

Nama : Irman Prayista

NIM : 1103210094

Kelas : TK-45-02

## Hasil Analisis Simulasi

### 1. Simulasi *Introduction to Localization and Filtering Theory* Menggunakan Numpy dan matplotlib

Terdapat 5 simulasi yang telah dilakukan, untuk simulasi pertama adalah implementasi *Kalman Filter* untuk estimasi posisi robot. Program ini digunakan untuk estimasi posisi robot dengan menggunakan observasi yang terkontaminasi *noise*. Terdapat 2 fungsi, yaitu prediksi yang berfungsi untuk memprediksi posisi robot berdasarkan posisi sebelumnya dan ketidakpastian proses dan *update* yang berfungsi untuk memperbarui posisi estimasi dengan informasi baru dari observasi yang terkontaminasi *noise* menggunakan *Kalman Gain*.

Terdapat juga simulasi filter partikel untuk estimasi posisi robot. Program ini digunakan untuk estimasi posisi robot menggunakan observasi yang terkontaminasi *noise*. Proses ini dimulai dengan inisialisasi partikel acak di seluruh ruang posisi dan memberikan bobot yang sama untuk setiap partikel. Setiap iterasi melibatkan 2 langkah, yaitu prediksi untuk menambahkan *noise* pada posisi partikel dan *update* untuk perbarui bobot partikel berdasarkan jarak antara partikel dan observasi. Setelah *update* akan dilakukan *resampling* untuk memilih partikel yang lebih sesuai dengan observasi dan estimasi posisi dihitung sebagai rata – rata dari posisi partikel.

Terdapat simulasi lainnya seperti lokalisasi dengan sensor IMU dan lidar. Program ini digunakan untuk mengimplementasikan fusi antara sensor IMU dan lidar untuk lokalisasi robot. Data IMU berasal dari penjumlahan kumulatif *noise* dan data lidar berasal dari *linieritas* dengan penambahan *noise*. Fusi data dilakukan dengan memberikan bobot lebih pada data lidar (70%) dibandingkan dengan data IMU (30%) karena lidar lebih akurat dalam pengukuran posisi.

Ada juga simulasi lainnya seperti *Extended Kalman Filter* (EKF) untuk navigasi. Program ini digunakan memperkirakan posisi robot. *Extended Kalman Filter* (EKF) digunakan untuk model non-linier yang memerlukan *update* prediksi dan observasi secara *iteratif*. Filter ini memperbarui posisi dan kecepatan robot berdasarkan observasi yang tercemar *noise*. Prediksi menggunakan kecepatan untuk memperbarui posisi robot dan *update* menggunakan observasi posisi untuk memperbaiki estimasi.

Selain itu terdapat juga simulasi terakhir, yaitu Filter Partikel untuk Navigasi. Program ini digunakan untuk memperkirakan posisi robot. Pada setiap langkah, filter ini memperbarui posisi partikel berdasarkan kecepatan dan menambahkan *noise* untuk simulasi pergerakan. Bobot partikel diperbarui berdasarkan perbedaan antara posisi partikel dan observasi yang diperoleh. Setelah itu, dilakukan *resampling* untuk memilih partikel – partikel terbaik.

## **2. Simulasi *Kalman Filter* untuk Lokalitas Robot Menggunakan Webots**

Implementasi algoritma *Kalman Filter* untuk memperkirakan posisi robot berdasarkan data sensor jarak dan *encoder* roda. *Kalman Filter* digunakan untuk menggabungkan pengukuran sensor jarak dan prediksi dari *encoder* roda untuk mengurangi ketidakpastian dan mendapatkan estimasi posisi yang lebih akurat. Robot menggunakan dua motor roda yang dikontrol dalam mode kecepatan, serta *encoder* untuk membaca jarak tempuh setiap roda. *Input* prediksi dihitung dari rata-rata pergerakan kedua roda, sementara pembaruan posisi berasal dari sensor jarak. Filter ini menggunakan *Kalman Gain* untuk memberikan bobot pada masing-masing *input*. Penggunaan *Kalman Filter* untuk menangani *noise* pada data sensor, sehingga posisi robot dapat diperkirakan lebih presisi.