# ROB-104 — Leçon 10

## Projet intégrateur : navigation autonome et collaborative

Maîtrise en robotique, intelligence artificielle et systèmes autonomes • Manuel illustré • Formation à distance

Mission : concevoir, simuler et valider deux robots mobiles autonomes partageant un environnement, avec localisation, planification, perception, coordination et réaction aux défaillances.

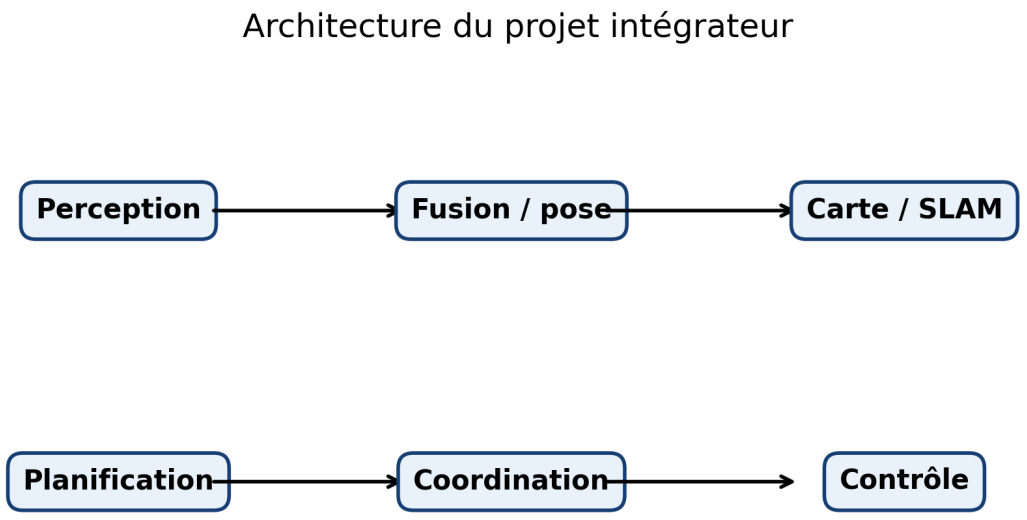



*Figure 10.1 — Architecture fonctionnelle du projet, à relier aux blocs ROS 2 et Nav2.*

## 1. Compétences et résultats attendus

- Concevoir une architecture modulaire de navigation ROS 2/Nav2.
- Créer une carte et localiser un robot avec covariance mesurable.
- Planifier des missions pour deux robots dans une zone partagée.
- Gérer piétons, obstacles imprévus et conflits de réservation.

Faculté d’informatique • Département de robotique, IA et systèmes autonomes • 2026

- Détecter les défaillances et démontrer un arrêt sûr ou un repli validé.
- Présenter des preuves expérimentales reproductibles et un film pédagogique.

## 2. Cahier des charges du campus simulé

Le terrain d'essai est un campus intérieur de 12 × 8 m avec trois zones de livraison, un passage étroit, des obstacles statiques et au moins un piéton simulé. Deux robots différentiels possèdent chacun encodeurs, IMU et LiDAR ou caméra de profondeur. Les données et performances sont simulées : aucune valeur ne doit être présentée comme résultat expérimental réel avant exécution.

