# Il filtro di Kalman

Angelo Zagami - 13/10/2021

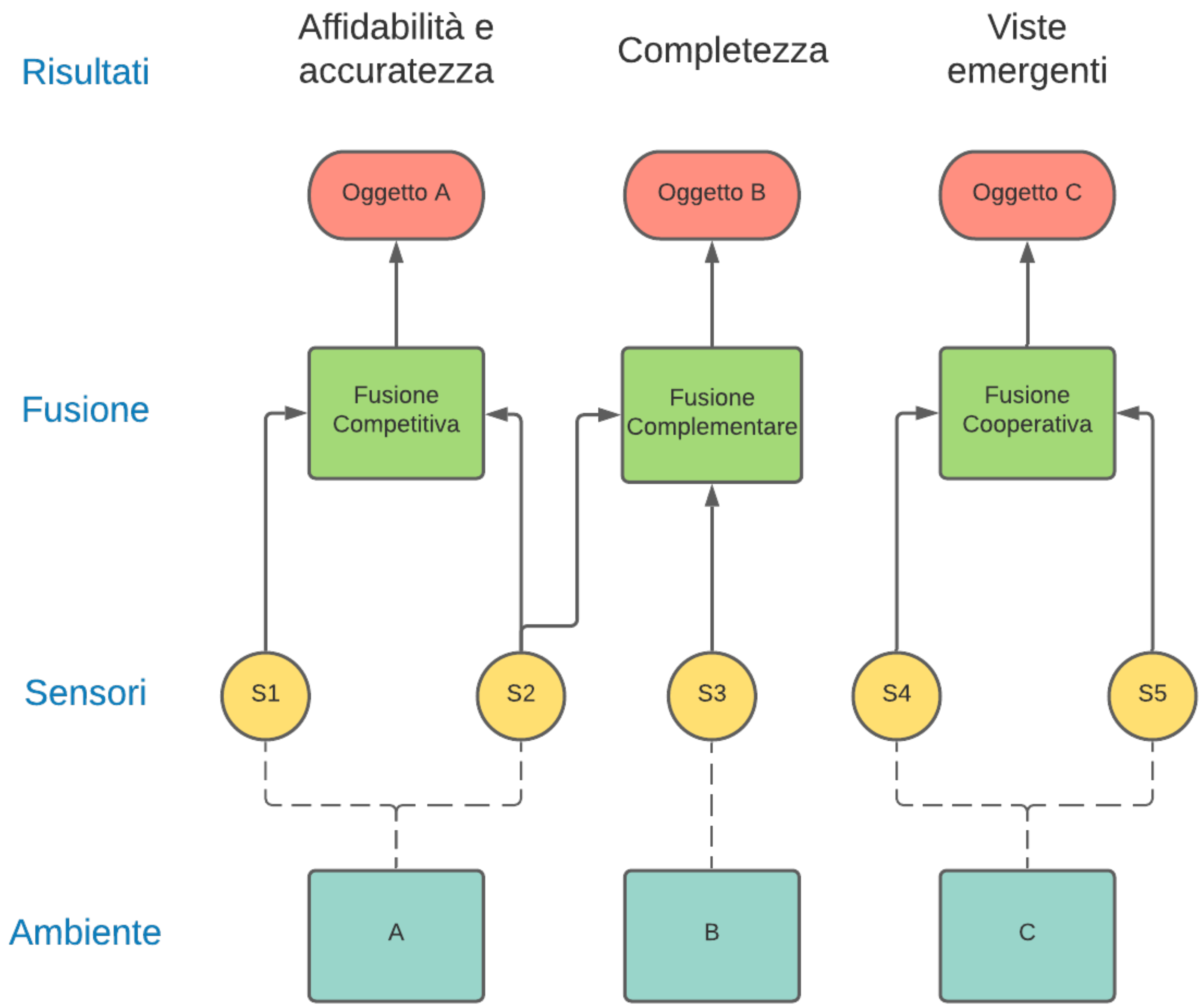
# Sommario

- Introduzione
- Filtro di Kalman
- Fase di init
- Fase di predict
- Fase di update
- Filtro di Kalman esteso (EKF)
- Live demo

# Introduzione

# Introduzione

La *fusione multisensoriale* è il processo di combinazione di dati provenienti da sensori (o altre fonti) in modo tale che le informazioni risultanti abbiano meno incertezza di quanto sarebbe possibile se esse fossero utilizzate singolarmente.



# Introduzione

Tecniche di sensor fusion più comuni:

- Algoritmi basati sul teorema del limite centrale
- Algoritmi basati su reti bayesiane
- Convolutional neural network
- Filtri di Kalman

# Filtro di Kalman

# Filtro di Kalman

È un processo matematico iterativo che utilizza una serie di equazioni e input di dati consecutivi per stimare rapidamente il vero valore di variabili come posizione, velocità, ecc. dell’oggetto misurato, nonostante i valori misurati contengano errori, incertezze o variazioni imprevedibili e/o casuali

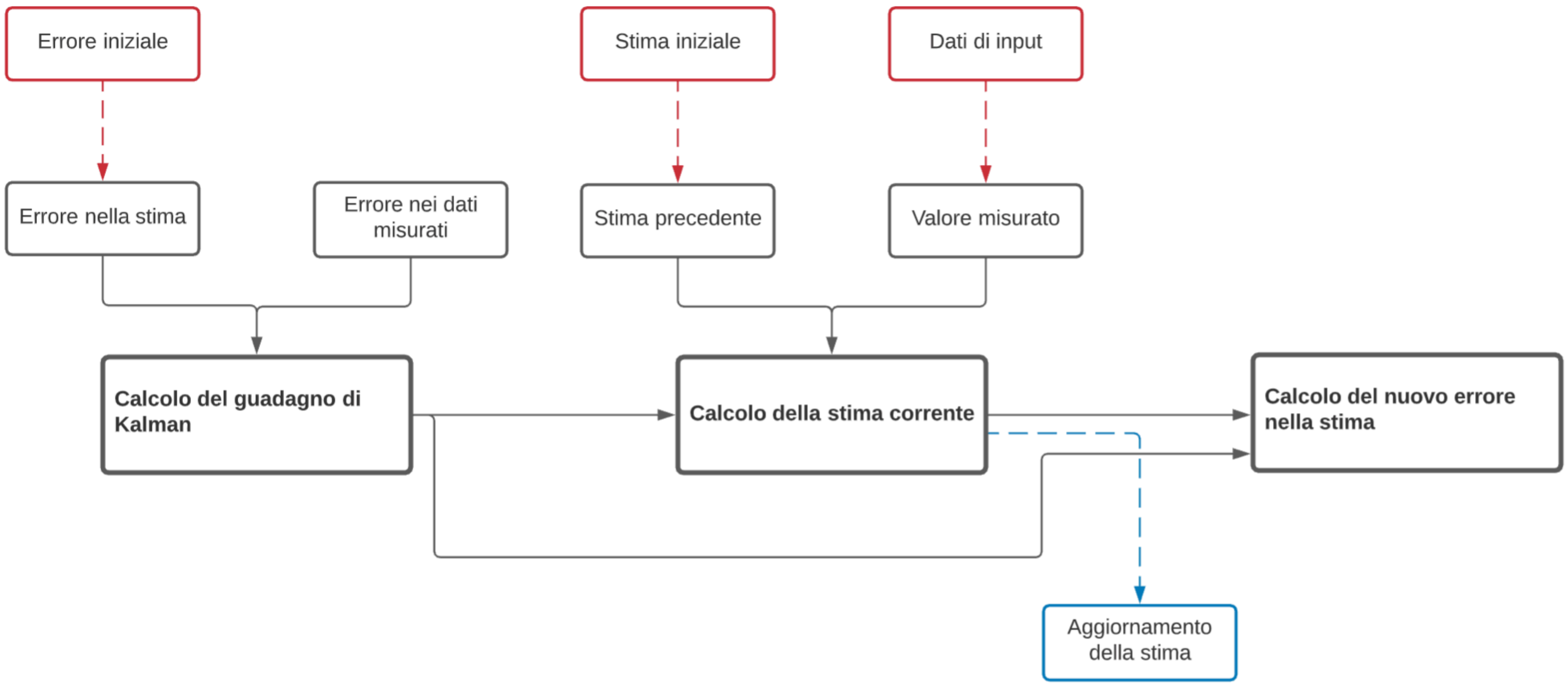
# Filtro di Kalman

Perché il filtro funzioni sono necessarie delle ipotesi semplificative:

