



UNIVERSITÀ DEGLI STUDI DI ROMA
“TOR VERGATA”

FACOLTA' DI INGEGNERIA

Corso di laurea in Ingegneria Informatica

Tesi di laurea

**Filtro di Kalman Esteso per la
localizzazione di robot mobili**

Relatore
Prof. O. M. Grasselli

Correlatore
Dott. F. Martinelli

Laureando
Fulvio Forni

A.A. 2003/2004

Ringrazio il Prof. Grasselli per tutte le attenzioni ed i consigli per una corretta formulazione del filtro di Kalman, e il Dott. Martinelli per la pazienza e il tempo dedicatomi nonché per le innumerevoli correzioni.

ai miei genitori.

Indice

Introduzione	4
1 Filtro di Kalman discreto	5
1.1 Richiami di probabilità	5
1.1.1 Processi indipendenti e processi incorrelati	6
1.1.2 Densità spettrale di potenza e rumore bianco	7
1.2 Sistemi discreti e rumore bianco	8
1.3 Osservatori dello stato a tempo discreto	10
1.3.1 Caso stazionario	10
1.3.2 Caso non stazionario	11
1.4 Osservatore ottimo a tempo discreto	12
2 Fusione sensoriale con caratterizzazioni statistiche note	18
2.1 Relazioni di misura lineari	18
2.2 Relazioni di misura non lineari	20
3 Localizzazione robotica	22
3.1 Il carattere locale del problema	22
3.2 Modello cinematico e rilevamenti odometrici	23
3.3 Definizione del problema	25
3.4 Filtro di Kalman Esteso	25
3.4.1 Fase di predizione (prediction phase)	27
3.4.2 Predizione del valore misurato dai sonar	28
3.4.3 L'algoritmo: fase di stima (estimation phase)	29
3.4.4 Ancora su fase di predizione e fase di stima	33
4 Simulazioni	34
4.1 Caratteristiche robot	34
4.2 Ambiente di simulazione	35
4.3 Analisi dei disturbi	35
4.4 Risultati delle simulazioni	36
Conclusioni	42
Bibliografia	50

Introduzione

Il problema della *localizzazione*, in robotica mobile, consiste nella stima della posizione del robot mediante acquisizione di informazioni dall'ambiente. L'insieme dei rilevamenti è, in generale, di natura eterogenea e caratterizzato dalla presenza di errori di misura: sono necessarie procedure in grado di integrare fra loro le informazioni provenienti da differenti dispositivi sensoriali, per ottenere una stima di posizione accurata. Tecniche di *fusione sensoriale* possono essere utilizzate quando l'insieme dei rilevamenti abbia caratteristiche di ridondanza e complementarità. Quando più sensori forniscono informazione ridondante si ha un accrescimento dell'affidabilità delle misure rispetto all'eventuale malfunzionamento del singolo sensore. Per complementarità si fa riferimento ad un rilevamento effettuato su diversi principi fisici, avente pertanto differente andamento degli errori: le misure inerenti ad una medesima caratteristica ambientale avranno una certa correlazione mentre le relative incertezze saranno incorrelate.

E' difficile ipotizzare metodi generali per la progettazione di un sistema di fusione sensoriale, in quanto fortemente vincolato all'architettura di elaborazione e rilevamento dei dati. Si devono valutare accuratamente gli utilizzi per i quali il robot è stato pensato, il tipo di ambiente nel quale opererà, le tipologie e le prestazioni dei vari sensori presenti nonché le caratteristiche di movimento quali, ad esempio, massima velocità o raggio d'azione.

Nella tesi sviluppata sono stati elaborati dati odometrici (encoder ottici sulle ruote) e rilevamenti acustici, con una metodologia basata sul filtro di Kalman esteso. L'impiego di questo filtro è immediatamente giustificato quando si consideri la rumorosità di sensori acustici a basso costo e l'intrinseca fragilità dell'integrazione diretta di dati tachimetrici, dalla quale dipende anche la necessità dell'impiego di tecniche di fusione sensoriale. La semplice integrazione numerica dei rilevamenti odometrici, infatti, porta rapidamente ad un disallineamento tra la posizione effettiva del robot e quella misurata (slittamenti, presenza di discontinuità sul suolo, piccoli urti, disallineamento delle ruote, errata calibrazione dei raggi, etc.).

Il discorso è strutturato sullo sviluppo del filtro di Kalman, a tempo discreto, per sistemi stazionari e non stazionari ed è poi particolarizzato nel dettaglio dell'architettura hardware. Alla fine sono riportati i risultati ottenuti, in particolare il confronto tra la posizione stimata con l'uso del filtro e la semplice integrazione numerica.

Capitolo 1

Filtro di Kalman discreto

1.1 Richiami di probabilità

Definizione 1 Si consideri un processo stocastico discreto $v(i) \in \mathbb{R}^n$, $i \in \mathbb{Z}$, $n \in \mathbb{N}$. Si definiscono le seguenti quantità:

- $m(i) \triangleq E\{v(i)\}$
media del processo,
- $C_v(i, j) \triangleq E\{v(i)v'(j)\}$, $i, j \in \mathbb{Z}$
matrice dei momenti congiunti del secondo ordine (second order joint moment matrix)
- $R_v(i, j) \triangleq E\{[v(i) - m(i)][v(j) - m(j)]'\}$, $i, j \in \mathbb{Z}$
matrice di covarianza (covariance matrix)
- $Q(i) \triangleq E\{[v(i) - m(i)][v(i) - m(i)]'\}$, $i \in \mathbb{Z}$
matrice di varianza (variance matrix) corrispondente a $R_v(i, j)$ per $i = j$.

Nota 1 $Q(i)$ è simmetrica.

dimostrazione

Considerando $v_j(i)$ come il j -esimo elemento del vettore $v(i)$ e definito $a_j \triangleq v_j(i) - m_j(i)$ si può riscrivere:

$$Q(i) = E \left\{ \begin{pmatrix} a_1 a_1 & a_1 a_2 & \cdots & a_1 a_n \\ a_2 a_1 & a_2 a_2 & \cdots & a_2 a_n \\ \vdots & \vdots & \ddots & \vdots \\ a_n a_1 & a_n a_2 & \cdots & a_n a_n \end{pmatrix} \right\} = E \left\{ \begin{pmatrix} a_1 a_1 & a_2 a_1 & \cdots & a_n a_1 \\ a_1 a_2 & a_2 a_2 & \cdots & a_n a_2 \\ \vdots & \vdots & \ddots & \vdots \\ a_1 a_n & a_2 a_n & \cdots & a_n a_n \end{pmatrix} \right\} \quad (1.1)$$

per la commutatività del prodotto tra scalari. Quindi

$$Q(i) = Q'(i). \quad (1.2)$$

■

Nota 2 $Q(i)$ è *semidefinita positiva*.

dimostrazione

Si consideri $x \in \mathbb{R}^n$ vettore generico. La condizione su $Q(i)$ può essere riscritta come

$$\begin{aligned} x' Q(i) x &\geq 0 \quad \forall x \\ &\Rightarrow \\ x' Q(i) x &= x' E\{[v(i) - m(i)][v(i) - m(i)]'\} x \\ &= E\{x' [v(i) - m(i)][v(i) - m(i)]' x\}. \end{aligned} \quad (1.3)$$

Il prodotto $x' [v(i) - m(i)]^1$ ha come risultato un numero scalare pertanto

$$\begin{aligned} x' Q(i) x &= E\{x' [v(i) - m(i)][v(i) - m(i)]' x\}. \\ &= E\{(x' [v(i) - m(i)])^2\} \geq 0 \end{aligned} \quad (1.4)$$

■

Se il processo è stazionario, $m(i)$ e $Q(i)$ sono indipendenti da i e costanti, e $C_v(i, j)$ e $R_v(i, j)$ dipendono solo dalla differenza tra i e j .

1.1.1 Processi indipendenti e processi incorrelati

Due eventi A,B si dicono *statisticamente indipendenti* se, definita $P(x)$ la probabilità dell'evento x ,

$$P(AB) = P(A)P(B)$$

dove $P(AB)$ si riferisce alla probabilità di $(A \cap B)^2$.

Due processi stocastici discreti $v(i), w(i) \in \mathbb{R}^n$ si dicono *indipendenti* se

$$P[v(i) \cap w(j)] = P[v(i)]P[w(j)] \quad \forall i, j \in \mathbb{Z} \quad (1.5)$$

ovvero se sono indipendenti due generiche realizzazioni.

Due processi stocastici, discreti $v(i), w(i)$ sono incorrelati se

$$\begin{aligned} R_{v,w}(i, j) &= E\{[v(i) - \bar{v}(i)][w(j) - \bar{w}(j)]'\} = 0 \quad \forall i, j \in \mathbb{Z} \\ \bar{v}(i) &= E(v(i)) \\ \bar{w}(i) &= E(w(i)) \end{aligned} \quad (1.6)$$

¹Ed ugualmente $[v(i) - m(i)]' x$

²Intersezione, AND logico

1.1.2 Densità spettrale di potenza e rumore bianco

Definizione 2 Per un processo $v(i) \in \mathbb{R}^n$ a tempo discreto, stocastico, stazionario in senso lato, si definisce, se esiste, matrice delle densità spettrali di potenza (power spectral density matrix),

$$\Sigma_v(\theta) = \sum_{i=-\infty}^{\infty} z^{-i} R_v(i), \quad z = e^{j\theta}, \quad -\pi \leq \theta < \pi \quad (1.7)$$

con $R_v(i-k)$ matrice delle covarianze, dipendente unicamente dalla differenza $i-k$.

Teorema 1 Sia $v(i) \in \mathbb{R}^n$, $i \in \mathbb{Z}$, stocastico, stazionario in senso lato, a media nulla con densità spettrale di potenza $\Sigma_v(\theta)$, allora

$$E\{v(i)v(i)'\} = R_v(0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \Sigma_v(\theta) d\theta. \quad (1.8)$$

dimostrazione Valgono le seguenti uguaglianze:

$$\begin{aligned} \frac{1}{2\pi} \int_{-\pi}^{\pi} \Sigma_v(\theta) d\theta &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\sum_{i=-\infty}^{\infty} R_v(i) e^{-j\theta i} \right) d\theta = \\ \sum_{i=-\infty}^{\infty} R_v(i) \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} e^{-j\theta i} d\theta \right) &= R_v(0) \end{aligned} \quad (1.9)$$

La seconda uguaglianza vale grazie alla coincidenza tra la serie degli integrali e l'integrale della serie. L'integrale è un operatore lineare e, se non altera le caratteristiche di convergenza della serie, può essere applicato singolarmente ad ogni elemento della sommatoria. In questo caso l'intervallo di integrazione è finito $[-\pi, \pi]$ e la funzione integranda è limitata nell'intervallo specificato, inoltre, per la periodicità della funzione integranda, il risultato dell'integrazione è sempre zero per $i \neq 0$. L'integrale non altera quindi la caratteristica di convergenza della serie ed è pertanto valida la seconda uguaglianza. ■

Definizione 3 Si definisce rumore bianco (white noise) un processo stocastico $w(i) \in \mathbb{R}^n$ a media nulla caratterizzato da:

$$\begin{aligned} E\{w(i)w'(i)\} &= V(i) \\ E\{w(i)w'(j)\} &= 0 \quad \forall i, j, i \neq j \end{aligned} \quad (1.10)$$

con $V(i) = V'(i) \geq 0$. Se $V(i)$ non dipende da i è anche stazionario in senso lato.

