

homer_navigation

Known Issues / Todo's

Die Pfadplanung dauert im Gegensatz zur alten Robbie-Implementierung zu lange. Je mehr freie Fläche vorhanden ist, desto länger dauert die Berechnung eines Pfades.

Introduction

Das Package "homer_navigation" mit seiner gleichnamigen Node ist für die Navigation des Roboters zuständig. Es hält stets die aktuelle Karte des Roboters vor, die z.B. vom map_manager verschickt wurde und berechnet anhand von dieser Karte einen Pfad vom Roboter zu einem gegebenen Zielpunkt. Dafür verwendet die Navigation die sogenannte Pfad-Transformation - eine Kombination aus Distanz-Transformation und Hindernistransformation -, in Verbindung mit A-Stern, um einen optimalen Pfad zu finden. Das Verhältnis zwischen Distanz- und Hindernis-Transformation kann über einen Parameter angepasst werden. Nach der Pfadberechnung ist das Package außerdem dafür zuständig den Roboter diesen Pfad zum Ziel abfahren zu lassen.

Topics

Publisher

- /robot_platform/MoveRobot (robot_platform/MoveRobot): Über dieses Topic wird der aktuelle Fahrbefehl an die Roboterplattform geschickt.
- /robot_platform/TurnRobot (robot_platform/TurnRobot): Über dieses Topic wird der aktuelle Drehbefehl an die Roboterplattform geschickt.
- /robot_platform/StopRobot (robot_platform/StopRobot): Über dieses Topic wird der Roboter am Ziel beziehungsweise vor einem Hindernis gestoppt.
- /homer_navigation/target_reached (std_msgs/Empty): Wenn der Roboter sein Ziel erreicht hat, wird eine Message über dieses Topic verschickt.
- /homer_navigation/target_unreachable (map_messages/TargetUnreachable): Über dieses Topic wird mitgeteilt, wenn der Roboter sein Ziel nicht erreichen kann und die Navigation abgebrochen wurde. Ein Statuscode wird mitgeliefert (siehe Package map_messages).
- /homer_navigation/path (nav_msgs/Path): Hierüber wird der aktuelle Pfad zur Visualisierung an die GUI geschickt.

Subscriber

- `/map (nav_msgs/OccupancyGrid)`: Die jeweils aktuelle Map wird empfangen, um sie für die Pfadplanung und Hindernisvermeidung während der Navigation zu verwenden.
- `/pose (geometry_msgs/PoseStamped)`: Die aktuelle Pose des Roboters wird als Startpunkt der Pfadplanung verwendet.
- `/scan (nav_msgs/LaserScan)`: Der aktuelle LaserScan wird zur Hindernisvermeidung verwendet.
- `/robot_platform/task_finished (std_msgs/Empty)`: Wird vom Roboter nach erfolgter Ausführung einer Bewegung verschickt. Die Navigation verwendet dies am Ende der Navigation, um nach der letzten Drehung auf dem Zielpunkt die TargetReached-Nachricht zu versenden.
- `/homer_navigation/start_navigation (map_messages/StartNavigation)`: Startet die Pfadplanung und anschließend die Navigation zur mitgelieferten Zielpose
- `/homer_navigation/stop_navigation (map_messages/StopNavigation)`: Stoppt die aktuelle Navigation.
- `/homer_navigation/navigate_to_POI (map_messages/NavigateToPOI)`: Startet die Pfadplanung und anschließend die Navigation zum POI mit angegebenen Namen. Der entsprechende POI wird anschließend vom `map_manager` erfragt.
- `/homer_navigation/unknown_threshold (std_msgs/Int8)`: Über dieses Topic kann der Schwellwert verändert werden, ab dem ein Zellwahrscheinlichkeit als belegt und damit unbefahrbar markiert wird. Standard ist 50 (%).

Launch Files

- `homer_navigation.launch`: Startet die Navigation und lädt alle Navigations-Parameter in den Parameterserver.

Parameter

`homer_navigation`

- `/homer_navigation/safe_path_weight`: Gewicht, um Bevorzugung sicherer (d.h. größerer Abstand zu Hindernissen) Pfade im Gegensatz zu kürzeren Pfaden einzustellen. 0 bedeutet kürzester Pfad, aber keine Sicherheit. Je höher, desto sicherer.
- `/homer_navigation/allowed_obstacle_distance/min`: Mindestdistanz zum nächsten Hindernis, die der Roboter bei der Navigation besitzen muss.