- Il modello del sistema e il modello di osservazioni sono lineari
- Il rumore di qualsiasi natura, sia esso di processo o di misura, viene modellizzato con funzioni normali a media nulla, dunque di tipo Gaussiano
- Il rumore di processo (Q) e il rumore di misura (R) sono tra loro indipendenti
- Il rumore, di qualunque natura (Q e R), e le variabili di stato (x) sono tra loro indipendenti
- I processi stocastici Q e R sono incorrelati con l'ipotesi iniziali del vettore di stato e della matrice di covarianza

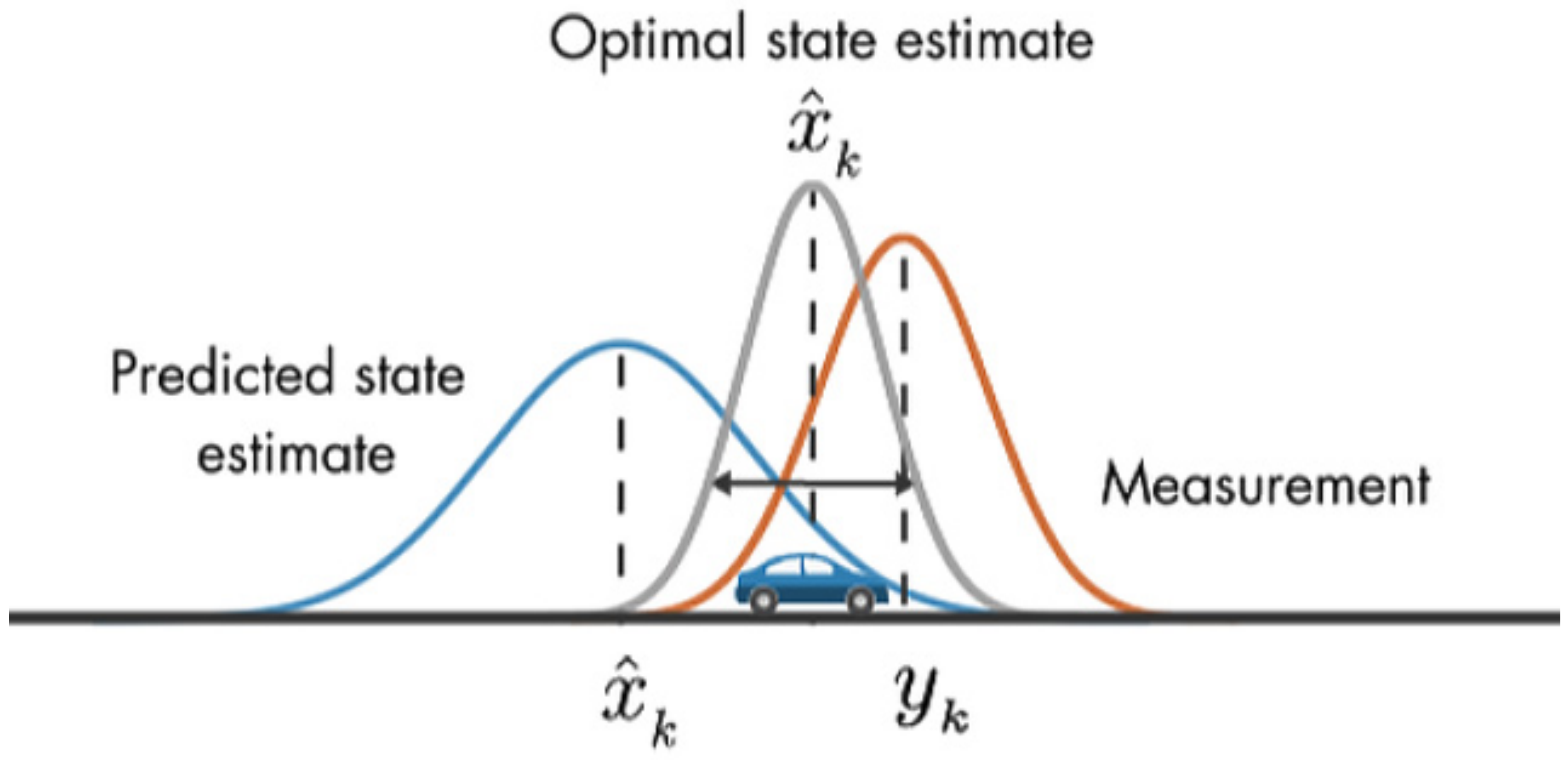
# Filtro di Kalman

È possibile schematizzare il funzionamento del filtro, ipotizzando il funzionamento con una sola variabile, come segue:



