Huỳnh Chí Thành

20/04/2025

**BỘ LỌC KALMAN**

**1. Giới thiệu**

**1.1 Tác giả**

Bộ lọc Kalman là một thuật toán lọc đệ quy được phát triển bởi Rudolf E. Kálmán vào năm 1960. Thuật toán này đã trở thành một trong những công cụ quan trọng nhất trong lĩnh vực xử lý tín hiệu, điều khiển tự động, và các hệ thống điều hướng. Bộ lọc Kalman cho phép ước lượng trạng thái của hệ thống động dựa trên chuỗi các phép đo không hoàn hảo (chứa nhiễu).

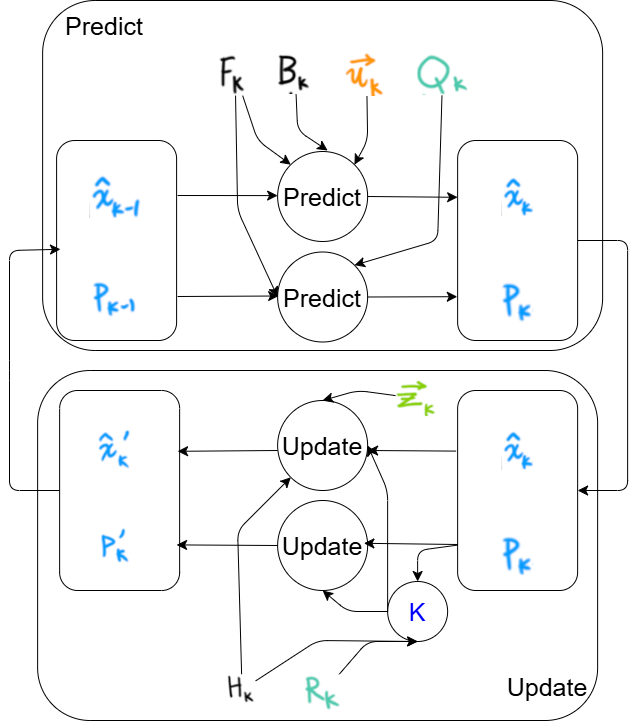
**1.2 Lý do chọn bộ lọc Kalman**

Bộ lọc Kalman được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau do khả năng:

* Khử nhiễu hiệu quả từ dữ liệu đo lường
* Ước lượng các biến trạng thái không thể đo trực tiếp
* Kết hợp tối ưu thông tin từ nhiều nguồn khác nhau
* Thích ứng tốt với các hệ thống thời gian thực

**2. Cơ sở lý thuyết**

Bộ lọc Kalman hoạt động theo quy trình hai bước lặp đi lặp lại: dự đoán (predict) và cập nhật (update). Thuật toán này dựa trên ý tưởng rằng, nếu chúng ta biết trạng thái hiện tại của hệ thống và mô hình động học của nó, chúng ta có thể dự đoán trạng thái tiếp theo. Sau đó, khi có phép đo mới, chúng ta kết hợp dự đoán này với phép đo để có được ước lượng tốt hơn.



Trong đó :

* **X**: Biến trạng thái, biểu diễn giá trị hiện tại được ước lượng.
* **P**: Độ không chắc chắn của ước lượng (covariance).
* **Q**: Nhiễu quá trình (process noise), phản ánh độ không chắc chắn trong mô hình hệ thống.
* **R**: Nhiễu đo lường (measurement noise), phản ánh độ không chắc chắn trong các phép đo.
* **K**: Hệ số Kalman (Kalman gain), xác định mức độ tin cậy giữa dự đoán và phép đo.
* **:**là ma trận chuyển trạng thái
* **:** là ma trận điều khiển đầu vào
* **:** là vector điều khiển (input)

**2.2 Công thức toán học**

* Bước Dự đoán (Prediction):
* Bước Cập nhật thông số (Update):

**3. Ứng dụng thực tế**

Khi Tag đứng yên, chúng ta có thể áp dụng hai bộ lọc Kalman độc lập cho hai tọa độ (x,y) để khử nhiễu từ ESP32. Các thông số được đơn giản hóa như sau:

* = 1 (ma trận đơn vị trong trường hợp vô hướng)
* \* = 0 (không có đầu vào điều khiển)
* = 1 nghĩa là bạn đang đo lường trực tiếp trạng thái hệ thống. Trong trường hợp của bạn, nếu X là vị trí x hoặc y, và measurement cũng là vị trí x hoặc y, thì = 1 là hợp lý.

Từ đó ta có bộ lọc Kalman đơn giản:

* Predict:
* Update: