

## Лабораторная работа 7 — Расширенный фильтр Калмана

Используйте готовый проект `kf_framework`, в котором уже описана необходимая для работы расширенного фильтра Калмана последовательность действий и реализована визуализация.

В архиве `kf_framework` содержатся следующие папки:

`data` содержит файлы с данными о мире и показаниями датчиков.

`code` содержит “заготовку” для написания расширенного фильтра Калмана.

Вы можете попробовать запустить расширенный фильтр Калмана в терминале: `python kalman_filter.py`. Однако он будет работать корректно только после того, как вы дополните код.

- (a) Робот с дифференциальным приводом перемещается в плоскости, т.е. его состояние описывается тремя координатами  $\langle x, y, \theta \rangle$ . Модель процесса робота основана на одометрии. Напишите матрицу Якоби  $G_t$  (для незашумленной функции процесса  $g$ ) и реализуйте шаг предсказания `prediction_step` для расширенного фильтра Калмана, учитывая зашумленность модели:

$$Q_t = \begin{pmatrix} 0.2 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 0.2 \end{pmatrix}$$

- (b) Напишите матрицу Якоби  $H_t$  (для незашумленной функции измерений  $h$ ) и реализуйте шаг коррективки `correction_step` для расширенного фильтра Калмана, учитывая зашумленность датчиков:

$$R_t = \begin{pmatrix} 0.5 & 0 & 0 & \dots \\ 0 & 0.5 & 0 & \dots \\ 0 & 0 & 0.5 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \in \mathbb{R}^{\text{size}(z_t) \times \text{size}(z_t)}$$

*Подсказки:*

Для считывания данных, полученных с датчиков, и данных об ориентирах используются словари. Словари обеспечивают более простой способ доступа к структурам данных на основе одного или нескольких ключей. Функции `read_sensor_data` и `read_world_data` в файле `read_data.py` считывают данные из файлов и создают соответствующий словарь с отметками времени в качестве первичных ключей.

Чтобы получить доступ к данным датчика из словаря `sensor_readings`, используйте:

```
sensor_readings[timestamp, 'sensor']['id']
sensor_readings[timestamp, 'sensor']['range']
sensor_readings[timestamp, 'sensor']['bearing']
```

Чтобы получить доступ к данным одометрии, используйте:

```
sensor_readings[timestamp, 'odometry']['r1']
sensor_readings[timestamp, 'odometry']['t']
sensor_readings[timestamp, 'odometry']['r2']
```

Чтобы получить доступ к положениям ориентиров из словаря `landmarks`, используйте:

```
position_x = landmarks[id][0]
position_y = landmarks[id][1]
```