



SAPIENZA
UNIVERSITÀ DI ROMA

Sviluppo di un sistema di Collision Avoidance tramite il metodo dell'Artificial Potential Field

Facoltà di Ingegneria dell'Informazione, Informatica e Statistica
Corso di Laurea in Ingegneria Informatica e Automatica

Candidato

Davide Albano

Matricola 1708530

Relatore

Prof. Giorgio Grisetti

Anno Accademico 2020/2021

Tesi discussa il
di fronte a una commissione esaminatrice composta da:
(presidente)

Sviluppo di un sistema di Collision Avoidance tramite il metodo dell'Artificial Potential Field

Tesi di Laurea. Sapienza – Università di Roma

© 2021 Davide Albano. Tutti i diritti riservati

Questa tesi è stata composta con \LaTeX e la classe Sapthesis.

Versione:

Email dell'autore: albano.1708530@studenti.uniroma1.it

Sommario

L'obiettivo di questo lavoro è l'implementazione di un sistema che permetta a un robot di muoversi all'interno di un ambiente evitando eventuali ostacoli. Il robot preso in considerazione non conosce l'ambiente in cui dovrà muoversi e può quindi basarsi soltanto sulle rilevazioni fatte tramite il laser scanner.

Per l'implementazione è stato usato ROS (Robot Operating System) un framework open source usato per gestire le operazioni e le comunicazioni dei vari nodi di un robot. Per la simulazione è stato usato il pacchetto `stage_ros`, mentre per la visualizzazione delle informazioni sulla traiettoria calcolate dal robot è stato usato il tool RVIZ.

Data l'assenza di informazioni iniziali riguardo alla mappa è stato necessario scegliere un algoritmo di local motion planning. In particolare è stato scelto il metodo dell'Artificial Potential Field, che è spesso usato per la sua semplicità sia dal punto di vista dello sviluppo sia dal punto di vista computazionale.

Indice

1	Introduzione	7
1.1	Formulazione problema	7
2	ROS	9
2.1	Caratteristiche principali	9

Capitolo 1

Introduzione

1.1 Formulazione problema

Lo sviluppo di robot mobili a navigazione autonoma si suddivide in due parti: global path planning e local motion control. Il global path planning utilizza le informazioni che si hanno sulla mappa per trovare il percorso più corto per andare dal punto di partenza all'obiettivo.

Tuttavia può capitare che le informazioni sull'ambiente in cui il robot dovrà muoversi siano scarse, non aggiornate o mancanti. Questo porta alla necessita di sviluppare algoritmi di local motion control, che permettono al robot di calcolare in tempo reale una traiettoria in grado di evitare gli ostacoli incontrati durante la navigazione.

Capitolo 2

ROS

2.1 Caratteristiche principali

ROS (Robot Operating System) è un framework open source utilizzato per lo sviluppo di applicazioni per la robotica. Mette infatti a disposizione strumenti e librerie utili per aiutare gli sviluppatori software nella realizzazione di applicazioni robotiche a partire dalla scrittura fino all'esecuzione e al debugging del codice.

ROS presenta inoltre alcune caratteristiche di un sistema operativo (gestione di processi, di pacchetti e delle loro dipendenze e astrazione di dispositivi hardware a basso livello) e di un middleware perché permette la comunicazione tra processi/macchine diverse.

Infine costituisce un'architettura distribuita in cui è possibile gestire in maniera asincrona un insieme di moduli software che possono essere scritti in vari linguaggi tra cui C++, Python e Lisp.