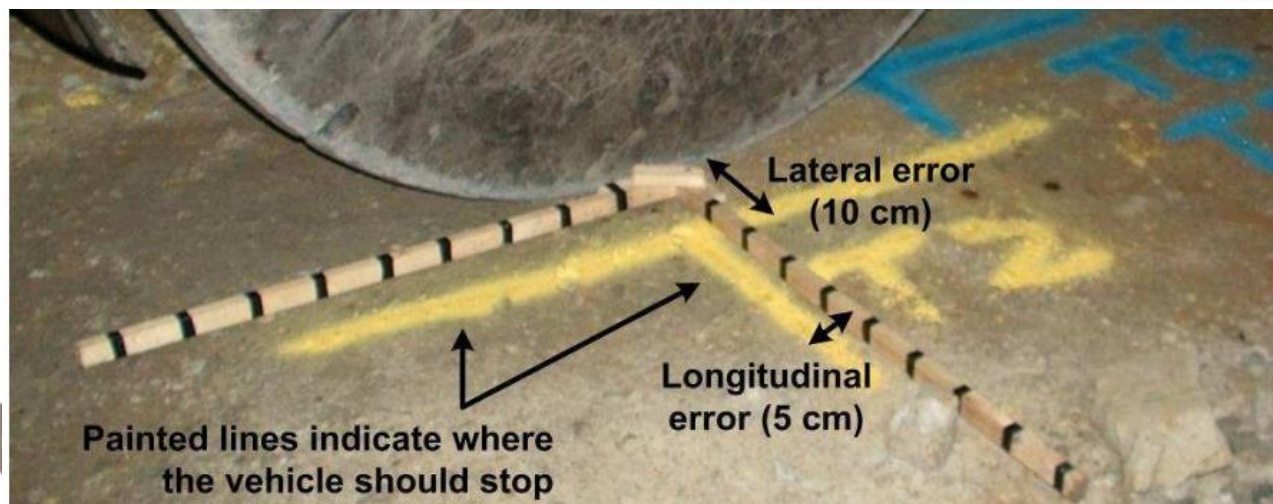
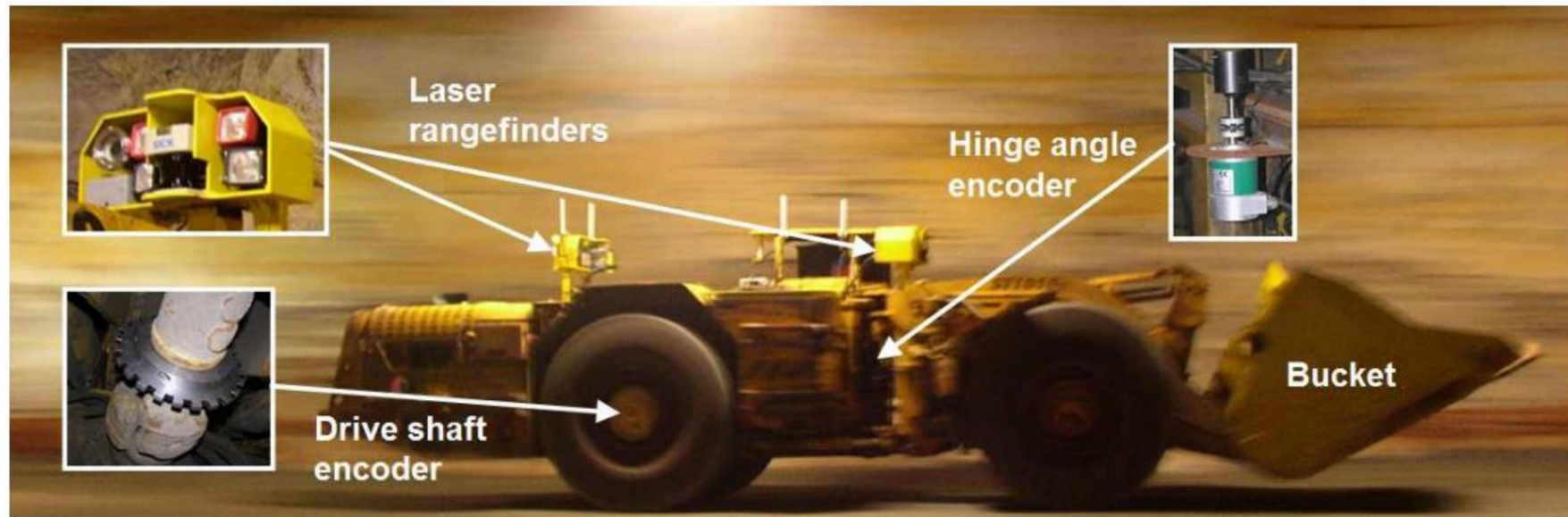


# Tramway virtuel avec *Teach-and-Repeat*



# Solution : Tramway virtuel (*Trackless Trammimg*)

- Équivalent de déposer un rail virtuel au sol



Atlas Copco Infrastructureless Guidance System for High-Speed Autonomous Underground Trammimg, J. Larsson, J. Appelgren, J. Marshall, T. Barfoot, *Tech Report*, 2008.

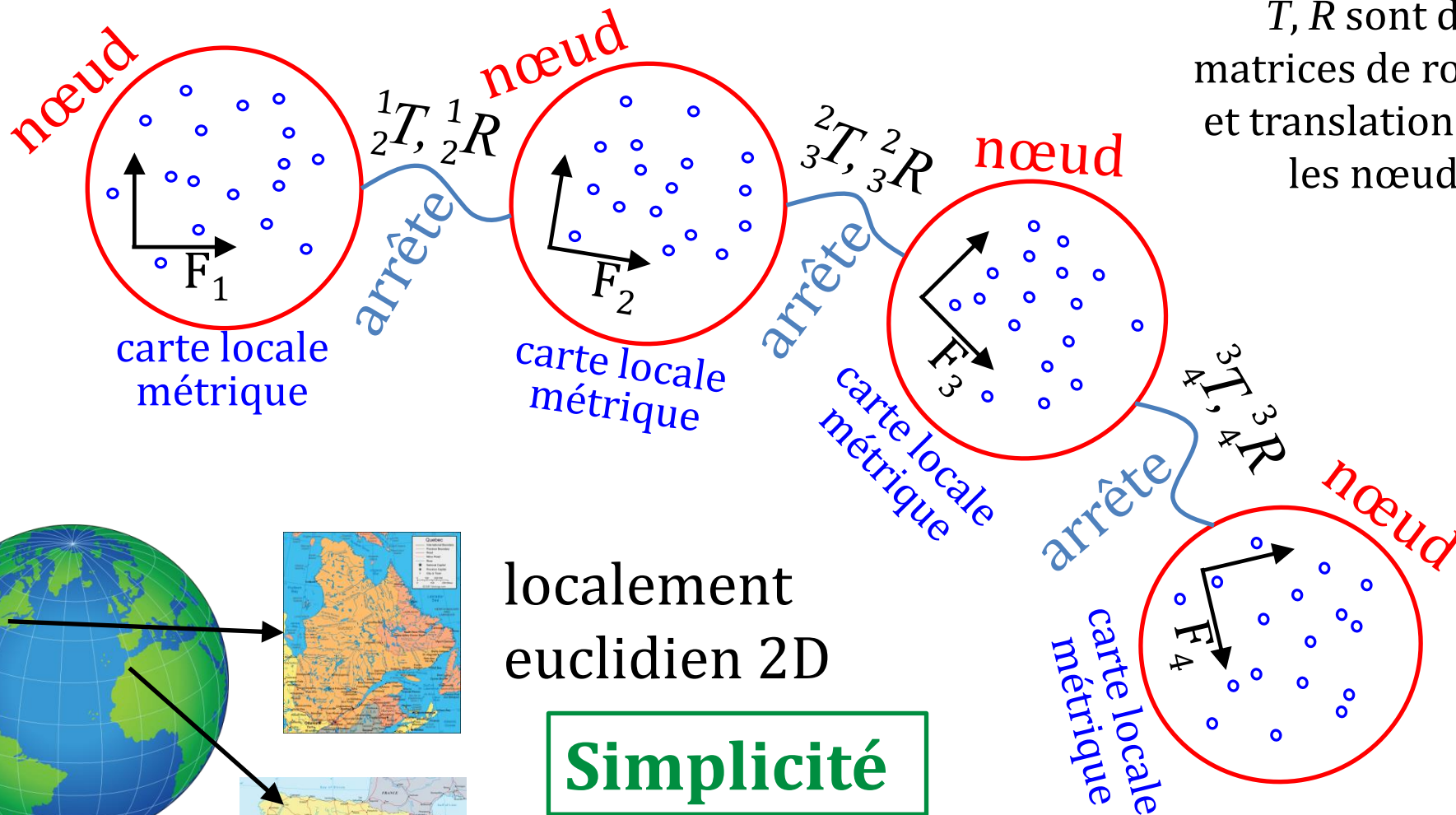
# Teach-and-Repeat

- Algorithme en deux phases
  - *Teach* : un humain conduit le robot, dépose un **rail virtuel** dans l'environnement
  - *Repeat* : le robot peut se conduire de manière autonome sur ce **rail**
- Données stockée dans une **carte topométrique**
- Fonctionne pour une multitude de capteurs



# Carte topométrique == Graphe

$T, R$  sont des matrices de rotation et translation entre les nœuds



localement  
euclidien 2D

**Simplicité**

Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy, B. Kuipers, et al., *ICRA* 2004.

# Phase 1 : *Teaching*

- Un humain expert déplace le robot dans un environnement, une seule fois
- L'ordinateur construit à la volée une **carte topométrique**
  - **carte 2D/3D repères** (repères visuels / scan LiDAR) attachée à chaque nœud
  - enregistre la **trajectoire** du robot dans cette carte locale
  - enregistre aussi les commandes au robot
  - nœuds à intervalle régulier (0.3 m – 2 m)

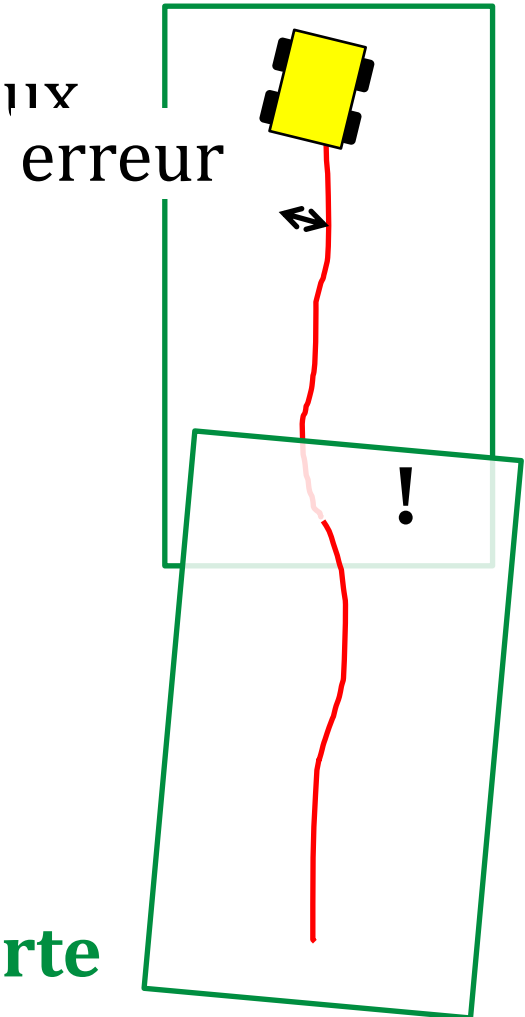




# Phase 2 : *Repeating*



- Part d'un **nœud** connu
- Envoie les commandes pré-enregistrées aux moteurs
- Estime régulièrement l'erreur sur la **trajectoire**
- Corrige les commandes aux moteurs (asservissement sur l'erreur)
- Système détecte quand le robot quitte un **nœud (carte locale)**
- Charge la nouvelle carte
- Suit la **trajectoire** dans cette **nouvelle carte**



# Méthode de localisation

---

- Plusieurs modalités possibles :

- Caméra stéréo + repères visuels

Furgale P. T. and Barfoot T. D. “Visual Teach and Repeat for Long-Range Rover Autonomy”. *Journal of Field Robotics*, 2010.