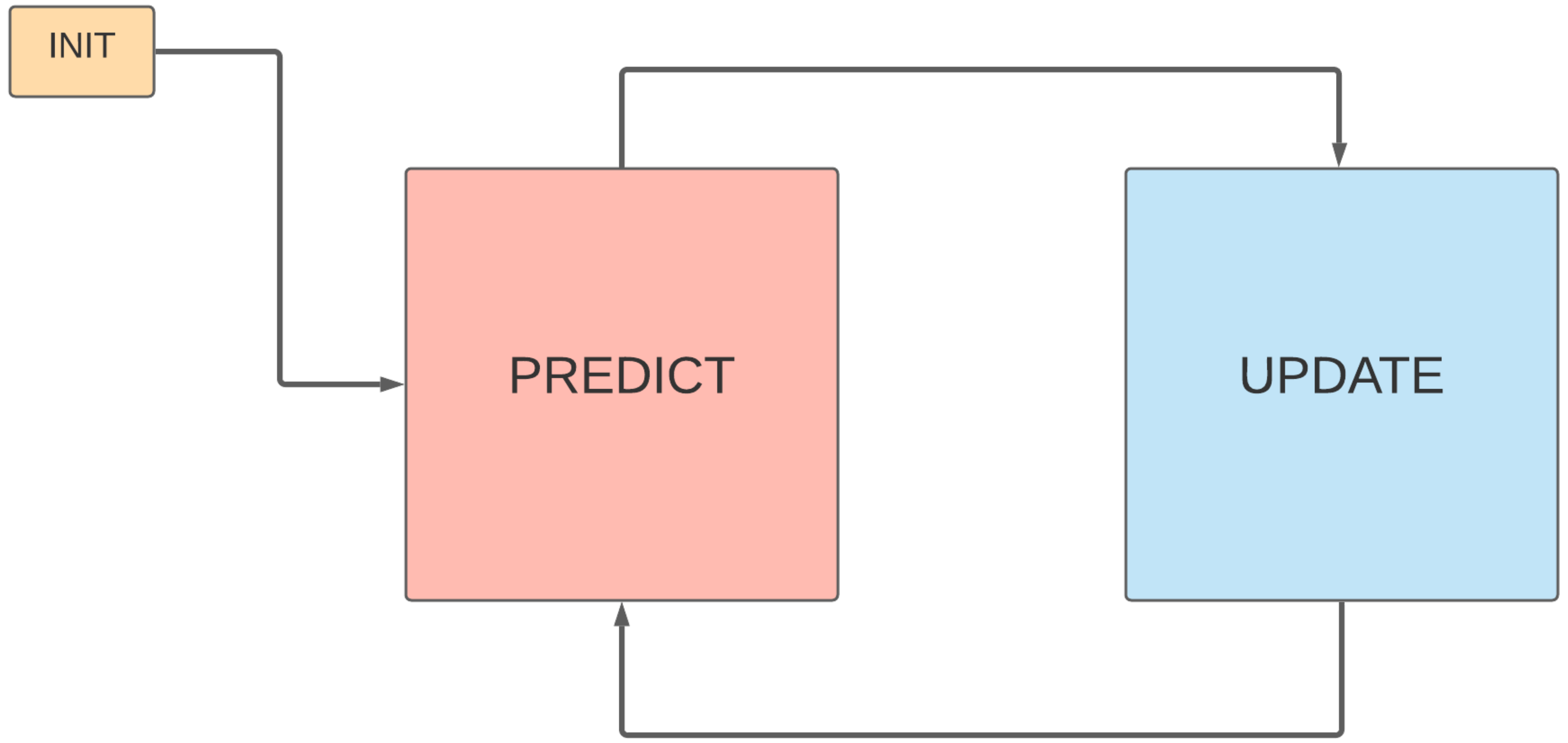
# Filtro di Kalman

Il parametro fondamentale del filtro è il **guadagno di Kalman**. Esso stabilisce il peso assegnato alla stima corrente e ai dati di misura, andando ad indicare quanto ciascuno dei due valori andrà ad incidere sulla nuova stima.