Per la matrice $V(i)$ valgono le stesse considerazioni delle note 1 e 2 sviluppate sulla matrice $Q(i)$, cui si deve aggiungere la restrizione $V(i) > 0$ ³. Tale restrizione è necessaria quando si consideri il rumore introdotto come errore sulla misura effettuata⁴: si indica in questo modo l'assenza di misure perfette, esenti da errore. Nel paragrafo 1.4 verrà svolta un'analisi più approfondita di una matrice simile alla $V(i)$ qui introdotta⁵.

³La notazione $V(i) > 0$ indica $\nexists \eta \neq 0$ tale che $\eta' V(i) \eta = 0$.

⁴Valore di misura rispetto a valore vero

⁵E' necessario definire meglio il ruolo del rumore bianco nello studio degli osservatori asintotici ed in particolar modo in relazione all'ottimizzazione degli stessi

1.2 Sistemi discreti e rumore bianco

Si consideri il sistema

$$x(i+1) = A(i)x(i) + B(i)w(i) \quad (1.11)$$

con $x(i) \in \mathbb{R}^n$ e $w(i) \in \mathbb{R}^p$ rumore bianco con matrice delle varianze $V(i)$.

Teorema 2 *Dato un sistema caratterizzato dall'equazione (1.11)⁶, supposto $x(i_0) \triangleq x_0$ sia un vettore di variabili aleatorie (realizzazione del processo al tempo i_0) incorrelato con il rumore $w(i)$ ⁷, e definendo:*

$$\bar{x}(i) \triangleq E\{x(i)\} \quad (1.12)$$

$$Q(i) \triangleq E\{[x(i) - \bar{x}(i)][x(i) - \bar{x}(i)]'\} \quad (1.13)$$

$$\bar{x}_0 \triangleq \bar{x}(i_0) = E\{x(i_0)\} \quad (1.14)$$

$$Q_0 \triangleq Q(i_0) = E\{[x_0 - \bar{x}_0][x_0 - \bar{x}_0]'\} \quad (1.15)$$

valgono le seguenti relazioni:

$$\bar{x}(i) = \Phi(i, i_0)\bar{x}_0 \quad i \geq i_0 \quad (1.16)$$

$$Q(i+1) = A(i)Q(i)A'(i) + B(i)V(i)B'(i) \quad (1.17)$$

con $\Phi(i, i_0)$ matrice di transizione dello stato, espressa da:

$$\begin{aligned} \Phi(i_0, i_0) &= I \\ \Phi(i, i_0) &= A(i-1)A(i-2)\dots A(i_0) \quad i > i_0 \end{aligned} \quad (1.18)$$

dimostrazione Considerando la soluzione dell'equazione (1.11) alle differenze in forma esplicita

$$x(i) = \Phi(i, i_0)x_0 + \sum_{h=i_0}^{i-1} \Phi(i, h+1)B(h)w(h), \quad (1.19)$$

si può calcolare il primo membro della (1.12):

$$\begin{aligned} \bar{x}(i) &= E\{x(i)\} = E\{\Phi(i, i_0)x(i_0) + \sum_{h=i_0}^{i-1} \Phi(i, h+1)B(h)w(h)\} \\ &= \Phi(i, i_0)E\{x_0\} + \sum_{h=i_0}^{i-1} \Phi(i, h+1)B(h)E\{w(h)\} \\ &= \Phi(i, i_0)E\{x_0\} = \Phi(i, i_0)\bar{x}_0 \end{aligned}$$

⁶Con rumore bianco già definito

⁷ $E\{[x_0 - \bar{x}_0][w(i) - \bar{w}(i)]'\} = E\{[x_0 - \bar{x}_0]w'(i)\} = 0$

ottenendo così la (1.16). In modo analogo:

$$\begin{aligned}
Q(i) &= E\{[x(i) - \bar{x}(i)][x(i) - \bar{x}(i)]'\} \\
Q(i+1) &= E\{[x(i+1) - \bar{x}(i+1)][x(i+1) - \bar{x}(i+1)]'\} \\
&\Rightarrow \\
Q(i+1) &= E\{[A(i)x(i) + B(i)w(i) - A(i)\bar{x}(i)][A(i)x(i) + B(i)w(i) - A(i)\bar{x}(i)]'\} \\
&= A(i)E\{[x(i) - \bar{x}(i)][x(i) - \bar{x}(i)]' A'(i)\} + B(i)E\{w(i)w'(i)\}B'(i) \\
&\quad + A(i)E\{[x(i) - \bar{x}(i)][w'(i)]\}B'(i) + B(i)E\{w(i)[x(i) - \bar{x}(i)]'\}A'(i) \\
&= A(i)Q(i)A'(i) + B(i)V(i)B'(i) \\
&\quad + A(i)E\left\{\left[\Phi(i, i_0)x_0 + \sum_{h=i_0}^{i-1} \Phi(i, h+1)B(h)w(h) - \Phi(i, i_0)\bar{x}_0\right][w'(i)]\right\}B'(i) \\
&\quad + B(i)E\left\{[w(i)]\left[\Phi(i, i_0)x_0 + \sum_{h=i_0}^{i-1} \Phi(i, h+1)B(h)w(h) - \Phi(i, i_0)\bar{x}_0\right]'\right\}A'(i) \\
&= A(i)Q(i)A'(i) + B(i)V(i)B'(i) \\
&\quad + A(i)\left(\Phi(i, i_0)E[(x_0 - \bar{x}_0)w'(i)] + \left[\sum_{h=i_0}^{i-1} \Phi(i, h+1)B(h)E[w(h)w'(i)]\right]\right)B'(i) \\
&\quad + B(i)\left(E[w(i)(x_0 - \bar{x}_0)']\Phi'(i, i_0) + \left[\sum_{h=i_0}^{i-1} E[w(i)w'(h)]B'(h)\Phi'(i, h+1)\right]\right)A'(i) \\
&= A(i)Q(i)A'(i) + B(i)V(i)B'(i)
\end{aligned}$$

dove l'ultima uguaglianza è dovuta alle ipotesi su $w(i)$ (in particolare la (1.10)) e all'incorrelazione tra x_0 e $w(i)$. Si è ottenuta la (1.17). ■

Si noti che la (1.17) consente il calcolo ricorsivo di $Q(i)$ noto il valore iniziale Q_0 della matrice di varianza.

Vale inoltre il seguente teorema (non dimostrato)

Teorema 3 *Per il sistema (1.11), sotto le stesse ipotesi del Teorema 2, se $\text{Re}[\lambda_i] < 1$ per ogni autovalore della matrice A , e se inoltre la matrice di varianza $V(i)$ di $w(i)$ è costante, allora per $i_0 \rightarrow -\infty$ la matrice di covarianza $R_x(i, j)$ tende ad un valore asintotico che dipende unicamente dalla differenza tra i e j . Inoltre sotto le stesse ipotesi, per $i_0 \rightarrow -\infty$, la matrice delle varianze $Q(i)$ tende ad un valore asintotico calcolabile risolvendo l'equazione algebrica*

$$Q = AQA' + BV B'. \quad (1.20)$$

di cui è l'unica soluzione.

1.3 Osservatori dello stato a tempo discreto

1.3.1 Caso stazionario

Si consideri un sistema S lineare e stazionario, di equazioni

$$\begin{aligned}x(i+1) &= Ax(i) + Bu(i) \\ y(i) &= Cx(i) + Du(i) \\ x(i) &\in \mathbb{R}^n, u(i) \in \mathbb{R}^p \text{ e } y(i) \in \mathbb{R}^q.\end{aligned}\tag{1.21}$$

si vuole un sistema dinamico, avente stato $\xi(i)$ della stessa dimensione di $x(i)$, per il quale valgano:

- a.** $\xi(i)$ converga asintoticamente a $x(i)$, $\forall x(i_0), \forall \xi(i_0), \forall u(\cdot)$, eventualmente con rapidità di convergenza preassegnabile;
- aa.** se $\xi(i_0) = x(i_0)$ allora $\xi(i) = x(i) \quad \forall i \geq 0, \forall x(0)$ del sistema S, $\forall u(\cdot)$.

Un sistema dinamico che rispetti le **a.**, **aa.** è un osservatore dello stato del sistema S. Gli ingressi dell'osservatore sono costituiti, in generale, dalla misura dell'uscita $y(i)$ del sistema S e dall'ingresso $u(i)$ e, nel caso ideale di assenza di errori o rumore di misura, permettono di risalire allo stato di S, se il sistema è osservabile. L'osservatore sarà descritto dall'equazione:

$$\xi(i+1) = H\xi(i) + Vy(i) + Ru(i)\tag{1.22}$$

con le matrici H, V, R definite in funzione delle condizioni **a.** e **aa.**.

Definendo l'errore

$$\begin{aligned}e(i) &\triangleq x(i) - \xi(i) \\ \Rightarrow \\ e(i+1) &= x(i+1) - \xi(i+1) \\ &= Ax(i) + Bu(i) - H\xi(i) - Vy(i) - Ru(i) \\ &= Ax(i) + Bu(i) - H\xi(i) - V(Cx(i) + Du(i)) - Ru(i) \\ &= He(i) - Hx(i) - V(Cx(i) + Du(i)) - Ru(i) + Ax(i) + Bu(i) \\ &= He(i) + (A - VC - H)x(i) + (B - VD - R)u(i)\end{aligned}\tag{1.24}$$

L'ipotesi **a.** implica che il sistema d'errore abbia convergenza asintotica a zero con velocità preassegnabile mentre l'ipotesi **aa.** impone $e(i) = 0$ se $e(0) = 0$.

La condizione **aa.** sulla (1.24), per il caso particolare $u(i) = 0, \forall i \geq 0$, implica che

$$H = A - VC\tag{1.25}$$

altrimenti con $e(0) = 0$ non si avrebbe $e(i) = 0, \forall i \geq 0$.

Per $u(i) \neq 0$, la condizione **aa.**, impone

$$R = B - VD\tag{1.26}$$

La condizione **a.** infine obbliga gli autovalori della matrice H ad essere interni alla circonferenza unitaria ($i \in \mathbb{Z}$). Si ottiene:

$$\xi(i+1) = (A - VC)\xi(i) + Vy(i) + (B - VD)u(i) \quad (1.27)$$

e il sistema (1.27) è un osservatore se gli autovalori della matrice H sono interni alla circonferenza di raggio unitario. E' possibile soddisfare l'ultima specifica

- se la coppia (A, C) è osservabile;
- Se la coppia (A, C) è rilevabile, ovvero la parte inosservabile del sistema ha già autovalori di modulo minore di 1.

Sotto queste condizioni un'opportuna scelta della matrice V sposta gli autovalori della H all'interno della circonferenza unitaria. Nel caso della prima affermazione inoltre può essere imposto qualsiasi insieme di autovalori.

1.3.2 Caso non stazionario

Si consideri un sistema S lineare, non stazionario, di equazioni

$$\begin{aligned} x(i+1) &= A(i)x(i) + B(i)u(i) \\ y(i) &= C(i)x(i) + D(i)u(i) \\ x(i) &\in \mathbb{R}^n, u(i) \in \mathbb{R}^p \text{ e } y(i) \in \mathbb{R}^q. \end{aligned} \quad (1.28)$$

si vuole un sistema dinamico, avente stato $\xi(i)$ della stessa dimensione di $x(i)$, per il quale valgano le **a.**, **aa.** Se valgono le condizioni il sistema sarà un osservatore dello stato.

L'osservatore sarà descritto dall'equazione:

$$\xi(i+1) = H(i)\xi(i) + V(i)y(i) + R(i)u(i) \quad (1.29)$$

con le matrici $H(i), V(i), R(i)$ definite in funzione delle condizioni **a.** e **aa.**

Definendo l'errore

$$e(i) \triangleq x(i) - \xi(i) \quad (1.30)$$

$\Rightarrow$

$$\begin{aligned} e(i+1) &= x(i+1) - \xi(i+1) \\ &= A(i)x(i) + B(i)u(i) - H(i)\xi(i) - V(i)y(i) - R(i)u(i) \\ &= A(i)x(i) + B(i)u(i) - H(i)\xi(i) - V(i)(C(i)x(i) + D(i)u(i)) - R(i)u(i) \\ &= H(i)e(i) - H(i)x(i) - V(i)(C(i)x(i) + D(i)u(i)) - R(i)u(i) + A(i)x(i) + B(i)u(i) \\ &= H(i)e(i) + (A(i) - H(i) - V(i)C(i))x(i) + (B(i) - R(i) - V(i)D(i))u(i) \end{aligned} \quad (1.31)$$

La condizione **aa.** sulla (1.31), per il caso particolare $u(i) = 0, \forall i \geq 0$, implica che

$$H(i) = A(i) - V(i)C(i) \quad (1.32)$$

Per $u(i) \neq 0$, la condizione **aa.**, impone

$$R(i) = B(i) - V(i)D(i) \quad (1.33)$$

La condizione **a.** infine obbliga il sistema d'errore ad essere asintoticamente stabile. Si ottiene:

$$\xi(i+1) = (A(i) - V(i)C(i))\xi(i) + V(i)y(i) + (B(i) - V(i)D(i))u(i) \quad (1.34)$$

e il sistema (1.34) è un osservatore se è asintoticamente stabile. Vale infatti il seguente teorema:

Teorema 4 *Per un osservatore descritto nell'equazione (1.34) con dinamica d'errore*

$$e(i+1) = (A(i) - V(i)C(i))e(i) \quad (1.35)$$

l'errore di ricostruzione ha la proprietà

$$e(i) \rightarrow 0 \text{ per } i \rightarrow \infty \quad \forall e(i_0),$$

se e solo se l'osservatore è asintoticamente stabile

1.4 Osservatore ottimo a tempo discreto

Obiettivo del paragrafo è derivare un osservatore, strutturalmente identico a quanto definito precedentemente, che minimizzi l'errore quadratico medio sulla ricostruzione dello stato.

Definizione 4 *Si consideri il sistema*

$$\begin{aligned} x(i+1) &= A(i)x(i) + B(i)u(i) + w_1(i) \\ y(i) &= C(i)x(i) + E(i)u(i) + w_2(i) \end{aligned} \quad i \geq i_0 \quad (1.36)$$

dove $[w'_1(i), w'_2(i)]'$ è un rumore bianco con matrice di varianza

$$V(i) = \begin{bmatrix} V_{11}(i) & V_{12}(i) \\ V_{12}(i) & V_{22}(i) \end{bmatrix}, \quad i \geq i_0 \quad (1.37)$$

e $x(i_0)$ è una variabile aleatoria vettoriale, incorrelata con w_1 e w_2 ovvero $E\{[x(i_0) - \bar{x}_0][w'_1]'\} = 0$ e $E\{[x(i_0) - \bar{x}_0][w'_2]'\} = 0$, caratterizzata da

$$E\{x(i_0)\} = \bar{x}_0, \quad E\{[x(i_0) - \bar{x}_0][x(i_0) - \bar{x}_0]'\} = Q_0, \quad (1.38)$$

nonché il suo l'osservatore di ordine pieno:

$$\hat{x}(i+1) = A(i)\hat{x}(i) + B(i)u(i) + K(i)[y(i) - C(i)\hat{x}(i) - E(i)u(i)] \quad (1.39)$$

Il problema dell'osservatore ottimo consiste nel trovare la sequenza dei guadagni $K^o(i_0), \dots, K^o(i-1)$, e lo stato iniziale $\hat{x}(i_0)$, che minimizzino il funzionale

$$E\{e'(i)e(i)\} \quad (1.40)$$

con $e(i) = x(i) - \hat{x}(i)$.

Per risolvere univocamente il problema dell'osservatore ottimo è necessario che la matrice $V_{22}(i)$ sia definita positiva $\forall i \geq i_0$ (ove i_0 è l'istante d'inizio delle osservazioni o misure sul sistema). Tale ipotesi non è lontana da una valutazione realistica del problema. Se per $i \geq i_0$ fosse $V_{22}(i) \geq 0$ ma non definita positiva, esisterebbe un $\eta \in \mathbb{R}^q$, $\eta \neq 0$ tale che $\eta' V_{22}(i) \eta = 0$, ovvero $E\{\eta' w_2(i) w_2'(i) \eta\} = 0$, cioè la variabile aleatoria $\eta' w_2(i)$ avrebbe valore medio del proprio quadrato nullo; ovvero la misura (scalare) $\eta' y(i)$ sarebbe da considerare non affetta da rumori o errori aleatori di misura.

Si considera il sistema d'errore:

$$\begin{aligned} e(i+1) &= [A(i) - K(i)C(i)]e(i) + w_1(i) - K(i)w_2(i) \quad \forall i \geq i_0 \\ &\Rightarrow \\ e(i+1) &= [A(i) - K(i)C(i)]e(i) + \begin{bmatrix} I & -K(i) \end{bmatrix} \begin{bmatrix} w_1(i) \\ w_2(i) \end{bmatrix} \end{aligned} \quad (1.41)$$

per il quale si può introdurre una matrice di varianza

$$\tilde{Q}(i) = E\{[e(i) - \bar{e}(i)][e(i) - \bar{e}(i)]'\} \quad (1.42)$$

Si opera sul funzionale (1.40):

$$\begin{aligned} E\{e'(i)e(i)\} &= E\{[\bar{e}(i) + (e(i) - \bar{e}(i))]' [\bar{e}(i) + (e(i) - \bar{e}(i))]\} \\ &= E\{\bar{e}'(i)\bar{e}(i)\} + E\{[e(i) - \bar{e}(i)]' [e(i) - \bar{e}(i)]\} + \\ &\quad + E\{\bar{e}'(i)[e(i) - \bar{e}(i)]\} + E\{[e(i) - \bar{e}(i)]' \bar{e}(i)\} \\ &= \bar{e}'(i)\bar{e}(i) + E\{[e(i) - \bar{e}(i)]' [e(i) - \bar{e}(i)]\} + \\ &\quad + \bar{e}'(i)E\{[e(i) - \bar{e}(i)]\} + E\{[e(i) - \bar{e}(i)]' \} \bar{e}(i) \\ &= \bar{e}(i)' \bar{e}(i) + E\{[e(i) - \bar{e}(i)]' [e(i) - \bar{e}(i)]\} \\ &= \bar{e}'(i)\bar{e}(i) + \text{tr}[\tilde{Q}(i)] \end{aligned} \quad (1.43)$$

Il prodotto $\bar{e}'(i)\bar{e}(i)$ può essere minimizzato scegliendo $\hat{x}(i_0) = \bar{x}_0$. Infatti $E\{e(i_0)\} = E\{x(i_0) - \hat{x}(i_0)\} = 0$ per la scelta di $\hat{x}(i_0)$ effettuata. Considerando poi $e(i) = \check{\Phi}(i, i_0)e(i_0)$ ⁸, dal quale $E\{e(i)\} = \Phi(i, i_0)E\{e(i_0)\}$, si deduce che $\hat{e}(i) = 0$, $\forall i \geq$

⁸ $\check{\Phi}(i_0, i_0) = I$,
 $\check{\Phi}(i, i_0) = [A(i-1) - K(i-1)C(i-1)][A(i-2) - K(i-2)C(i-2)] \dots [A(i_0) - K(i_0)C(i_0)].$

i_0 .

La minimizzazione della traccia di $\tilde{Q}(i)$ è indipendente dal primo termine ed è sviluppata attraverso una generalizzazione del problema. E' possibile riscrivere la traccia di $\tilde{Q}(i)$ nel modo seguente:

$$\text{tr}[\tilde{Q}(i)] = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \tilde{Q}(i) \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & \cdots & 0 \end{bmatrix} \tilde{Q}(i) \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} + (\cdots). \quad (1.44)$$

Dalla (1.44) si può notare che il problema della minimizzazione della $\text{tr}(\tilde{Q}(i))$ è incluso in quello più generale

$$\min \left(x' \tilde{Q}(i) x \right) \quad \forall x \in \mathbb{R}^n \quad (1.45)$$

Infatti se $\tilde{Q}(i)$ è soluzione del problema posto nella (1.45) allora il prodotto $x' \tilde{Q}(i) x$ sarà minimo per qualsiasi scelta del vettore x , quindi sarà minimo anche per un vettore $x_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}'$ o $x_2 = \begin{bmatrix} 0 & 1 & \cdots & 0 \end{bmatrix}'$ e sarà minima la somma

$$x_1' \tilde{Q}(i) x_1 + x_2' \tilde{Q}(i) x_2 + (\dots) = \text{tr}[\tilde{Q}(i)].$$

Si elabora la matrice $\tilde{Q}(i)$:

$$\begin{aligned} \tilde{Q}(i+1) &= E\{[e(i+1) - \bar{e}(i+1)][e(i+1) - \bar{e}(i+1)]'\} \\ &= E\left\{ \left([A(i) - K(i)C(i)]e(i) + w_1(i) - K(i)w_2(i) - [A(i) - K(i)C(i)]\bar{e}(i) \right) \cdot \right. \\ &\quad \left. \cdot \left([A(i) - K(i)C(i)]e(i) + w_1(i) - K(i)w_2(i) - [A(i) - K(i)C(i)]\bar{e}(i) \right)' \right\} \\ &= E\left\{ \left([A(i) - K(i)C(i)][e(i) - \bar{e}(i)] + w_1(i) - K(i)w_2(i) \right) \cdot \right. \\ &\quad \left. \cdot \left([e(i) - \bar{e}(i)]' [A(i) - K(i)C(i)]' + w_1'(i) - w_2'(i)K'(i) \right) \right\} \\ &= \left([A(i) - K(i)C(i)]E\{[e(i) - \bar{e}(i)][e(i) - \bar{e}(i)]'\}[A(i) - K(i)C(i)]' \right) + \\ &\quad + \left(E\{w_1(i)w_1'(i)\} \right) + \left(K(i)E\{w_2(i)w_2'(i)\}K'(i) \right) - \left(E\{w_1(i)w_2'(i)\}K'(i) \right) + \\ &\quad - \left(K(i)E\{w_2(i)w_1'(i)\} \right) - \left([A(i) - K(i)C(i)]E\{[e(i) - \bar{e}(i)]w_2'(i)\}K'(i) \right) \\ &\quad - \left(K(i)E\{w_2(i)[e(i) - \bar{e}(i)]'\}[A(i) - K(i)C(i)]' \right) + \\ &\quad + \left([A(i) - K(i)C(i)]E\{[e(i) - \bar{e}(i)]w_1'(i)\} \right) + \\ &\quad + \left(E\{w_1(i)[e(i) - \bar{e}(i)]'\}[A(i) - K(i)C(i)]' \right) \\ &= \left([A(i) - K(i)C(i)]\tilde{Q}(i)[A(i) - K(i)C(i)]' \right) + \end{aligned}$$

$$\begin{aligned}
& + \left(V_{11}(i) \right) + \left(K(i)V_{22}(i)K'(i) \right) - \left(V_{12}(i)K'(i) \right) + \\
& - \left(K(i)V_{12}'(i) \right) - \left([A(i) - K(i)C(i)]E\{[e(i) - \bar{e}(i)]w_2'(i)\}K'(i) \right) \\
& - \left(K(i)E\{w_2(i)[e(i) - \bar{e}(i)]'\}[A(i) - K(i)C(i)]' \right) + \\
& + \left([A(i) - K(i)C(i)]E\{[e(i) - \bar{e}(i)]w_1'(i)\} \right) - \\
& + \left(E\{w_1(i)[e(i) - \bar{e}(i)]'\}[A(i) - K(i)C(i)]' \right) \\
= & [A(i) - K(i)C(i)]\tilde{Q}(i)[A(i) - K(i)C(i)]' + \\
& + V_{11}(i) - V_{12}(i)K'(i) - K(i)V_{12}'(i) + K(i)V_{22}(i)K'(i)
\end{aligned}$$

dove l'ultima uguaglianza trova giustificazione:

$$\begin{aligned}
E\{[e(i) - \bar{e}(i)]w_2'(i)\} &= E\{[e(i) - \bar{e}(i)]w_2'(i)\} = E\{[x(i) - \hat{x}(i) - \bar{x}(i) + \tilde{x}(i)]w_2'(i)\} \\
&= E\{[x(i) - \bar{x}(i)]w_2'(i)\} - E\{[\hat{x}(i) - \tilde{x}(i)]w_2'(i)\} = 0, \quad (1.46)
\end{aligned}$$

e su quest'ultima uguaglianza può essere ripetuto il ragionamento sviluppato nella dimostrazione del teorema (2), ricordando che $E\{[x(i_0) - \bar{x}_0]w_2'\} = 0$.

Allo stesso modo

$$\begin{aligned}
E\{[e(i) - \bar{e}(i)]w_1'(i)\} &= E\{[e(i) - \bar{e}(i)]w_1'(i)\} = E\{[x(i) - \hat{x}(i) - \bar{x}(i) + \tilde{x}(i)]w_1'(i)\} \\
&= E\{[x(i) - \bar{x}(i)]w_1'(i)\} - E\{[\hat{x}(i) - \tilde{x}(i)]w_1'(i)\} = 0, \quad (1.47)
\end{aligned}$$

Si è così ricavata una formulazione ricorsiva di $\tilde{Q}(i)$ che ne permette il calcolo una volta nota la $\tilde{Q}(i_0)$:

$$\begin{aligned}
\tilde{Q}(i+1) &= [A(i) - K(i)C(i)]\tilde{Q}(i)[A(i) - K(i)C(i)]' + \\
& + V_{11}(i) - V_{12}(i)K'(i) - K(i)V_{12}'(i) + K(i)V_{22}(i)K'(i) \quad (1.48)
\end{aligned}$$

$$\tilde{Q}(i_0) = Q_0 \quad (1.49)$$

nella quale la (1.49) è giustificata da:

$$\begin{aligned}
\tilde{Q}(i_0) &= E\left\{[e(i_0) - \bar{e}(i_0)][e(i_0) - \bar{e}(i_0)]'\right\} \\
&= E\left\{[(x(i_0) - \bar{x}(i_0)) - (\hat{x}(i_0) - \tilde{x}(i_0))][(x(i_0) - \bar{x}(i_0)) - (\hat{x}(i_0) - \tilde{x}(i_0))']\right\}, \\
&= E\left\{[x(i_0) - \bar{x}(i_0)][x(i_0) - \bar{x}(i_0)]'\right\} = Q_0
\end{aligned}$$

La (1.48) è equivalente a:

$$\begin{aligned}
\tilde{Q}(i+1) &= [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}](V_{22} + CQC')^{-1} \cdot \\
& \cdot [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}]' + \\
& - (AQC' + V_{12})(V_{22} + CQC')^{-1}(CQA' + V_{12}') + \\
& + AQA' + V_{11} \quad (1.50)
\end{aligned}$$

dove si è omessa la dipendenza da i per semplificare la notazione. $(V_{22} + CQC')$ è sempre invertibile in quanto la somma di una matrice definita positiva e di una semidefinita positiva dá luogo ad una matrice definita positiva. Una matrice definita positiva ha tutti autovalori non nulli, ovvero determinante diverso da 0: è sempre invertibile.

Supponendo che in i la matrice $\tilde{Q}(i)$ sia ottima, scegliendo

$$K^o(i) = (A(i)Q(i)C'(i) + V_{12}(i))(V_{22}(i) + C(i)Q(i)C'(i))^{-1} \quad (1.51)$$

si ottiene la matrice $\tilde{Q}(i+1)$ ottima all'istante $i+1$.

dimostrazione

$$\begin{aligned} x' \tilde{Q}(i+1)x &= x' [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}] (V_{22} + CQC')^{-1} \cdot \\ &\quad \cdot [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}]' x + \\ &\quad - x' (AQC' + V_{12})(V_{22} + CQC')^{-1} (CQA' + V_{12}') x + \\ &\quad + x' AQA' x + x' V_{11} x \end{aligned} \quad (1.52)$$

definendo

$$\begin{aligned} \gamma_1' &\triangleq x' [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}] \\ \gamma_2' &\triangleq x' (AQC' + V_{12}) \\ \gamma_3' &\triangleq x' A \\ \gamma_4 &\triangleq x' V_{11} x \end{aligned}$$

si possono riscrivere le (1.52)

$$x' \tilde{Q}(i+1)x = \gamma_1' (V_{22} + CQC^{-1}) \gamma_1 - \gamma_2' (V_{22} + CQC')^{-1} \gamma_2 + \gamma_3' Q \gamma_3 + \gamma_4.$$

La scelta di K modifica unicamente il primo elemento del secondo membro. La matrice $(V_{22} + CQC^{-1})$ è definita positiva ed è pre e post moltiplicata per lo stesso vettore pertanto il prodotto $\gamma_1' (V_{22} + CQC^{-1}) \gamma_1$ è necessariamente positivo. Il minimo si ottiene determinando il γ_1 per il quale $\gamma_1' (V_{22} + CQC^{-1}) \gamma_1 = 0$.

$$\begin{aligned} \gamma_1' (V_{22} + CQC^{-1}) \gamma_1 &\geq 0 \\ &\Rightarrow \\ \gamma_1' &= 0 = x' [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}] \end{aligned} \quad (1.53)$$

La (1.53) deve valere $\forall x$

$\Rightarrow$

$$\begin{aligned} [K - (AQC' + V_{12})(V_{22} + CQC')^{-1}] &= 0 \\ K(i) &= (A(i)Q(i)C'(i) + V_{12}(i))(V_{22}(i) + C(i)Q(i)C'(i))^{-1} \end{aligned}$$

■

Come precedentemente sviluppato l'ottimalità della $\tilde{Q}(i+1)$ rispetto alla $\tilde{Q}(i)$ definisce l'ottimalità di $\text{tr}[\tilde{Q}(i+1)]$ rispetto $\text{tr}[\tilde{Q}(i)]$ ovvero la minimizzazione dell'incremento dell'errore quadratico medio nella stima tra l'istante i e quello $i+1$.

La formulazione ricorsiva ottima di $\tilde{Q}(i)$ consente di affermare che, noto $\tilde{Q}(i_0)$, se il sistema d'errore utilizza matrice dei guadagni $K^o(i)$ data dall'equazione (1.51), allora l'errore quadratico medio sulla stima è minimo. Per ogni passo i infatti, grazie ad una scelta opportuna di $K^o(i)$, si ha matrice $\tilde{Q}(i+1)$ ottima.

Teorema 5 (filtro di Kalman) *Per il sistema descritto dalle (1.36), la matrice $K^0(i)$, $i \geq i_0$, dei guadagni ottimi dell'osservatore è ottenuta dalla formulazione ricorsiva*

$$K^o(i) = [A(i)Q(i)C'(i) + V_{12}(i)][V_{22}(i) + C(i)Q(i)C'(i)]^{-1}$$

con $\tilde{Q}(i)$ calcolata

$$\tilde{Q}(i+1) = -(AQC' + V_{12})(V_{22} + CQC')^{-1}(CQA' + V'_{12}) + AQA' + V_{11}$$

⁹ per $i \geq i_0$, con condizioni iniziali

$$\begin{aligned}\tilde{Q}(i_0) &= Q_0 \\ \hat{x}(i_0) &= \bar{x}_0\end{aligned}\tag{1.54}$$

⁹Nell'ultima equazione si è omessa la dipendenza dal tempo per brevità di notazione

Capitolo 2

Fusione sensoriale con caratterizzazioni statistiche note

2.1 Relazioni di misura lineari

Il problema che si vuole affrontare consiste nella determinazione di un parametro $x \in \mathbb{R}^n$ non noto, attraverso un insieme di misure $y_i \in \mathbb{R}^p$ con $i = 1, \dots, N^1$, generalmente rumorose. La relazione che lega il parametro alle misure può essere esplicitata:

$$y_i = C_i x + u_i \quad (2.1)$$

con u_i rumore di misura, $E\{u_i\} = 0$, $E\{u_i u_i'\} = U_i \geq 0$, $E\{u_i u_j'\} = 0 \forall i \neq j$.

Riscrivendo la relazione di misura (2.1) come un sistema dinamico a tempo discreto, lineare, non stazionario, si può costruire un osservatore dello stato che stimi il parametro cercato. Utilizzando poi il filtro di Kalman, si ottiene una stima con errore quadratico medio minimo:

$$\begin{aligned} x_{i+1} &= x_i \\ y_i &= C_i x_i + u_i \end{aligned} \quad (2.2)$$

del quale si può definire un osservatore

$$\hat{x}_i = \hat{x}_{i-1} + K_i(y_i - C_i \hat{x}_{i-1}) \quad (2.3)$$

La differenza $(y_i - C_i \hat{x}_{i-1})$ corregge la stima del parametro effettuata al passo immediatamente precedente. L'affidabilità del fattore correttivo è fortemente vincolata alla qualità delle misure. L'impiego dell'algoritmo di Kalman consente di

¹ $n \in \mathbb{N}$, $p \in \mathbb{N}$, $N \in \mathbb{N}$

minimizzare l'entità degli errori sulla stima finale del parametro, mantenendo i vantaggi dell'impiego di più sensori. La nuova stima consisterà in un aggiornamento della vecchia stima basato sulla differenza tra l'ipotetica misura dei sensori, $C_i \hat{x}_{i-1}$, e la misura reale y_i

Considerando l'osservatore (2.3) e introducendo il sistema d'errore

$$\begin{aligned} e_i &= x - \hat{x}_i = x - \hat{x}_{i-1} + K_i(y_i - C_i \hat{x}_{i-1}) \\ &= (I - K_i C_i) e_{i-1} + K_i u_i \end{aligned} \quad (2.4)$$

con relativa matrice di varianza:

$$\begin{aligned} S_i &\triangleq E\{[e_i - \bar{e}_i][e_i - \bar{e}_i]'\} = E\{e_i e_i'\} = \\ &= E\{[x - \hat{x}_i][x - \hat{x}_i]'\}. \\ \bar{e}_i &= E\{e_i\}, \end{aligned}$$

si vogliono scegliere i guadagni K_i affinché sia minimo l'errore quadratico medio $E\{e_i' e_i\}$. La matrice di varianza può essere sviluppata nel seguente modo²:

$$\begin{aligned} S_i &= E\{e_i e_i'\} = E\{[(I - K_i C_i) e_{i-1} + K_i u_i][(I - K_i C_i) e_{i-1} + K_i u_i]'\} \\ &= (I - K_i C_i) E\{e_{i-1} e_{i-1}'\} (I - K_i C_i)' + K_i E\{u_i u_i'\} K_i' \\ &= (I - K_i C_i) S_{i-1} (I - K_i C_i)^{-1} + K_i U_i K_i' \end{aligned} \quad (2.5)$$

dove l'ultima uguaglianza è giustificata dall'ipotesi di incorrelazione tra lo stato del sistema e il disturbo aleatorio sulla misura³. Si può riscrivere (completamento ai quadrati):

$$\begin{aligned} S_i &= [K_i - (S_{i-1} C_i')(U_i + C_i S_{i-1} C_i')^{-1}](U_i + C_i S_{i-1} C_i') \cdot \\ &\quad \cdot [K_i - (S_{i-1} C_i')(U_i + C_i S_{i-1} C_i')^{-1}]' + \\ &\quad - (S_{i-1} C_i')(U_i + C_i S_{i-1} C_i')(C_i S_{i-1}) + S_{i-1} \end{aligned} \quad (2.6)$$

Da questa nuova formulazione rimangono definiti i guadagni K_i^o

$$K_i^o \triangleq (S_{i-1} C_i')(U_i + C_i S_{i-1} C_i')^{-1} \quad (2.7)$$

la cui ottimalità è stata trattata nel precedente capitolo. Da cui:

$$\begin{aligned} S_i &= -(S_{i-1} C_i')(U_i + C_i S_{i-1} C_i')(C_i S_{i-1}) + S_{i-1} \\ &= -K_i^o C_i S_{i-1} + S_{i-1} \\ &= [I - K_i^o C_i] S_{i-1} \end{aligned} \quad (2.8)$$

In conclusione un osservatore dello stato costituito dalle equazioni

$$\begin{aligned} \hat{x}_i &= \hat{x}_{i-1} + K_i^o (y_i - C_i \hat{x}_{i-1}) \\ K_i^o &= (S_{i-1} C_i')(U_i + C_i S_{i-1} C_i')^{-1} \\ S_i &= [I - K_i^o C_i] S_{i-1} \end{aligned} \quad (2.9)$$

²La procedura dettagliata per determinare i guadagni ottimi è stata sviluppata nel precedente capitolo. Si riportano i risultati principali adattati ai sistemi trattati

³Anche questa ipotesi è stata utilizzata nel corso del precedente capitolo

minimizza l'errore quadratico medio $E\{[x_i - \hat{x}_i]'[x_i - \hat{x}_i]\}$.

2.2 Relazioni di misura non lineari

Nel caso più generale la relazione che lega la misura al parametro non è esplicita come nel caso precedente.

- $\hat{z}_i \in \mathbb{R}^m$, misure dirette delle quantità z_i
- $f_i(z_i, x) = 0$, $i = 1, \dots, N$, insieme di relazioni, in generale non lineari e non esplicite, che legano le quantità z_i al parametro x da stimare. $f_i : \mathbb{R}^m \times \mathbb{R}^n \rightarrow \mathbb{R}^p$
- $\hat{z}_i = z_i + w_i$, misure affette da rumore, w_i è caratterizzato da:

$$\begin{aligned} E[w_i] &= 0 \\ E[w_i w_i'] &= W_i \geq 0 \\ E[w_i w_j'] &= 0, \quad i \neq j \end{aligned}$$

Si vogliono elaborare le relazioni non lineari, al fine di ottenere una forma strutturalmente simile al precedente paragrafo, per riuscire a determinare un metodo ricorsivo di stima.

Nota la stima $\hat{x}_{i-1}$ e lo stato iniziale $\hat{x}_0$, si sviluppa in serie di Taylor, arrestato al primo ordine, centrato su $(\hat{z}_i, \hat{x}_{i-1})$:

$$f_i(z_i, x) = 0 \cong f_i(\hat{z}_i, \hat{x}_{i-1}) + \frac{\partial f_i}{\partial z} \big|_{(\hat{z}_i, \hat{x}_{i-1})} (z_i - \hat{z}_i) + \frac{\partial f_i}{\partial x} \big|_{(\hat{z}_i, \hat{x}_{i-1})} (x - \hat{x}_{i-1}).$$

e fatte le posizioni:

$$u_i \triangleq \frac{\partial f_i}{\partial z} \big|_{(\hat{z}_i, \hat{x}_{i-1})} (z_i - \hat{z}_i) \quad (2.10)$$

$$C_i \triangleq \frac{\partial f_i}{\partial x} \big|_{(\hat{z}_i, \hat{x}_{i-1})} \quad (2.11)$$

si ottiene⁴:

$$\begin{aligned} 0 &= f_i(\hat{z}_i, \hat{x}_{i-1}) + u_i + C_i(x - \hat{x}_{i-1}) \\ C_i x + u_i &= -f_i(\hat{z}_i, \hat{x}_{i-1}) + C_i \hat{x}_{i-1} \end{aligned} \quad (2.12)$$

⁴Considerando le nonlinearità della relazione come un errore di misura u_i .

il secondo membro della (2.12) è calcolabile una volta effettuate le misure e può essere considerato come una misura diretta (y_i). Si è ottenuta nuovamente la forma

$$y_i = C_i x + u_i \quad (2.13)$$

ed è possibile applicare l'osservatore precedentemente descritto.

Il rumore u_i ha caratteristiche dipendenti da w_i in quanto la differenza ($z_i - \hat{z}_i$) dipende unicamente da questo rumore.

$$\begin{aligned} E[u_i] &= E \left[\frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} (z_i - \hat{z}_i) \right] \\ &= \frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} E[w_i] = 0 \\ E[u_i u_i'] &= E \left[\left(\frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} (z_i - \hat{z}_i) \right) \left(\frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} (z_i - \hat{z}_i) \right)' \right] \\ &= \left(\frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} \right) E[w_i w_i'] \left(\frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} \right)' \\ &= \frac{\partial f_i}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} W_i \frac{\partial f_i'}{\partial z} \Big|_{(\hat{z}_i, \hat{x}_{i-1})} = U_i \end{aligned} \quad (2.14)$$

Capitolo 3

Localizzazione robotica

3.1 Il carattere locale del problema

La posizione del robot può essere stimata prelevando informazioni dall'ambiente circostante: le informazioni necessarie alla stima differiscono in base al tipo di movimentazione e ambiente operativo, al livello di precisione necessario e alla qualità dei sensori utilizzati. E' difficile ipotizzare una soluzione generale del problema che vada oltre la semplice intenzione di elaborare i dati sfruttandone "ridondanza" e "complementarità". Quanto esposto di seguito perde pertanto ogni, seppur minima, pretesa di generalità, cercando di ottenere una soluzione valida per una specifica architettura hardware.



Figura 3.1: Nomad 150

Lo studio è stato sviluppato sul robot riportato in figura 3.1. La movimentazione è garantita da tre ruote in grado di operare simultaneamente cambi di direzione

(rotazione sull'asse perpendicolare al terreno). Il corpo del robot non è vincolato alla base e può ruotare sull'asse centrale. Il robot acquisisce informazioni dall'ambiente mediante encoder sulle ruote e sensori acustici posti sulla circonferenza (16 sonar). La soluzione del problema utilizza due tipi di rilevamenti: odometrici ed acustici.

3.2 Modello cinematico e rilevamenti odometrici

La soluzione più semplice per la stima di posizione consiste nell'integrazione diretta nel tempo delle informazioni di velocità. Supponendo un ambiente bidimensionale, si può definire la configurazione del robot rispetto ad un riferimento fisso solidale con l'ambiente, mediante un vettore di tre parametri.

$$X_r = \begin{bmatrix} x & y & \theta \end{bmatrix}' \quad (3.1)$$

Nell'ipotesi di perfetta aderenza delle ruote al suolo, esatta calibrazione delle ruote, misure non rumorose (etc.), ovvero in assenza di errori odometrici, possono essere usate le equazioni:

$$\begin{aligned} \dot{x} &= v(t) \cos \theta \\ \dot{y} &= v(t) \sin \theta \\ \dot{\theta} &= \omega(t) \end{aligned} \quad (3.2)$$

dove $v(t)$ e $\omega(t)$ sono le velocità di traslazione e rotazione del robot misurate. Scegliendo opportunamente il tempo di campionamento T , è possibile discretizzare le (3.2). Operando su una sola equazione

$$\begin{aligned} \frac{x((k+1)T) - x(kT)}{T} &= \nu(kT) \cos \left(\theta(kT) + \frac{\omega(kT)T}{2} \right) \\ \Rightarrow \\ x((k+1)T) &= x(kT) + T\nu(kT) \cos \left(\theta(kT) + \frac{\omega(kT)T}{2} \right) \\ \Rightarrow \\ x(k+1) &= x(k) + T\nu(k) \cos \left(\theta(k) + \frac{\omega(k)T}{2} \right) \end{aligned}$$

ed estendendo il risultato a tutte le equazioni, si ottiene

$$\begin{aligned} x(k+1) &= x(k) + T\nu(k) \cos \left(\theta(k) + \frac{\omega(k)T}{2} \right) \\ y(k+1) &= y(k) + T\nu(k) \sin \left(\theta(k) + \frac{\omega(k)T}{2} \right) \\ \theta(k+1) &= \theta(k) + T\omega(k) \end{aligned} \quad (3.3)$$

Il moto del robot presenta continui slittamenti, piccoli urti, discontinuità del suolo, che portano rapidamente l'integrazione diretta dei dati odometrici a risultati erronei: si verifica un disallineamento tra la posizione effettiva e quella misurata. Il modello descritto dalle (3.3) non è sufficiente e deve essere completato con l'inserimento di disturbi aleatori:

$$\begin{aligned}x(k+1) &= x(k) + Tv(k) \cos\left(\theta(k) + \frac{\omega(k)T}{2}\right) + v_x(k+1) \\y(k+1) &= y(k) + Tv(k) \sin\left(\theta(k) + \frac{\omega(k)T}{2}\right) + v_y(k+1) \\\theta(k+1) &= \theta(k) + T\omega(k) + v_\theta(k+1)\end{aligned}\tag{3.4}$$

con $v(k) = [v_x \ v_y \ v_\theta]'$ disturbi gaussiani a media nulla, esplicitamente:

$$\begin{aligned}E\{v(k)\} &= 0 \\E\{v(k)v'(h)\} &= Q(k, \Delta s) \quad h = k \\E\{v(k)v'(h)\} &= 0 \quad h \neq k\end{aligned}$$

con $Q(k, \Delta s)$, simmetrica e semidefinita positiva, dipendente dalla distanza (Δs) tra due step consecutivi¹.