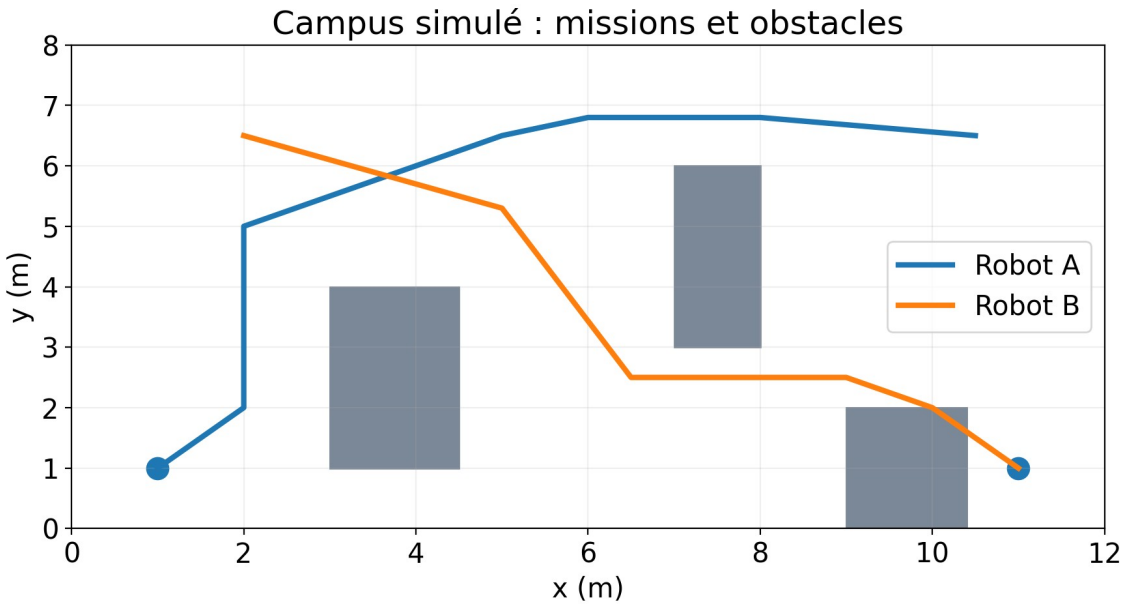



*Figure 10.2 — Exemple d'environnement à reproduire dans Gazebo; les tracés servent de référence pédagogique.*

### Hypothèses et contraintes

- Vitesse nominale initiale limitée à 0,5 m/s en simulation.
- Rayon géométrique du robot : 0,22 m, à adapter au modèle utilisé.
- Repères map, odom et base_link cohérents; timestamps synchronisés.

- Toute perte critique de perception impose une réduction de vitesse ou un arrêt suivant la preuve de sécurité.
- Le passage étroit n'admet qu'un robot à la fois, avec réservation et expiration.

## 3. Architecture logicielle et flux de données

Chaque robot possède son espace de noms, ses repères TF, son estimation d'état, sa costmap locale, son contrôleur et son arrêt sûr. Le gestionnaire de flotte attribue les tâches et réserve les ressources partagées. Les échanges ne remplacent pas la détection locale des obstacles.

La pile Nav2 réalise planification et contrôle; la localisation peut utiliser AMCL sur une carte connue, ou un système SLAM pendant la construction de carte. La fusion des encodeurs et de l'IMU réduit certaines erreurs, mais elle ne supprime pas la dérive sans observation absolue.

### Interfaces à documenter

/robot_1/odom et /robot_2/odom : poses et vitesses estimées

/robot_1/scan et /robot_2/scan : données LiDAR, selon simulation

/tf et /tf_static : transformations, avec noms de frames distincts

/robot_1/cmd_vel et /robot_2/cmd_vel : commandes contrôlées

/fleet/reservations : exemple de topic applicatif à créer

Les noms des topics de flotte sont des propositions de conception; ils ne sont pas des interfaces ROS 2 standard fournies automatiquement.

## 4. Modèle d'état et critères d'estimation

Un état minimal pour un robot plan peut s'écrire $x=[p_x,p_y,\theta,v,\omega]^T$. Le mouvement différentiel sur $\Delta t$ suit approximativement $p_x(k+1)=p_x(k)+v$

$\cos\theta\,\Delta t$, $p_y(k+1)=p_y(k)+v\sin\theta\,\Delta t$, $\theta(k+1)=\theta(k)+\omega\Delta t$. Les bruits du modèle et des capteurs déterminent la covariance de l'estimation.

La validation compare la pose estimée à une vérité de référence issue de la simulation. L'erreur de position $e=\sqrt{[(x_{est}-x_{ref})^2+(y_{est}-y_{ref})^2]}$ est calculée à chaque échantillon; on rapporte moyenne, médiane, maximum et quantiles.