- Intensité LiDAR 3D

Dong H. J. and Barfoot T. D. “Lighting-Invariant Visual Odometry using Lidar Intensity Imagery and Pose Interpolation”, *Int’l Conf. on Field and Service Robotics (FSR)*, 2012.

- LiDAR 3D



Lidar-based Teach-and-Repeat of Mobile Robot Trajectories, C. Sprunk, G. D. Tipaldi, A. Cherubini, W. Burgard, *IROS* 2013.

# *TnR* : contournement de problèmes

---

- **Planification des déplacements du robot**
  - Identification où l'on peut rouler est difficile
  - Commandes à envoyer aux moteurs pour suivre une trajectoire
- **Carte (localisation)**
  - Gestion une seule carte globale métrique est coûteuse en calcul et en espace mémoire
  - Fusion explicite de cartes locales



# Plateforme robotique mobile

# Husky A200 de Clearpath Robotics

---

- Véhicule autonome tout-terrain
- Poids : 50 *kg*
- 4 roues motorisées par 2 moteurs électriques
  - roues d'un côté sont actionnées par un même moteur
- Rotation par *skid-steering*
- Autonomie 2-3 heures
- Charge utile : 75 *kg*
- Vitesse maximale : 1 *m/s*
- Pente maximale : 30°



# ROS : Robot Operating System

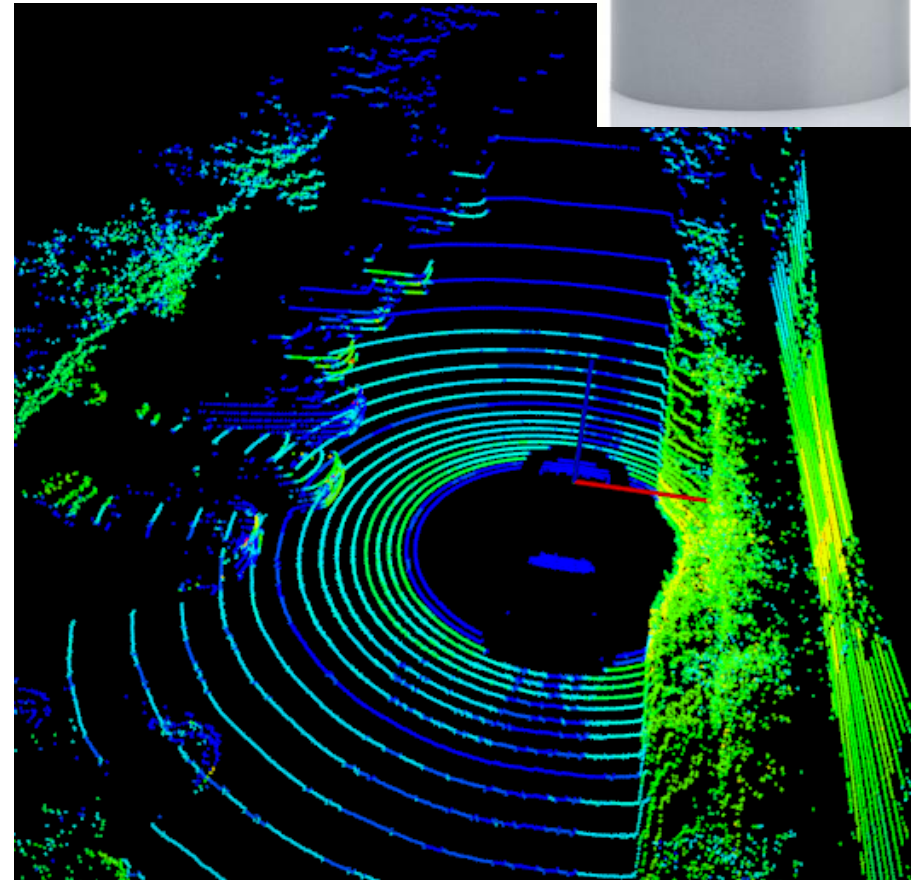
---

- Infrastructure logicielle *open-source*
- Standard *de facto* en robotique
- Offre des interfaces standards pour différents capteurs (modèles, compagnies)
- 1,000+ algorithmes implémentés (localisation, cartographie, planification, calibration, vision, etc) par des grands laboratoires de recherche
- Outils d'enregistrement/visualisation des données