Per migliorare l'accuratezza della stima, e risolvere la sostanziale instabilità del metodo di integrazione diretta, sono necessarie altre misure. Dato un generico vettore $y(k)$ di misure al tempo k si vuole un algoritmo di stima che minimizzi la differenza tra posizione reale e stimata.

$$\hat{X}_r(k+1) = \Lambda(\hat{X}_r(k), y(k))\tag{3.5}$$

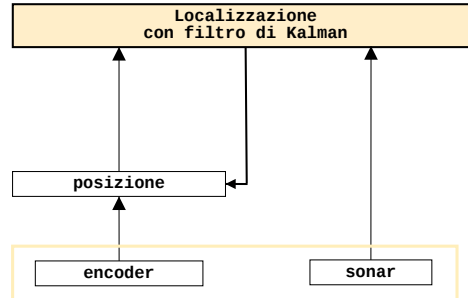


Figura 3.2: Ipotesi sull'algoritmo di stima

¹E' consuetudine ipotizzare una matrice diagonale: i primi due elementi della diagonale costituiscono l'incertezza sulla stima di posizione, il terzo elemento esprime l'incertezza sull'orientamento

3.3 Definizione del problema

I limiti delle informazioni odometriche possono essere superati quando si considerino le informazioni complementari fornite da altri sensori. Nel caso specifico sono presenti 16 sensori sonar gestibili mediante un vettore dati $Y(k) = [y_1, \dots, y_{16}]'$. Le misure sonar sono dipendenti dall'ambiente di lavoro e dalla posizione del robot:

$$Y(k) = h(k, X_r) + \omega(k) \quad (3.6)$$

dove con $h(k, X_r)$ si indica la relazione esistente tra la posizione del robot e il rilevamento dati sonar; $\omega(k)$ è un rumore gaussiano a media nulla, con matrice di covarianza diagonale, $R(k)$, utilizzato per modellare gli errori di misura.

Considerando il legame tra “ambiente, posizione robot” e “rilevamento sonar” è possibile porre formalmente il problema della localizzazione come la ricerca di una funzione Λ che fornisca in modo ricorsivo nel tempo la configurazione ottima del robot, combinando le equazioni (3.4) del modello del sistema e le informazioni ultrasoniche (3.6):

$$X_r(k+1) = \Lambda(X_r(k), Y(k+1)) \quad (3.7)$$

Ad ogni ciclo di lavoro l'algoritmo aggiorna la stima di posizione $X_r(k)$ e la matrice di covarianza dell'errore associato nelle corrispondenti variabili al passo successivo, $X_r(k+1)$ e relativa covarianza. L'algoritmo utilizzerà un filtro di Kalman esteso (EKF).

3.4 Filtro di Kalman Esteso

L'ambiente nel quale avviene la movimentazione del robot è descritto da una mappa planare costituita da segmenti rappresentanti le pareti principali (con coordinate relative al sistema di riferimento fisso). Il robot è in grado di misurare la velocità delle ruote e la variazione di orientamento ed utilizza un sistema di rilevatori ultrasonici (sonar) per valutare la distanza dalle pareti.

L'idea principale per risolvere il problema della localizzazione è la seguente: noto lo stato $X_r(k)$ del robot e date le misure di velocità lineare ed angolare si può ipotizzare la nuova posizione $X_{r,ipotesi}(k+1)$. Nella nuova posizione ipotetica possono essere calcolate le risposte dei sonar. Quando il robot andrà nello stato $X_r(k+1)$ si potranno confrontare le risposte calcolate sullo stato ipotetico, con le misure effettive dei sonar e correggere eventuali errori di posizione.

In questo modo si confronta il risultato dell'integrazione diretta delle velocità con le misure dei sensori, riducendo continuamente potenziali errori di stima.

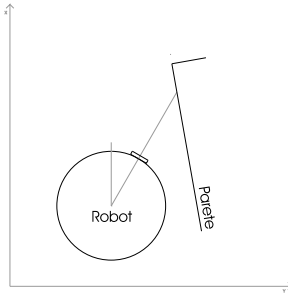


Figura 3.3: Relazione del robot con l'ambiente

Si esplicita la relazione (3.6):

$$h(k, X_r) = \begin{bmatrix} h_1(k, X_r) \\ \vdots \\ h_{16}(k, X_r) \end{bmatrix} \quad (3.8)$$

$$h_i(k, X_r) = \sqrt{(S_{i,x} - x_p)^2 + (S_{i,y} - y_p)^2} \quad (3.9)$$

La relazione (3.9) è la spiegazione dell'origine della misura sonar e definisce il rapporto intercorrente tra questa e la posizione del robot. Con S_i si indica la posizione del sensore i -esimo (nelle sue coordinate assolute x e y) che dipende dalla posizione e dall'orientamento del robot; (x_p, y_p) è la posizione di un generico punto della parete, individuato dalla retta ortogonale alla tangente della circonferenza nel punto di collocazione del sonar e la parete stessa. Si è omessa la dipendenza dal tempo k per semplicità di notazione.

Definizione 5

$$\hat{x}_r(k | k) \triangleq \begin{bmatrix} \hat{x}(k | k) & \hat{y}(k | k) & \hat{\theta}(k | k) \end{bmatrix}'$$

denota la stima dello stato del robot al tempo k basato sui primi k vettori di misure dei sonar.

Data la definizione (5), la soluzione del problema può essere riscritta:

$$\hat{X}_r(k+1|k+1) = \Lambda(\hat{X}_r(k|k), Y(k+1)). \quad (3.10)$$

Ad ogni passo l'algoritmo stima in funzione dei rilevamenti sperimentali, lo stato del robot e la relativa matrice di covarianza². L'uso del filtro di Kalman nella progettazione dell'algoritmo permette di ridurre l'incertezza sulla stima dello stato dovuta alla presenza di rumore nelle misure e alla movimentazione del robot.

²Indica il grado di accuratezza della stima

3.4.1 Fase di predizione (prediction phase)

Si consideri noto lo stato $\hat{X}_r(k|k)$ (stimato al tempo k , dopo k misure sonar) e la matrice di covarianza

$$P(k|k) = E\{[\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)][\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)]'\} \quad (3.11)$$

si può fare una predizione al tempo $k+1$

$$\begin{aligned} \hat{x}(k+1|k) &= \hat{x}(k|k) + T\nu(k) \cos\left(\hat{\theta}(k|k) + \frac{\omega(k)T}{2}\right) \\ \hat{y}(k+1|k) &= \hat{y}(k|k) + T\nu(k) \sin\left(\hat{\theta}(k|k) + \frac{\omega(k)T}{2}\right) \\ \hat{\theta}(k+1|k) &= \hat{\theta}(k|k) + T\omega(k) \end{aligned} \quad (3.12)$$

$$P(k+1|k) = E\{[\hat{x}_r(k+1|k) - \bar{\hat{x}}_r(k+1|k)][\hat{x}_r(k+1|k) - \bar{\hat{x}}_r(k+1|k)]'\} \quad (3.13)$$

Lo stato $X_r(k+1|k)$ è facilmente calcolabile mentre la relazione (3.13) deve essere semplificata. Si linearizzano le equazioni (3.3) sul punto $X = \hat{x}_r(k|k)$:

$$\begin{aligned} f(k, X) &= \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} x(k) + Tv(k) \cos\left(\theta(k) + \frac{\omega(k)T}{2}\right) \\ y(k) + Tv(k) \sin\left(\theta(k) + \frac{\omega(k)T}{2}\right) \\ \theta(k) + T\omega(k) \end{bmatrix} \\ &\Rightarrow \\ F(k) &\triangleq \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial \theta} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial \theta} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial \theta} \end{bmatrix}_{x_r=\hat{x}_r(k|k)} \\ &= \begin{bmatrix} 1 & 0 & -Tv(k) \sin\left(\theta(k) + \frac{\omega(k)T}{2}\right) \\ 0 & 1 & Tv(k) \cos\left(\theta(k) + \frac{\omega(k)T}{2}\right) \\ 0 & 0 & 1 \end{bmatrix}_{X_r=\hat{X}_r(k|k)} \end{aligned} \quad (3.14)$$

Note le velocità di traslazione e rotazione del robot al tempo k (rilevamenti odometrici), $F(k)$ rimane definita. Da cui

$$\begin{aligned} P(k+1|k) &= E\{[x_r(k+1|k) - \bar{x}_r(k+1|k)][x_r(k+1|k) - \bar{x}_r(k+1|k)]'\} \\ &= E\{[(F(k)\hat{x}_r(k|k) + v_{x_r}(k)) - (F(k)\bar{\hat{x}}_r(k|k))] \cdot \\ &\quad \cdot [(F(k)\hat{x}_r(k|k) + v_{x_r}(k)) - (F(k)\bar{\hat{x}}_r(k|k))]' \} \\ &= F(k)E\{[\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)][\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)]'\}F'(k) + \\ &\quad + E\{v_{x_r}(k)v_{x_r}'(k)\} - F(k)E\{[\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)][v_{x_r}'(k)]\} + \\ &\quad + E\{[v_{x_r}(k)][\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)]'\}F'(k) \\ &= F(k)E\{[\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)][\hat{x}_r(k|k) - \bar{\hat{x}}_r(k|k)]'\}F'(k) + \end{aligned} \quad (3.15)$$

$$+E\{v_{x_r}(k)v'_{x_r}(k)\} \quad (3.16)$$

$$= F(k)P(k|k)F'(k) + Q(k, l) \quad (3.17)$$

$$\text{con } Q(k, l) \triangleq E\{v_{x_r}(k)v'_{x_r}(k)\}$$

$$\text{e } \hat{\tilde{x}} = E[\hat{x}]$$

dove l'uguaglianza tra la (3.15) e la (3.16) è vera nell'ipotesi di incorrelazione tra gli errore odometrici nello stato e lo stato stesso; la matrice $Q(k, l)$, simmetrica e semidefinita positiva, sarà dipendente dalla distanza percorsa, l .

Le equazioni (3.12) e la (3.17) calcolano la stima dello stato e della matrice di covarianza al tempo $k + 1$ basandosi sulle informazioni note al tempo k . I risultati trovati possono essere utilizzati per effettuare la predizione sulle misure dei sonar.

$$\hat{Y}(k + 1) = h(k, \hat{x}_r(k + 1|k)).$$

Questo risultato, nella seconda parte dell'algoritmo, sarà confrontato con le misure reali dei sensori per ottenere un miglioramento della stima di posizione.