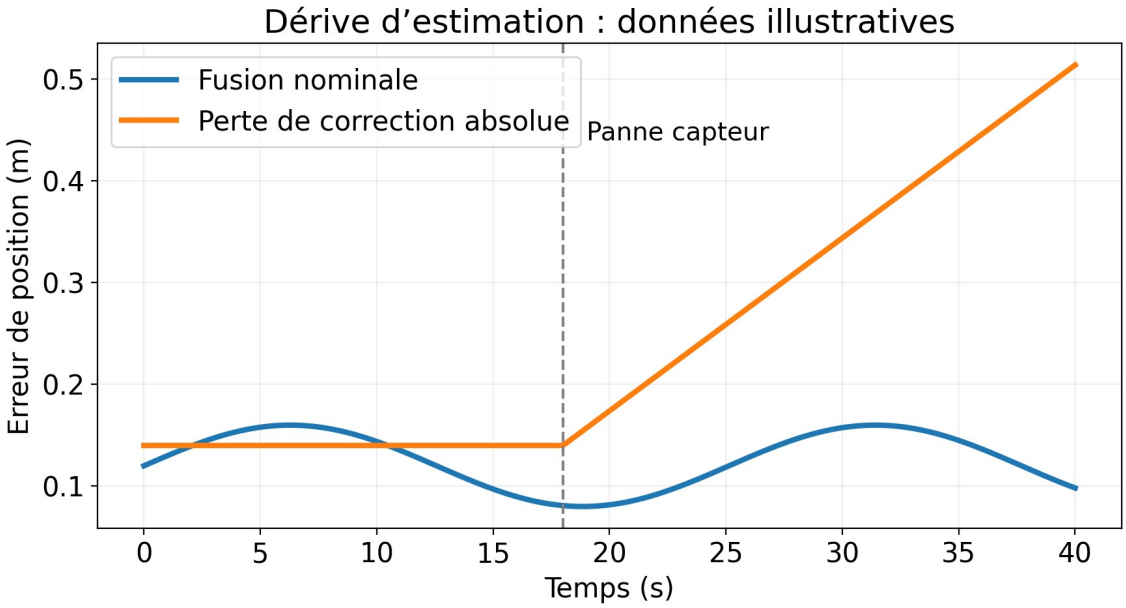



*Figure 10.3 — Effet conceptuel d'une perte de correction sur l'erreur de position; courbes illustratives, non résultats mesurés.*

## 5. Planification et coordination de deux robots

Le gestionnaire attribue les missions selon la disponibilité, la distance et la priorité. Un corridor partagé est traité comme une ressource exclusive : demande, attribution, entrée, libération. Les réservations doivent expirer si la communication est interrompue. Le robot vérifie toujours que la zone est physiquement libre avant d'entrer.

### Exemple numérique : arbitrage

Robot A demande le passage de t=12 à 17 s; B de t=14 à 20 s. Les créneaux se chevauchent sur [14,17]. Si A est prioritaire, B attend la libération confirmée, puis recalcule son itinéraire et son heure d'arrivée. Le temps d'attente minimal théorique est 3 s, hors marges de dégagement.

## 6. Gestion des obstacles dynamiques et distance d'arrêt

Pour une vitesse v, une latence $\tau$ et une décélération disponible $a>0$, la distance d'arrêt simplifiée est $d=v\tau+v^2/(2a)$. À $v=0{,}5$ m/s, $\tau=0{,}25$ s et $a=0{,}8$ m/s², $d=0{,}125+0{,}15625\approx0{,}281$ m avant toute marge. Avec 0,25 m de marge supplémentaire, la distance libre nécessaire atteint environ 0,531 m.

La vitesse doit être réduite lorsque les capteurs deviennent moins fiables, lorsque la densité d'obstacles augmente ou lorsqu'un piéton pénètre dans la zone d'anticipation. La couche de sécurité doit pouvoir neutraliser une commande non admissible.

## 7. Défaillances à injecter et modes dégradés

### Perte LiDAR

Couper le flux /scan pendant 2 s; vérifier la détection de timeout et la transition vers un état sûr.

### Biais IMU

Ajouter une dérive progressive; suivre l'innovation et la covariance.

### Perte réseau

Interrompre le gestionnaire de flotte; interdire l'entrée dans une zone réservée sans validation.

### Obstacle soudain

Faire apparaître un piéton dans le corridor; mesurer distance minimale et temps de réaction.

### Blocage mécanique

Simuler une roue immobile et vérifier la cohérence entre commande et odométrie.

