

ANALISIS TUGAS ROBOTIKA WEEK 6

1) Implementasi dan Simulasi Probabilistic Roadmap (PRM) dengan Visualisasi Matplotlib From Scratch.

Pada tugas pertama, saya fokus pada implementasi algoritma Probabilistic Roadmap (PRM) untuk perencanaan jalur. Tujuan utama algoritma ini adalah untuk menemukan jalur dari titik awal ke titik tujuan dengan menggunakan peta probabilistik yang terdiri dari node-node acak yang terhubung satu sama lain. Saya menulis kode Python untuk melakukan beberapa langkah penting di bagian ini. Pertama, saya mendefinisikan posisi titik awal dan tujuan dalam file "environment.txt" dan "environment 2.txt". Saya juga mendefinisikan beberapa hambatan, seperti persegi panjang yang menghalangi jalur. Kemudian algoritma membuat node acak di area bebas hambatan dan menghubungkan node berdasarkan seberapa dekat mereka berada.

Saya menggunakan Matplotlib untuk visualisasi,. Dengan Matplotlib, saya dapat menggambarkan peta lingkungan, menunjukkan posisi titik awal, tujuan, dan jalur yang dibentuk antara node-node acak. Setiap node yang dihasilkan digambarkan sebagai titik, dan jalur yang menghubungkan node-node tersebut digambarkan sebagai garis. Visualisasi ini sangat membantu untuk memahami bagaimana algoritma PRM membangun peta dan mencari jalur terpendek yang menghindari hambatan. Selama pengembangan, saya juga melakukan eksperimen dengan parameter-parameter seperti jumlah sampel (node) yang diambil. Jumlah sampel yang lebih banyak biasanya menghasilkan peta yang lebih baik, namun membutuhkan lebih banyak waktu komputasi.

2) Implementasi dan Simulasi PRM dengan Visualisasi Rviz.

Pada tugas kedua, saya mengadaptasi algoritma PRM untuk digunakan dalam ROS (Robot Operating System) dan menambahkan visualisasi di RViz. ROS memberikan platform yang kuat untuk mengembangkan aplikasi robotik, dan RViz adalah alat visualisasi yang memungkinkan saya untuk melihat hasil simulasi secara langsung. Di sini, saya menjalankan node PRM di ROS, yang mengimplementasikan algoritma yang sama dengan tugas pertama, namun dengan menggunakan arsitektur ROS.

Dengan menggunakan ROS dan RViz, saya dapat memvisualisasikan peta yang dibangun oleh PRM secara dinamis. RViz memungkinkan saya untuk melihat jalur yang ditemukan oleh algoritma secara real-time, termasuk menunjukkan posisi titik awal, tujuan, dan jalur yang menghubungkan node-node acak. Saya juga dapat menyesuaikan parameter seperti jumlah node yang disampling dan tetangga terdekat yang dipertahankan dalam peta. Hal ini memberikan fleksibilitas lebih untuk eksperimen dan melihat bagaimana perubahan parameter mempengaruhi hasil perencanaan jalur. Misalnya, jika jumlah node yang diambil terlalu sedikit, peta yang dibangun akan lebih kasar, dan jalur yang ditemukan mungkin tidak optimal. Sebaliknya, jika jumlah node terlalu banyak, waktu komputasi bisa menjadi lebih lama.

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

Pada tugas ketiga, saya beralih ke algoritma Rapidly-Expanding Random Trees (RRT), yang juga diimplementasikan dalam ROS dan divisualisasikan dengan RViz. RRT adalah algoritma yang berbeda dari PRM karena menggunakan pendekatan pohon yang tumbuh secara acak dari titik awal menuju titik tujuan. Dalam implementasi ini, pohon tumbuh dengan memperluas cabang-cabangnya menuju arah yang lebih jauh, dan setiap cabang yang baru dibentuk hanya akan diterima jika tidak bertabrakan dengan hambatan.

Saya menggunakan ROS untuk menjalankan algoritma RRT, dan RViz untuk memvisualisasikan proses perencanaan jalur. Di RViz, saya bisa melihat pohon yang tumbuh secara progresif, dengan setiap iterasi memperlihatkan cabang pohon yang semakin mendekati titik tujuan. Salah satu tantangan dalam tugas ini adalah mengatur parameter RRT, seperti jumlah iterasi yang dibutuhkan dan bagaimana pohon berkembang. Saya dapat mengatur parameter ini untuk mempengaruhi seberapa cepat pohon tumbuh dan seberapa tepat jalur yang dihasilkan. Misalnya, lebih banyak iterasi dapat menghasilkan jalur yang lebih baik, tetapi membutuhkan lebih banyak waktu komputasi. RViz memberikan tampilan visual yang jelas, memungkinkan saya untuk melihat secara langsung apakah pohon berhasil menemukan jalur yang optimal atau perlu penyesuaian lebih lanjut.

➤ Kesimpulan

Saya memperoleh pemahaman tentang penerapan algoritma perencanaan jalur dalam robotik dari ketiga tugas yang saya selesaikan. Pada tugas pertama, implementasi PRM dengan Python dan visualisasi Matplotlib menunjukkan cara algoritma membangun peta probabilistik untuk mencari jalur terpendek, dengan eksperimen parameter jumlah node. Pada tugas kedua, adaptasi PRM dalam ROS dengan visualisasi RViz memungkinkan perencanaan jalur secara dinamis dan real-time serta fleksibilitas dalam menyesuaikan parameter. Pada tugas ketiga, implementasi RRT dalam ROS dan visualisasi RViz menunjukkan cara algoritma membangun peta probabilistik untuk mencari jalur terpendek, dengan Secara keseluruhan, ketiga algoritma ini menawarkan pengetahuan yang signifikan tentang perencanaan jalur dalam lingkungan terhambat, dengan visualisasi sebagai alat bantu yang sangat bermanfaat.