

homer__mapping

Introduction

Das Package `homer__mapping` besteht aus einer gleichnamigen Node. Diese ist für die Lokalisierung und Kartierung des Roboters mit Hilfe der Odometrie des Roboters und eines Laserscanners zuständig. Das SLAM-Problem wird durch den Partikelfilter-Algorithmus gelöst. Die Node erwartet kontinuierlich Odometrie-Werte und Laserdaten und verschickt in konstanten Abständen korrigierte Poseschätzungen über das Topic `/pose` und tf-Transformation `/map` -> `/base_link`. Zudem kann der Roboter sich auf einer vorher geladenen Karte lokalisieren, sowie eine aktuell erstellte Karte abgespeichert werden. Es besteht die Option, die Kartierung ein- oder auszuschalten. Beim Laden einer Karte wird die Kartierung automatisch ausgeschaltet.

Topics

Publisher

- `/pose` (`geometry_msgs/PoseStamped`): Die aktuell ermittelte Pose relativ zu Karte (im Frame `/map`) des Roboters aus dem Partikelfilter.
- `/homer_mapping/slam_map` (`nav_msgs/OccupancyGrid`): Das aktuelle Karte des Roboters.

Subscriber

- `/odom` (`nav_msgs/Odometry`): Die aktuellen Odometrie-Werte vom Roboter. Diese werden für die Partikelfilter benötigt. `/scan` (`sensor_msgs/LaserScan`): Die aktuelle Lasermessung, die vom Partikelfilter benötigt wird.
- `/homer_mapping/userdef_pose` (`geometry_msgs/Pose`): Mit diesem Topic kann die die aktuell vom Partikelfilter berechnete Pose auf eine benutzerdefinierte gesetzt werden. Der Partikelfilter arbeitet nun mit dieser weiter.
- `/homer_mapping/do_mapping` (`map_messages/DoMapping`): Mit diesem Topic kann die Kartierung ein- oder ausgeschaltet werden.
- `/map_manager/reset_maps` (`std_msgs/Empty`): Hiermit kann die aktuelle Karte zurückgesetzt werden.
- `/map_manager/loaded_map` (`nav_msgs/OccupancyGrid`): Mit diesem Topic kann die aktuelle Karte durch eine andere (geladene) Karte ausgetauscht werden.
- `/map_manager/mask_slam` (`nav_msgs/OccupancyGrid`): Im Occupancy-Grid dieses Topics stehen Informationen, welche Teile der aktuellen Karte durch andere Werte (frei oder belegt) ersetzt werden sollen.

Launch Files

- `homer_mapping.launch`: Dieses Launchfile lädt die Parameterdatei `homer_mapping.yaml` und startet die Node `homer_mapping` sowie die Node `map_manager` im gleichnamigen Package.
- `homer_mapping_rviz.launch`: Dieses Launchfile hat die gleiche Funktionalität wie das obige, wobei es zusätzlich `rviz` startet.

Parameter

`homer_mapping.yaml`

- `/homer_mapping/size`: Size beschreibt die Größe einer Seite der Karte in Metern. Die Karte ist quadratisch
- `/homer_mapping/resolution`: Resolution ist die Länge einer (quadratischen) Zelle der Karte in Metern
- `/homer_mapping/backside_checking`: Wenn auf "true" gesetzt, wird verhindert, dass Vorder- und Rückseite einer dickeren Wand auf die gematcht werden.
- `/homer_mapping/obstacle_borders`: Wenn auf "true" gesetzt, wird um eingetragene Hindernisse ein kleiner Rand unbekanntes Gebiet gelassen.
- `/homer_mapping/measure_sampling_step`: Minimale Distanz in Metern, die zwischen zwei aufeinanderfolgenden Messpunkten im Laserscan vorhanden sein muss, um sie für die Poseberechnung zu verwenden
- `/homer_mapping/laser_scanner/free_reading_distance`: Minimale Distanz in Metern, die als hindernissfrei angenommen wird, wenn der aktuelle Messpunkt fehlerhaft ist
- `/particlefilter/error_values/rotation_error_rotating`: Rotationsfehler in Prozent, der beim Rotieren des Roboters auftritt
- `/particlefilter/error_values/rotation_error_translating`: Rotationsfehler in Grad, der beim Fahren von einem Meter auftritt
- `/particlefilter/error_values/translation_error_translating`: Translationsfehler in Prozent, der beim Geradeausfahren auftritt
- `/particlefilter/error_values/translation_error_rotating`: Translationsfehler in Metern, der beim Rotieren von einem Grad auftritt
- `/particlefilter/error_values/move_jitter_while_turning`: Streuung der neu berechneten Pose in Meter pro Grad Drehung
- `/particlefilter/hyper_slamfilter/particlefilter_num`: Anzahl der Partikelfilter im Hyperpartikelfilter (ist standardmäßig auf 1 gesetzt)
- `/particlefilter/particle_num`: Anzahl der Partikel in jedem Partikelfilter
- `/particlefilter/max_rotation_per_second`: Maximale Rotation in Radiant pro Sekunde, die der Roboter sich drehen darf, ohne dass das Mapping ausgesetzt wird

- `/particlefilter/wait_time`: Minimale Zeit, die zwischen zwei Mapping-Schritten verstrichen sein muss
- `/particlefilter/update_min_move_angle`: Minimale Rotation in Grad, die durchgeführt werden muss, damit ein Partikelfilterschritt ausgeführt wird...
- `/particlefilter/update_min_move_dist`: ...oder minimale Distanz in Metern, die der Roboter geradeaus fährt...
- `/particlefilter/max_update_interval`: ...oder minimale Wartezeit, in der der Roboter still steht.
- `/selflocalization/scatter_var_xy`: Streuung der Partikel in x/y-Richtung in Metern beim Setzen einer benutzerdefinierten Pose
- `/selflocalization/scatter_var_theta`: Streuung der Ausrichtung der Partikel in Radiant beim Setzen einer benutzerdefinierten Pose