3.4.2 Predizione del valore misurato dai sonar

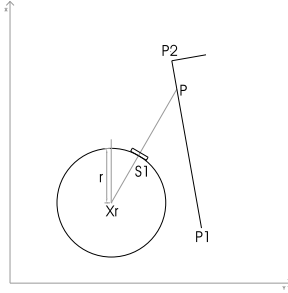


Figura 3.4: Interazione sonar – ambiente

Noto lo stato del robot X_r ³ si calcola posizione e orientamento dell'i-esimo sensore. Considerando la figura 3.4:

$$\begin{aligned} S_{i,x} &= x + r \cos(\theta_0 + \theta) \\ S_{i,y} &= y + r \sin(\theta_0 + \theta) \\ S_{i,\theta} &= \theta_0 + \theta \end{aligned} \quad (3.18)$$

con θ_0 angolo sensore rispetto all'orientamento $\theta = 0$ del robot.

Definita la posizione e l'orientamento del sensore è importante capire quale sarà il punto sulla parete colpito. La parete è descritta mediante i punti estremi

³Si omette la dipendenza dal tempo k per semplicità di notazione

P_1 e P_2 . Il segmento P_1P_2 è approssimabile, a meno dei suoi limiti, da una retta e risolvendo nelle incognite m_p, q_p

$$\begin{aligned} P_{1,y} &= m_p P_{1,x} + q_p \\ P_{2,y} &= m_p P_{2,x} + q_p \end{aligned}$$

si ottengono i parametri della retta cercata.

Il sensore misura la distanza ortogonale al suo orientamento. Anche in questo caso è possibile approssimare la direzione di misura con una retta

$$S_{i,y} = \tan(S_{i,\theta}) S_{i,x} + q_s$$

che deve essere risolta nell'incognita q_s .

Le due rette determinate si intersecheranno nel punto P cercato. Si deve risolvere nelle incognite x_p, y_p

$$\begin{aligned} y_p &= m_p x_p + q_p \\ y_p &= \tan(S_{i,\theta}) x_p + q_s \end{aligned}$$

ottenendo le coordinate del punto sulla parete.

La predizione dell' i -esimo sensore sarà quindi:

$$h_i(k, x_r) = \sqrt{(S_{i,x} - x_p)^2 + (S_{i,y} - y_p)^2}.$$

3.4.3 L'algoritmo: fase di stima (estimation phase)

La fase di stima dell'algoritmo inizia quando sono note le misure dei sonar al tempo $k + 1$. Si definisce

$$\tilde{Y}(k+1) \triangleq Y(k+1) - \hat{Y}(k+1) \quad (3.19)$$

e la fase di stima consiste nella sua elaborazione volta all'aggiornamento della posizione del robot, $X_r(k+1|k+1)$. Come per lo stato X_r , si può definire una matrice di covarianza:

$$\begin{aligned} S(k+1) &\triangleq E\{\tilde{y}(k+1)\tilde{y}'(k+1)\} \\ &= E\{[y(k+1) - \hat{y}(k+1)][y(k+1) - \hat{y}(k+1)]'\} \end{aligned}$$

della quale si deve trovare una forma più semplice per il calcolo. I passaggi sono simili al caso precedente. Si linearizza la relazione $h(k, x_r)$

$$h(k, x_r) = \begin{bmatrix} h_1(k, x_r) \\ h_2(k, x_r) \\ \dots \\ h_N(k, x_r) \end{bmatrix} = \begin{bmatrix} \left((S_{1,x} - x_1)^2 + (S_{1,y} - y_1)^2 \right)^{\frac{1}{2}} \\ \left((S_{2,x} - x_2)^2 + (S_{2,y} - y_2)^2 \right)^{\frac{1}{2}} \\ \dots \\ \left((S_{N,x} - x_N)^2 + (S_{N,y} - y_N)^2 \right)^{\frac{1}{2}} \end{bmatrix}. \quad (3.20)$$

$$= \begin{bmatrix} \left((x_{r,x} + r \cos(\theta_1 + x_{r,\theta}) - x_1)^2 + (x_{r,y} + r \sin(\theta_1 + x_{r,\theta}) - y_1)^2 \right)^{\frac{1}{2}} \\ \left((x_{r,x} + r \cos(\theta_2 + x_{r,\theta}) - x_2)^2 + (x_{r,y} + r \sin(\theta_2 + x_{r,\theta}) - y_2)^2 \right)^{\frac{1}{2}} \\ (...) \\ \left((x_{r,x} + r \cos(\theta_N + x_{r,\theta}) - x_N)^2 + (x_{r,y} + r \sin(\theta_N + x_{r,\theta}) - y_N)^2 \right)^{\frac{1}{2}} \end{bmatrix}.$$

Le righe sono tutte simili tra loro:

$$\begin{aligned} \frac{\partial h_i}{\partial x_{r,x}} &= \left((x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i)^2 + (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i)^2 \right)^{-1/2} \cdot \\ &= \cdot \left((x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i) \left(1 - \frac{\partial x_i}{\partial x_{r,x}} \right) + (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i) \left(-\frac{\partial y_i}{\partial x_{r,x}} \right) \right) \\ \frac{\partial h_i}{\partial x_{r,y}} &= \left((x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i)^2 + (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i)^2 \right)^{-1/2} \cdot \\ &\cdot \left((x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i) \left(-\frac{\partial x_i}{\partial x_{r,y}} \right) + (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i) \left(1 - \frac{\partial y_i}{\partial x_{r,y}} \right) \right) \\ \frac{\partial h_i}{\partial x_{r,\theta}} &= \left((x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i)^2 + (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i)^2 \right)^{-1/2} \cdot \\ &\cdot \left[(x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i) \cdot \frac{\partial}{\partial x_{r,\theta}} (r \cos(\theta_i + x_{r,\theta}) - x_i) + \right. \\ &\left. + (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i) \cdot \frac{\partial}{\partial x_{r,\theta}} (r \sin(\theta_i + x_{r,\theta}) - y_i) \right] \end{aligned} \quad (3.21)$$

Chiamando con d_i , per semplificare la notazione dei passaggi, la distanza misurata dall' i -esimo sensore ovvero

$$d_i \triangleq \left((x_{r,x} + r \cos(\theta_i + \theta) - x_i)^2 + (x_{r,y} + r \sin(\theta_i + \theta) - y_i)^2 \right)^{\frac{1}{2}}$$

e considerando che

$$\begin{aligned} (x_{r,x} + r \cos(\theta_i + x_{r,\theta}) - x_i) &= d_i \cos(\theta_i + x_{r,\theta}) \\ (x_{r,y} + r \sin(\theta_i + x_{r,\theta}) - y_i) &= d_i \sin(\theta_i + x_{r,\theta}) \end{aligned}$$

si possono riscrivere le (3.21):

$$\begin{aligned} \frac{\partial h_i}{\partial x_{r,x}} &= \cos(\theta_i + x_{r,\theta}) \left(1 - \frac{\partial x_i}{\partial x_{r,x}} \right) + \sin(\theta_i + x_{r,\theta}) \left(-\frac{\partial y_i}{\partial x_{r,x}} \right) \\ \frac{\partial h_i}{\partial x_{r,y}} &= \cos(\theta_i + x_{r,\theta}) \left(-\frac{\partial x_i}{\partial x_{r,y}} \right) + \sin(\theta_i + x_{r,\theta}) \left(1 - \frac{\partial y_i}{\partial x_{r,y}} \right) \\ \frac{\partial h_i}{\partial x_{r,\theta}} &= \cos(\theta_i + x_{r,\theta}) \frac{\partial}{\partial x_{r,\theta}} (r \cos(\theta_i + x_{r,\theta}) - x_i) + \\ &\quad + \sin(\theta_i + x_{r,\theta}) \frac{\partial}{\partial x_{r,\theta}} (r \sin(\theta_i + x_{r,\theta}) - y_i) \end{aligned} \quad (3.22)$$

Si devono svolgere ancora le derivate parziali di x_i, y_i , punti sulla parete, dipendenti dalla posizione e dall'orientamento del robot. Nel precedente paragrafo si è

sviluppato il calcolo per determinare un generico punto (x_p, y_p) sulla parete quando fosse nota la configurazione del robot. Risolvendo i vari sistemi di equazioni proposti si ottiene la seguente soluzione:

$$\begin{aligned} y_p &= -\frac{-m_p x_{r,y} \cos(\theta_i + x_{r,\theta}) + m_p \sin(\theta_i + x_{r,\theta}) x_{r,x} + q_p \sin(\theta_i + x_{r,\theta})}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ x_p &= -\frac{q_p \cos(\theta_i + x_{r,\theta}) - x_{r,y} \cos(\theta_i + x_{r,\theta}) + x_{r,x} \sin(\theta_i + x_{r,\theta})}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \end{aligned} \quad (3.23)$$

con $(x_i, y_i) = (x_p, y_p)$ e

$$\begin{aligned} m_p &= \frac{P_{1,y} - P_{2,y}}{P_{1,x} - P_{2,x}} \\ q_p &= -\frac{-P_{2,x} P_{1,x} + P_{2,x} P_{1,y}}{P_{1,x} - P_{2,x}}. \end{aligned}$$

Le derivate parziali:

$$\begin{aligned} \frac{\partial x_i}{\partial x_{r,y}} &= -\frac{\sin(\theta_i + x_{r,\theta})}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ \frac{\partial x_i}{\partial x_{r,\theta}} &= \frac{\cos(\theta_i + x_{r,\theta})}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ \frac{\partial y_i}{\partial x_{r,y}} &= -\frac{q_p - x_{r,y} + m_p x_{r,x}}{m_p^2 \cos(\theta_i + x_{r,\theta})^2 - 2 \sin(\theta_i + x_{r,\theta}) m_p \cos(\theta_i + x_{r,\theta}) + 1 - \cos(\theta_i + x_{r,\theta})^2} \\ \frac{\partial y_i}{\partial x_{r,\theta}} &= -\frac{m_p \sin(\theta_i + x_{r,\theta})}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ \frac{\partial x_i}{\partial x_{r,x}} &= \frac{m_p \cos(\theta_i + x_{r,\theta})}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ \frac{\partial y_i}{\partial x_{r,x}} &= -\frac{m_p (q_p - x_{r,y} + m_p x_{r,x})}{m_p^2 \cos(\theta_i + x_{r,\theta})^2 - 2 \sin(\theta_i + x_{r,\theta}) m_p \cos(\theta_i + x_{r,\theta}) + 1 - \cos(\theta_i + x_{r,\theta})^2} \end{aligned} \quad (3.24)$$

dalle quali, sostituendo nelle (3.22),

$$\begin{aligned} \frac{\partial h}{\partial x_{r,x}} &= \frac{m_p}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ \frac{\partial h}{\partial x_{r,y}} &= -\frac{1}{m_p \cos(\theta_i + x_{r,\theta}) - \sin(\theta_i + x_{r,\theta})} \\ \frac{\partial h}{\partial x_{r,\theta}} &= \frac{q_p \cos(\theta_i + x_{r,\theta}) - x_{r,y} \cos(\theta_i + x_{r,\theta}) + m_p \cos(\theta_i + x_{r,\theta}) x_{r,x}}{m_p^2 \cos(\theta_i + x_{r,\theta})^2 - 2 \sin(\theta_i + x_{r,\theta}) m_p \cos(\theta_i + x_{r,\theta}) + 1 - \cos(\theta_i + x_{r,\theta})^2} + \\ &\quad + \frac{\sin(\theta_i + x_{r,\theta}) m_p q_p - m_p x_{r,y} \sin(\theta_i + x_{r,\theta})}{m_p^2 \cos(\theta_i + x_{r,\theta})^2 - 2 \sin(\theta_i + x_{r,\theta}) m_p \cos(\theta_i + x_{r,\theta}) + 1 - \cos(\theta_i + x_{r,\theta})^2} \end{aligned} \quad (3.25)$$

Le (3.25) costituiscono gli elementi della matrice H .

$$H \triangleq \begin{bmatrix} \frac{\partial h_1}{\partial x_{r,x}} & \frac{\partial h_1}{\partial x_{r,y}} & \frac{\partial h_1}{\partial x_{r,\theta}} \\ \frac{\partial h_2}{\partial x_{r,x}} & \frac{\partial h_2}{\partial x_{r,y}} & \frac{\partial h_2}{\partial x_{r,\theta}} \\ \dots & & \\ \frac{\partial h_N}{\partial x_{r,x}} & \frac{\partial h_N}{\partial x_{r,y}} & \frac{\partial h_N}{\partial x_{r,\theta}} \end{bmatrix}$$

Inserendo nelle espressioni i vari coefficienti numerici si ottiene la matrice H calcolata sul punto X_r ⁴.

