Bidirectional navigation with Nav2

Guillaume Doisy - Lead System Architect
guillaume@dexory.com
ROSCon 2023 – 19/10/2023
- Why?
- How?
  - Method A: Inversion trick
  - Method B: Full Nav2 way
Bidirectionality: why?

- Staying agile without having a circular footprint. In other words, dealing with dead ends.
- **Advantages** of rectangular over circular robots:
  - Easier to manufacture
  - Better longitudinal stability (castor wheels can be placed further away from driving wheels)
  - Better threshold crossing capability
  - For the same width, can carry a much bigger payload
- **Drawbacks**: needs to be fully bidirectional, hence
  - Needs navigation sensor symmetry
  - Navigation software is more complex...
How?
Method A: Inversion trick

- **Inversion trick on the planner and the controller**
  - **Advantages**:
    - Transparent from the point of view of the navigation software
    - Works with old architectures
    - Simple planner: no orientation computation
  - **Drawbacks**:
    - Hacky
    - Assumes symmetricity of the payload (same capabilities on both sides)
How?
Method A: Inversion trick

Example: Wyca Robotics
Elena/Astrid:
- First implemented in ROS 1 + move_base_flex + BT, then ROS 2
- Logistics warehouse inventorying application
- Symmetric payload: cameras left and right
How?
Method B: Full nav2 way

Direction agnostic everywhere:
- Planner: **SmacHybrid**
- Controller: **MPPI**
- Twist pipeline (Sm other, Muxer, Collision Monitor)
How?
Method B: Full nav2 way

Planner: **SmacHybrid**
- Support non-circular robots: can perform polygonal collision checks, i.e. not only circumscribed radius
- Plan with orientation
- Opportunistic inversion with a priori knowledge
- Loop inversion
How?
Method B: Full nav2 way

Planner: SmacHybrid

Example:
- Start: Forward
- Goal: Forward
=> Classic path
How?
Method B: Full nav2 way

Planner: **SmarHybrid**

Example:
- Start: Backward
- Goal: Backward

=> Similar path, but inverted as
Start: Forward / Goal: Forward
How?
Method B: Full nav2 way

Planner: SmacHybrid

Example:
- Start: Forward
- Goal: Backward

=> Opportunistic inversion, while it is possible and in anticipation of the final pose
How?
Method B: Full nav2 way

Planner: SmacHybrid

Example:
- Start: Backward
- Goal: Forward

=> Opportunistic inversion, while it is possible and in anticipation of the final pose. Start: Backward / Goal: Forward
How?
Method B: Full nav2 way

Planner: SmacHybrid

Example: Loop path supported!
How?
Method B: Full nav2 way

Controller: MPPI

Ability to respect the path orientation, provided an appropriate planner is used, and with the parameter PathAngleCritic.mode: 2
How?
Method B: Full nav2 way

Controller: MPPI

- Ability to respect path inversions (i.e. don’t shortcut an inversion computed by the planner) with the parameter `enforce_path_inversion`
- Inversion pose is temporary considered as the goal of the controller
How?
Method B: Full nav2 way

Bidirectional ready twist pipeline, including the velocity smoother
How?
Method B: Full nav2 way

Bidirectional ready twist pipeline, including the velocity smoother
How?
Method B: Full nav2 way
Thanks!

Guillaume Doisy - Lead System Architect
guillaume@dexory.com
ROSCon 2023 – 19/10/2023