# Filtro di Kalman

L'algoritmo si compone di due fasi principali, ripetute in maniera alterna, e di una fase di inizializzazione dei parametri. Le due fasi sono dette *fase di predict* e *fase di update*




Fase di predict:

$$x_{k+1} = A_k x_k + B u_k + w_k$$

$$P_{k+1} = A P_k A^T + Q$$

Fase di update:

$$y_k = H x_k + z_k$$

$$K_k = \frac{P_k H^T}{H P_k H^T + R}$$

$$x_k = x_k + K_k [y_k - H x_k]$$

$$P_k = (I - K_k H) P_k$$

x: vettore di stato
A: matrice di transizione di stato
B: matrice di controllo
u: vettore di input
w: rumore sul processo

P: matrice di incertezza della stima
Q: matrice di covarianza del rumore di processo
y: vettore di misura
H: matrice di misura

z: rumore di misura
K: guadagno di Kalman
R: matrice di covarianza del rumore di misura
I: matrice identità

# Fase di init

# Fase di init

In questa fase viene inizializzato il filtro andando a definire il vettore di stato iniziale, indicato con $x_0$ e la matrice di incertezza sulla stima ad esso relativa, indicata con $P_0$. La scelta di $x_0$ influenza solo il tempo necessario al filtro a convergere al valore reale. La scelta di $P_0$, invece, è molto più importante. Essa dà informazioni su quanto il valore di $x_0$ sia preciso.

# Fase di predict

# Fase di predict

Questa fase si divide a sua volta in due passaggi fondamentali: la **predizione dello stato** e la **predizione della stima**.

# Fase di predict - State extrapolation

La predizione dello stato, o state extrapolation, consiste in un’equazione in grado di stimare lo stato futuro, indicato con $x_{k+1}$, a partire dallo stato attuale ($x_k$) e dal modello fisico del sistema.

# Fase di predict - State extrapolation

$$x_{k+1} = A_k x_k + B u_k + w_k$$

- $w_k$: e` il rumore che agisce sul processo, schematizzato come un rumore bianco con distribuzione normale a media nulla e varianza data dalla matrice Q;
- $u_k$: e` la matrice delle variabili di input o di controllo, con $u \in \mathscr{R}^l$. Essa include tutti quei parametri di input che agiscono sul sistema
- A: è la matrice di transizione di stato, descrivere i cambiamenti di stato nel sistema. È una matrice $n \times n$. La matrice A mette in relazione lo stato precedente con quello attuale in assenza di rumore;
- B: è la matrice di controllo, ha dimensioni n×l e mette in relazione le variabili di input con lo stato del sistema. Essa modifica il valore di $A_k x_k$ tenendo conto dell'influenza degli input.

# Fase di predict - Covariance extrapolation

Durante la fase di predizione di stato si va ad aggiornare la matrice di incertezza dello stato P, ottenendo così una stima dell'errore dello stato appena stimato. L'equazione con la quale questo avviene è la seguente:

$$P_{k+1} = AP_kA^T + Q$$

# Fase di predict - Covariance extrapolation

La matrice Q rappresenta la **matrice di covarianza del rumore di processo**. La matrice Q dovrebbe riflettere l'incertezza nel modello di stato assunto o qualsiasi caratteristica non considerata dal modello o non modellabili nello stato, come ad esempio l'attrito. La matrice Q gioca un ruolo molto importante nel funzionamento del filtro. Il valore di Q dovrebbe essere sufficientemente piccolo da mantenere il potenziale di apprendimento dalla misurazione ma non troppo grande per non far aumentare l'incertezza, rendendo di fatto inutile il processo di stima. La matrice Q è diagonale se il rumore non è correlato tra i vari stati del processo.

# Fase di predict - Covariance extrapolation

Una delle parti più complesse dell'applicazione del filtro è proprio il calcolo della matrice Q. Se il processo ha variabili di input un modo per calcolare la matrice è il seguente:

$$Q = B\sigma^2 B^T$$

Tuttavia, attraverso una fase di tuning, bisogna andare a ricavare il parametro $\sigma^2$. Bisogna tener presente che valori molto bassi della matrice Q causano del ritardo (lag) nella stima, viceversa, valori alti fanno si che la stima segua il valore delle misure, senza attenuarne il rumore

# Fase di update

# Fase di update

Durante la fase di update vengono letti i dati sensori, calcolato il guadagno di Kalman e aggiornato lo stato di conseguenza. Avviene dunque la correzione del valore di $x$ stimato nella fase di predict tenendo conto dei nuovi dati di input.