## 8. Protocole de laboratoire — cinq journées

### Jour 1 — Mise en place

Installer la simulation, inspecter les nœuds ROS 2, vérifier TF, capteurs et téléopération limitée.

### Jour 2 — Cartographie et localisation

Construire ou charger la carte, calibrer les capteurs et mesurer l’erreur de pose.

### Jour 3 — Navigation nominale

Exécuter trois destinations par robot et enregistrer les trajectoires et commandes.

### Jour 4 — Coordination et obstacles

Ajouter passage partagé, piétons et négociation de priorité.

### Jour 5 — Pannes et validation

Injecter cinq fautes, mesurer la réaction et présenter le rapport final.

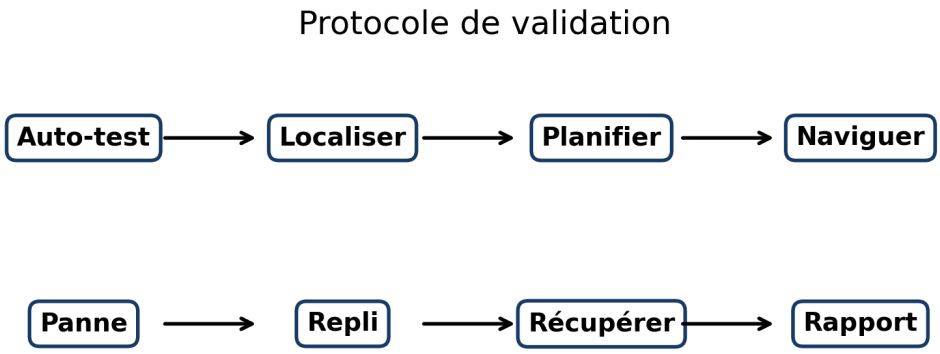



*Figure 10.4 — Cycle de validation à répéter pour chaque scénario.*

## 9. Tableau de résultats à compléter

| Essai | Durée | Erreur pose | Distance min. | Issue |
|---|---|---|---|---|
| Navigation nominale | À mesurer | À mesurer | À mesurer | À évaluer |
| Passage partagé | À mesurer | À mesurer | À mesurer | À évaluer |
| Piéton imprévu | À mesurer | À mesurer | À mesurer | À évaluer |
| Perte LiDAR | À mesurer | À mesurer | À mesurer | À évaluer |
| Perte réseau | À mesurer | À mesurer | À mesurer | À évaluer |

## 10. Barème d’évaluation du projet

Architecture et documentation des interfaces — 15 %

Cartographie et localisation — 20 %

Planification et navigation — 20 %

Coordination multi-robots — 15 %

Détection des pannes et sécurité — 20 %

## 11. Autoévaluation corrigée

**Question — Pourquoi des namespaces distincts ?**

Corrigé — Pour éviter les collisions de noms entre topics, nœuds et services des deux robots.

**Question — Pourquoi une réservation ne suffit-elle pas à prévenir une collision ?**

Corrigé — Un obstacle ou une panne peut rendre la zone occupée malgré la réservation; la perception locale reste indispensable.

**Question — Que représente la covariance ?**

Corrigé — L'incertitude associée à l'estimation d'état.

**Question — Pourquoi mesurer une distance minimale ?**

Corrigé — Pour vérifier quantitativement les marges de séparation dans les scénarios testés.

**Question — Quelle réaction après perte de perception critique ?**

Corrigé — Limiter la vitesse ou arrêter selon la preuve de sécurité disponible.

## 12. Film pédagogique : scénario dialogué

Scène 1 : le professeur présente les deux robots et le campus simulé. Scène 2 : les étudiants vérifient les capteurs et la carte. Scène 3 : les robots reçoivent leurs missions. Scène 4 : un conflit apparaît dans le passage étroit; une étudiante explique la réservation. Scène 5 : un piéton traverse, les robots ralentissent. Scène 6 : une panne LiDAR est injectée et le robot

concerné s’arrête. Scène 7 : après diagnostic, l’équipe valide la reprise et présente ses résultats.

Livrables : code source, configuration ROS 2/Nav2, carte, journaux, graphiques, rapport technique, vidéo de démonstration et réponses aux questions.