

Hendri Maulana Azwar

1103210202

TK-45-G09

## Analisis

### 1. Dijkstra Algorithm

Merupakan algoritma untuk mencari jalur terpendek dari satu titik ke titik lain dalam graf tanpa mempertimbangkan arah tujuan sampai titik akhir

- Kelebihan : Lebih optimal dan serbaguna
- Kekurangan : Tidak efisien untuk graf yang besar

### 2. A\* Algorithm

Merupakan algoritma untuk memperkirakan jarak dari node saat ini ke tujuan, sehingga algoritma dapat mengarahkan pencarian ke arah tujuan dengan lebih cepat.

- Kelebihan : Lebih efisien dan memiliki kompleksitas yang lebih baik
- Kekurangan : Bergantung pada heuristic

### 3. Cell Decomposition

Merupakan algoritma yang memecah ruang atau area menjadi “cell-cell” kecil yang lebih mudah dikelola. Algoritma kemudian akan mencari jalur di antara cell-cell yang bebas dari rintangan untuk menemukan jalur aman dari titik awal ke titik tujuan

- Kelebihan : Cocok untuk area yang kompleks dan lebih terstruktur dalam ruang kontinu
- Kekurangan : Batas akurasi pada resolusi cell dan tidak selalu optimal

### 4. ROS Motion Planning

Setiap algoritma memiliki trade-off yang khas antara kecepatan dan optimalitas jalur :

- GBFS : cepat dalam menemukan rute tetapi tidak selalu optimal, sehingga lebih cocok untuk kasus yang memerlukan respons cepat.
- Dijkstra : jalur yang dimiliki optimal tetapi membutuhkan waktu komputasi yang lama, lebih cocok untuk simulasi di mana jalur terpendek mutlak diperlukan.
- A\* : keseimbangan yang ideal, mengombinasikan kecepatan dan optimalitas, sehingga sangat cocok untuk kebanyakan aplikasi perencanaan jalur di ROS.

Trajectory optimization sangat membantu dalam menghasilkan jalur yang halus dan realistis. Perencanaan jalur yang dipadukan dengan optimisasi lintasan adalah kombinasi yang efektif untuk memastikan jalur yang aman, cepat, dan hemat energi