- `/homer_navigation/allowed_obstacle_distance/max`: Maximaldistanz zum nächsten Hindernis, die der Roboter bei der Navigation besitzen muss (wichtig für die Lokalisierung, abhängig von der Reichweite der Abstandssensoren).
- `/homer_navigation/safe_obstacle_distance/min`: Minstdistanz des Roboters zum nächsten Hindernis für eine sichere Navigation. Gewichtung wird mit `safe_path_weight` eingestellt.
- `/homer_navigation/safe_obstacle_distance/max`: Maximaldistanz des Roboters zum nächsten Hindernis.
- `/homer_navigation/frontier_safeness_factor`: Faktor der minimal erlaubten Distanz zum nächsten Hindernis, den eine Grenzzelle zu unbekanntem Gebiet mindestens haben muss, um als sicher und damit befahrbar zu gelten.
- `/homer_navigation/collision_distance`: Wenn die Distanz zum nächsten Hindernis in Metern (gemessen ab der Front des Roboters) kleiner wird als der angegebene Parameter, wird der aktuelle Fahrbefehl gestoppt und die Pfadplanung neugestartet.
- `/homer_navigation/collision_distance_near_target`: Nähe in Metern zum Ziel, ab der der Roboter bei einer Kollision nicht mehr zurückfährt und neuplant, sondern stoppt und eine TargetReached-Nachricht versendet.
- `/homer_navigation/backward_distance`: Strecke in Metern, die der Roboter bei einer bevorstehenden Kollision zurückfährt, bevor er einen neuen Pfad plant.
- `/homer_navigation/check_path`: Wenn auf "true" gesetzt, wird der aktuell geplante Pfad auf neu hinzukommende Hindernisse geprüft und gegebenenfalls neugeplant.
- `/homer_navigation/check_path_max_errors`: Wenn Pfade gecheckt werden, wird umgeplant, sobald sich "check_path_max_errors"-mal hintereinander ein Hindernis in der ankommenden Karte im Pfad befindet.
- `/homer_navigation/check_path_max_distance`: Der Pfad wird nur in der Nähe des Roboters bis zur im Parameter angegebenen Distanz auf neu hinzukommende Hindernisse überprüft.
- `/homer_navigation/turn_threshold_angle`: Wenn der Winkel zwischen der Orientierung des Roboters und der Strecke zum nächsten Wegpunkt unter dem angegebenen Wert liegt, soll geradeaus gefahren werden, ansonsten soll sich der Roboter um diesen Winkel in Richtung des Wegpunkts drehen.
- `/homer_navigation/waypoint_sampling_threshold`: Parameter, um Anzahl der Wegpunkte im Pfad zu minimieren. 0 bedeutet keine Reduzierung. Je höher der Wert, desto mehr Wegpunkte werden gelöscht.
- `/homer_navigation/speed_control/last_speedfactor_count`: Größe des Vektors, der die zuletzt berechneten Geschwindigkeitsfaktoren berechnet. Wird benutzt, um aus dem Mittelwert zusammen mit der aktuellen Distanz zum nächsten Hindernis die aktuelle Geschwindigkeit zu berechnen.

- `/homer_navigation/speed_control/min_move_speedfactor`: Minimaler Faktor der maximal möglichen Geschwindigkeit, mit der der Roboter jemals fahren darf.
- `/homer_navigation/speed_control/min_turn_speedfactor_moving`: Minimaler Faktor der maximal möglichen Drehgeschwindigkeit, mit der der Roboter sich während einer Geradeausfahrt drehen darf (aktuell mit dem `cu2wd` nicht möglich).
- `/homer_navigation/speed_control/min_turn_speedfactor_standing`: Minimaler Faktor der maximal möglichen Drehgeschwindigkeit, mit der sich der Roboter im Stand drehen darf.
- `/homer_navigation/speed_control/inner_danger_zone/x_min`: Wenn sich in der mit den nächsten vier Parametern beschriebenen Boundingbox um den Roboter ein Hindernis befindet, darf sich der Roboter nur mit der vorher eingestellten nachfolgend eingestellten Geschwindigkeit bewegen.
- `/homer_navigation/speed_control/inner_danger_zone/x_max`
- `/homer_navigation/speed_control/inner_danger_zone/y_min`
- `/homer_navigation/speed_control/inner_danger_zone/y_max`
- `/homer_navigation/speed_control/inner_danger_zone/speed_factor`: Faktor der maximalen Geschwindigkeit, mit der sich der Roboter in seiner inneren Gefahrenzone bewegen darf.
- `/homer_navigation/speed_control/outer_danger_zone/x_min`: Eine zweite Boundingbox, die die innere umschließen muss.
- `/homer_navigation/speed_control/outer_danger_zone/x_max`
- `/homer_navigation/speed_control/outer_danger_zone/y_min`
- `/homer_navigation/speed_control/outer_danger_zone/y_max`
- `/homer_navigation/speed_control/outer_danger_zone/speed_factor`: Faktor der maximalen Geschwindigkeit, mit der sich der Roboter in seiner äußeren Gefahrenzone bewegen darf.
- `/homer_navigation/max_rot_vel`: Maximal mögliche Rotationsgeschwindigkeit des Roboters in Radians/Sekunde.
- `/homer_navigation/max_trans_vel`: Maximal mögliche Geschwindigkeit des Roboters in Metern/Sekunde.