# Fase di update - Input

I dati di input corrispondenti alle misure alla k-esima iterazione, indicati con $y_k$ con $y \in \mathscr{R}^m$ sono dati dalla seguente equazione:

$$y_k = Hx_k + z_k$$

Dove:

- H: è la matrice di misurazione, ha dimensioni $m \times n$, rappresenta il modello di misurazione, mette in relazione lo stato con i dati di input;
- $z_k$: è il rumore di misurazione, una sorgente di rumore bianco con distribuzione normale a media nulla e varianza data dalla matrice R.

# Fase di update - Kalman Gain

Il guadagno di Kalman viene calcolato come segue:

$$K_k = \frac{P_k H^T}{H P_k H^T + R}$$

Dove R rappresenta la matrice di covarianza del rumore di misura, essa indica l'errore intrinseco della misurazione. Se le misure sono indipendenti tra loro la matrice R è diagonale. Essa si ottiene mantenendo lo stato del sistema costante e sottraendo alle misurazioni la loro media, ottenendo così il rumore.

# Fase di update - State update

A questo punto è possibile calcolare la stima dello stato corrente, andando a correggere quella precedentemente stimata mediante le informazioni acquisite e il guadagno di Kalman:

$$x_k = x_k + K_k[y_k - Hx_k]$$

# Fase di update - Covariance update

L'ultimo passaggio del filtro è quello di aggiornare la matrice di covarianza dello stato secondo i valori dell'iterazione corrente:

$$P_k = (I - K_k H)P_k$$

Dove I è una matrice identità delle stesse dimensioni di H.

# Fase di update

Alla prossima iterazione, il vettore di stato $x_k$ e la matrice $P_k$ diventeranno input per la prossima iterazione, assumendo il ruolo di stato precedente:

$$k \to k - 1$$

# Filtro di Kalman esteso

# Filtro di Kalman esteso

Il **filtro di Kalman esteso** (EKF - Extended Kalman filter) consente di eliminare il vincolo sulla linearità del sistema e sulla distribuzione Gaussiana del rumore, rendendo meno rigidi i requisiti rispetto alla sua versione base, seppur diminuendo la qualità delle stime che resta comunque più che valida. La densità di probabilità è approssimata da una gaussiana, che può distorcere la vera struttura e, a volte, potrebbe portare alla divergenza tra la previsione del filtro e le misurazioni.

# Filtro di Kalman esteso

L'EKF richiede che le funzioni di stato e di misura siano funzioni differenziabili, in modo tale che sia possibile effettuare la linearizzazione mediante l'espansione di Taylor del primo ordine. L'espansione di una funzione $f$ in $a$ è data dalla formula:

$$f(x) = f(a) + f'(a)(x - a)$$

# Filtro di Kalman esteso

Un generico processo stocastico è descritto da un sistema discreto:

$$\begin{cases} x_{k+1} & = f(x_k, u_k, v_k) \\ y_{k+1} & = h(x_k, w_k) \end{cases}$$

Le funzioni $f$ e $f$ sono rispettivamente la funzione di stato e la funzione di osservazione, mentre i termini $v_k$ e $w_k$ rappresentano il rumore di processo e di misura. La linearizzazione avviene mediante le seguenti equazioni:

$$A_{i,j} = \frac{\partial f_i}{\partial x_j}, \quad G_{i,j} = \frac{\partial h_i}{\partial y_j}$$

Dove $A$ e $G$ sono le matrici Jacobiane della funzione di stato e di osservazione.

Il processo di linearizzazione approssima $f$ e $f$ a funzioni lineari tangenti alle stesse.

# Filtro di Kalman esteso

Il filtro di Kalman esteso in generale non è uno stimatore ottimo. Inoltre, se la stima iniziale dello stato è errata, o se il processo è modellato in modo errato, il filtro può divergere rapidamente, a causa della sua linearizzazione. Un altro problema con il filtro di Kalman esteso è che la matrice di covarianza stimata tende a sottostimare la vera matrice di covarianza e quindi rischia di diventare inconsistente in senso statistico senza l'aggiunta di "rumore stabilizzante".

Detto questo, il filtro Kalman esteso può fornire prestazioni ragionevoli ed è probabilmente lo standard *de facto* nei sistemi di navigazione e GPS.

# Live demo

# Live demo

- Moto parabolico in 2D
- Driving Scenario Designer
- Simulink
- Simulink con EKF

# Grazie per l’attenzione!