# LiDAR Velodyne HDL-32E

- 32 lasers
- 360° azimuth
- +10° à -30° (surtout sol)
- 700,000 points/secondes
- portée 80-100 *m*
- précision de 2 *cm*

**Principal capteur pour  
localisation**



Prêt de l'agence spatiale  
canadienne



# Autre capteurs sur Husky

---

- Odométrie des roues via encodeurs optiques
  - 200,000 pulsation/*m*
- Centrale inertielle CH Robotics UM-6 (~300\$)
- DGPS Novatel SMART-V1G (20 *cm*)
- Wifi 54Mbps, portée 1 *km*

# Implémentation & Expérimentations

# Détails

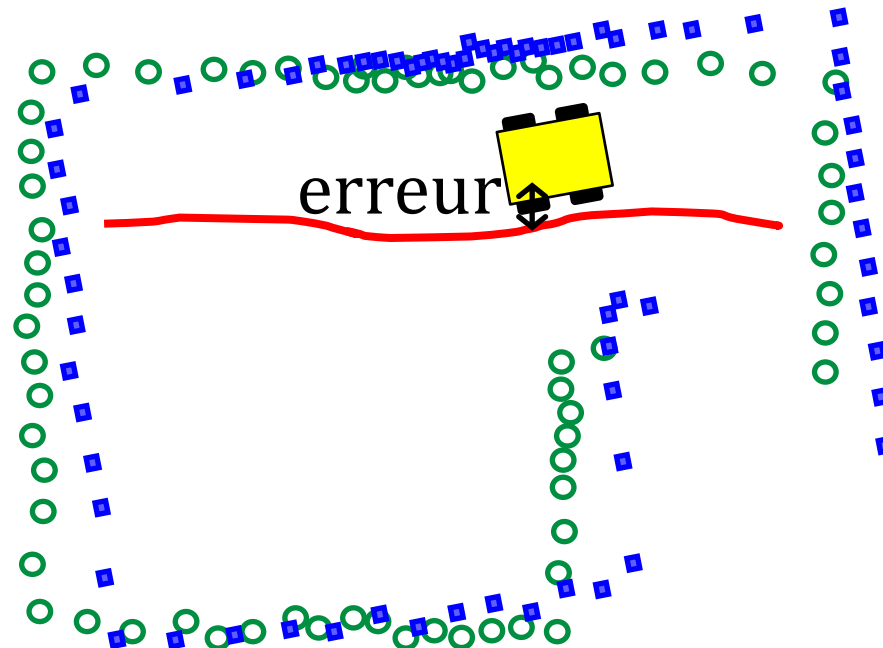
---

- Vitesse du robot :  $0.3 \text{ m/s}$
- Nœuds
  - au 10 à 20 *cm*
  - carte : 80,000 points LiDAR
  - 1 Mo données par nœud
- Localisation
  - Localisation initiale entre les scans via odométrie des roues
  - raffine avec ICP
  - corrections aux 2-4 *Hz*

# Positionnement par LiDAR : ICP

Se localise par rapport à la **carte enregistrée** lors du *Teach* via l'algorithme *Iterative Closest Point*

— trajectoire  
○ carte  
■ scan actuel



`libpointmatcher` de F. Pomerleau

# Test 1 : terrain plat, ouvert

---



Superposition vidéo Teach + vidéo Repeat



# Test 2 : forêt, terrain accidenté

---



Superposition vidéo Teach + vidéo Repeat

Erreur à la fin : 1-2 cm

# Test 3 : occlusions

---

Robustesse grâce aux nombreux points par scan LiDAR



Superposition vidéo Teach + vidéo Repeat

# Conclusion

---

- *Teach-and-Repeat* permet d'automatiser facilement les déplacements de robots mobiles en milieux non-structurés
- Nécessite une première phase de conduite manuelle (*Teach*)
- Expérimentations démontrent que l'approche est viable et robuste en forêt avec LiDAR
- **Défi** : gérer les changements court/long terme de l'environnement