

1. Implementasi dan Simulasi PRM dengan Visualisasi Matplotlib From Scratch

Langkah-langkah yang perlu dilakukan:

- Implementasi PRM dalam Python:
 - Buat file Python prm.py untuk mengimplementasikan algoritma PRM.
 - Algoritma PRM ini harus melakukan:
 - Generasi Node Acak: Tentukan jumlah node acak yang akan dihasilkan. Node ini mewakili titik dalam ruang konfigurasi yang harus dihubungkan.
 - Pembentukan Graf: Hubungkan node-node dengan membuat edge antar node yang terdekat sesuai dengan radius yang ditentukan. Hal ini bisa dilakukan dengan menghitung jarak antar node dan membuat edge pada jarak tertentu.
 - Pencarian Jalur Terpendek: Gunakan algoritma pencarian jalur terpendek (misalnya, Dijkstra atau A*) untuk menemukan rute dari titik awal ke titik tujuan di dalam graf yang telah terbentuk.
- Visualisasi dengan Matplotlib:
 - Gunakan Matplotlib untuk menggambarkan node-node dan edge yang terbentuk.
 - Visualisasikan hasilnya dengan memberikan warna atau tanda pada jalur terpendek agar mudah terlihat.
- Konfigurasi Parameter:
 - Buat file konfigurasi params.yaml untuk menyimpan pengaturan seperti jumlah node, radius koneksi, dan parameter lain yang dapat membantu menjalankan simulasi.
- Penggunaan ROS dan Terminal Ubuntu:
 - Buat file prm.launch untuk menjalankan skrip PRM di ROS.
 - Pastikan ROS sudah terpasang di Ubuntu, dan gunakan terminal untuk menjalankan file launch tersebut.

- Contoh Hasil Simulasi:

- Setelah menjalankan program, kalian dapat melihat contoh hasil simulasi sesuai dengan yang ditunjukkan di referensi tugas.

2. Implementasi dan Simulasi PRM dengan Visualisasi Rviz

- Langkah-langkah yang perlu dilakukan:

- Ikuti Panduan dari Repository GitHub:
- Buka repository GitHub yang sudah disediakan dalam tugas untuk mendapatkan panduan implementasi.
- Clone atau unduh repository tersebut untuk menjalankan program yang sudah tersedia.

- Jalankan Algoritma PRM di ROS:

- Buka terminal di Ubuntu dan jalankan simulasi sesuai dengan instruksi di repository. Pastikan kalian menjalankan PRM melalui ROS agar Rviz dapat digunakan sebagai platform visualisasi.

- Visualisasi di Rviz:

- Gunakan Rviz untuk memvisualisasikan roadmap dan jalur terpendek.
- Perhatikan apakah node, edge, dan jalur terpendek terlihat dengan jelas di Rviz. Pastikan konfigurasi di Rviz sesuai untuk mendapatkan tampilan yang diinginkan.

- Verifikasi Hasil:

- Pastikan simulasi berjalan dengan lancar.
- Cek kembali hasil visualisasi di Rviz dan bandingkan dengan contoh di repository GitHub untuk memastikan hasilnya sesuai.

3. Implementasi Algoritma Rapidly-Expanding Random Trees (RRT) di ROS dengan Visualisasi Rviz

- Ikuti panduan dari repository GitHub yang disediakan untuk mengimplementasikan dan mensimulasikan algoritma RRT.
 - Jalankan algoritma RRT melalui terminal Ubuntu menggunakan ROS, dan lakukan visualisasi hasil jalur di Rviz.
 - Pastikan jalur yang terbentuk dan perluasan node dari RRT terlihat jelas di Rviz.
 - Cek contoh hasil visualisasi di repository untuk memastikan visualisasi sudah sesuai.
- Hasil Analisis Simulasi

4. Hasil analisi simulasi

- Lakukan analisis untuk semua tugas yang telah diselesaikan, meliputi:
 - Implementasi dan Simulasi Probabilistic Roadmap (PRM) dengan Visualisasi Matplotlib From Scratch.
 - Implementasi dan Simulasi Probabilistic Roadmap (PRM) dengan Visualisasi Rviz.
 - Implementasi Algoritma Rapidly-Expanding Random Trees (RRT) di ROS dengan Visualisasi Rviz.
- Dalam analisis, kamu bisa membandingkan efektivitas algoritma, waktu komputasi, keakuratan jalur terpendek, serta kemudahan visualisasi di masing-masing platform.