Nota numericamente H , il calcolo della matrice di covarianza $S(k+1)$ si semplifica notevolmente. Infatti

$$\begin{aligned} S(k+1) &\triangleq E\{[y(k+1) - \hat{y}(k+1)][y(k+1) - \hat{y}(k+1)]'\} \\ &= E\{\hat{y}(k+1)\hat{y}'(k+1)\} + E\{y(k+1)y'(k+1)\} + \\ &\quad - E\{y(k+1)\hat{y}'(k+1)\} - E\{\hat{y}(k+1)y'(k+1)\} \\ &= E\{[H(k+1)\hat{x}_r(k+1|k)][H(k+1)\hat{x}_r(k+1|k)]'\} + E\{y(k+1)y'(k+1)\} \\ &= H(k+1)E\{\hat{x}_r(k+1|k)\hat{x}_r'(k+1|k)\}H'(k+1) + E\{y(k+1)y'(k+1)\} \\ &= H(k+1)P(k+1|k)H'(k+1) + R(k+1) \end{aligned}$$

dove la terza uguaglianza vale nell'ipotesi di incorrelazione tra $y(k+1)$ e $\hat{y}(k+1)$. La matrice di covarianza trovata è un indice sulla qualità del confronto effettuato.

L'ultimo passo dell'algoritmo, note le due matrici di covarianza $S(k+1)$ e $F(k+1)$, consiste nel calcolare i guadagni $K(k+1)$ del filtro di Kalman. Si ripetono le equazioni (2.9):

$$\begin{aligned} \hat{x}_i &= \hat{x}_{i-1} + K_i^o(y_i - C_i\hat{x}_{i-1}) \\ K_i^o &= (S_{i-1}C_i')(U_i + C_iS_{i-1}C_i')^{-1} \\ S_i &= [I - K_i^oC_i]S_{i-1} \end{aligned}$$

che si particolarizzano nel seguente modo:

- C_i in questo caso è $H(k+1)$, infatti come C_i , $H(k+1)$ trasforma lo stato x_r nell'uscita, costituita dalla predizione della misura dei sensori;
- S_i , errore quadratico medio nello stato, è equivalente a $P(k+1|k+1)$;
- U_i è l'errore nella misura diretta dei sonar, equivalente a $R(k+1)$;

da cui

$$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K_{k+1}^o\tilde{y}(k+1) \quad (3.26)$$

$$P(k+1|k+1) = [I - K_{k+1}^oH(k+1)]P(k+1|k) \quad (3.27)$$

$$\begin{aligned} K_{k+1}^o &= P(k+1|k)H'(k+1) \left(R(k+1) + H(k+1)P(k+1|k)H'(k+1) \right)^{-1} \\ &= P(k+1|k)H'(k+1)S^{-1}(k+1) \end{aligned} \quad (3.28)$$

⁴Anche per la matrice H si è omessa la dipendenza dal tempo

Queste equazioni completano la fase di stima.

3.4.4 Ancora su fase di predizione e fase di stima

L'algoritmo sviluppato elabora le informazioni in momenti successivi. Nella prima fase considera dati noti al tempo k generando ipotesi sullo stato del robot al tempo $k + 1$. Nella seconda fase esegue delle misure e le confronta con l'insieme delle predizioni effettuate al tempo k .

Le due fasi sono percorse per ogni passo k : si esegue per prima la fase di stima che definisce la la posizione $X_r(k, k)$, successivamente si esegue la fase di predizione per il tempo $k + 1$ che, utilizzando i dati odometrici del tempo k e la posizione $X_r(k|k)$, calcola la posizione ipotetica al tempo $k + 1$ e le relative misure ultrasoniche.

La qualità del confronto $\tilde{Y}(k)$ è valutata in base alle matrici di covarianza: i guadagni $K(k)$ hanno infatti valori dipendenti dalle covarianze. Ad ogni passo è necessario ripercorrere interamente tutto l'algoritmo esposto.

Capitolo 4

Simulazioni

4.1 Caratteristiche robot

Le caratteristiche riportate sono quelle considerate nelle simulazioni.

Impianto sensoriale:

- Encoder sulle ruote
- 16 sonar posti sulla circonferenza. Il range di misura è $[0, 6.477]$ m con una incertezza di ± 0.0254 m.

Caratteristiche di movimento:

- Massima velocità di traslazione: $20 \text{ inch/sec} = 0.508 \text{ m/s}$
- Massima velocità di rotazione: $45 \text{ gradi/sec} = 0.7854 \text{ rad/s}$

Il tempo di campionamento utilizzato, $T = 0.9$, è dovuto all'insieme di calcoli necessari per l'uso del filtro di Kalman esteso (inversione di matrici), ai tempi per acquisire i dati dei sedici sensori ultrasonici e a quelli per la trasmissione delle informazioni e la ricezione dei comandi. $T = 0.9$ non è sufficientemente piccolo per garantire la completa validità delle equazioni del moto discrete: l'eccessiva approssimazione introdotta può essere valutata come un errore odometrico ed affrontata ancora con il filtro di Kalman.

4.2 Ambiente di simulazione

L'algoritmo descritto è stato implementato in C++ e testato in un apposito ambiente simulativo. L'architettura di simulazione consiste in un rapporto client-server; il client è il programma che implementa il filtro di Kalman esteso mentre il server si preoccupa di attuare le disposizioni del client mostrando i risultati su un'opportuna interfaccia grafica. Il robot ed il server condividono gli stessi comandi: la differenziazione tra simulatore e robot si attua nelle relazioni con l'ambiente esterno in quanto, nel simulatore, sussistono evidenti semplificazioni.

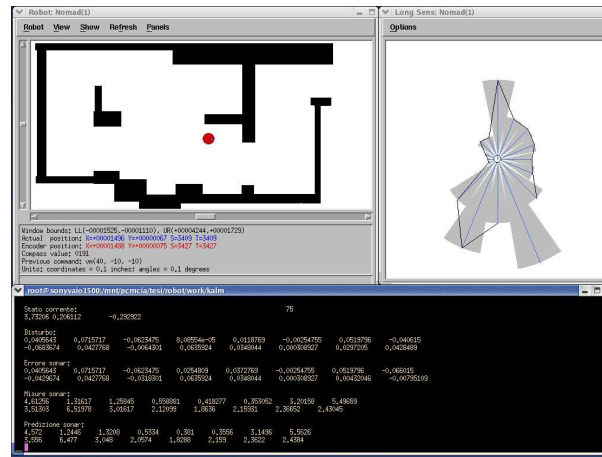


Figura 4.1: Ambiente di simulazione

Sulla sinistra della figura 4.2 è riprodotto bidimensionalmente lo spazio di lavoro del robot. Le superfici nere rappresentano eventuali pareti, porte e ostacoli. La finestra sulla destra riporta graficamente l'andamento delle misure sonar. Nella finestra di shell in basso è presente l'output del programma client.

Le misure degli encoder e dei sonar nel simulatore non sono perturbate da alcun errore, sono esatte: per valutare l'algoritmo di localizzazione sono stati introdotti quindi disturbi gaussiani.

4.3 Analisi dei disturbi

La movimentazione del robot è controllata con periodo inferiore o uguale a 0.1s. L'aggiornamento della stima di posizione ha periodo di 0.9s: è possibile ave-

re differenza di periodi di controllo e stima solo in ambiente simulato mediante l'interazione di diversi processi con un unico server¹.

Il controllo del robot con un tempo di campionamento molto più breve del relativo di stima comporta la possibilità di introdurre errori odometrici con una certa facilità. Un esempio:

- $t = 0$ il robot stima la velocità che poi sarà utilizzata nell'integrazione delle equazioni numeriche
- $t = 0.1$ il robot cambia movimentazione
- $t = 0.7$ il robot cambia movimentazione
- $t = 0.9$ il robot stima la nuova velocità

Quanto avvenuto nell'intervallo di tempo $[0.1, 0.7]$ non è considerato dal robot e porta ad un graduale disallineamento tra posizione stimata e posizione effettiva. La variazione di velocità nell'intervallo può essere vista come un errore odometrico e risolto risolvibile con il filtro di Kalman esteso.

Considerando già sufficiente l'errore sulla stima, a causa del lungo tempo di campionamento, non sono stati introdotti ulteriori disturbi sulle equazioni di stato.

Le misure sonar sono soggette ad un errore relativo alla bassa risoluzione dei sensori, cui è stato aggiunto un ulteriore disturbo gaussiano a media nulla e varianza unitaria, opportunamente premoltiplicato per una costante.

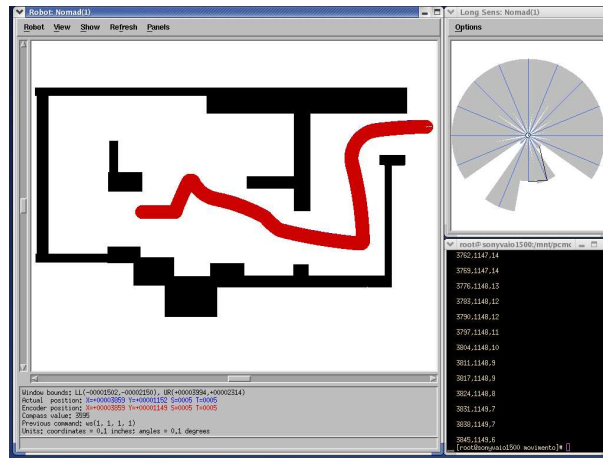
Il disturbo gaussiano inserito è generato da una distribuzione uniforme mediante l'applicazione del teorema del limite centrale². Per ogni estrazione con distribuzione gaussiana sono state utilizzate 256 estrazioni a distribuzione uniforme.

4.4 Risultati delle simulazioni

Come campione rappresentativo è analizzato un percorso lungo e fortemente variato (fig. 4.2)

¹Nel caso reale, nel robot, funziona un unico processo.

²Il codice in Appendice B



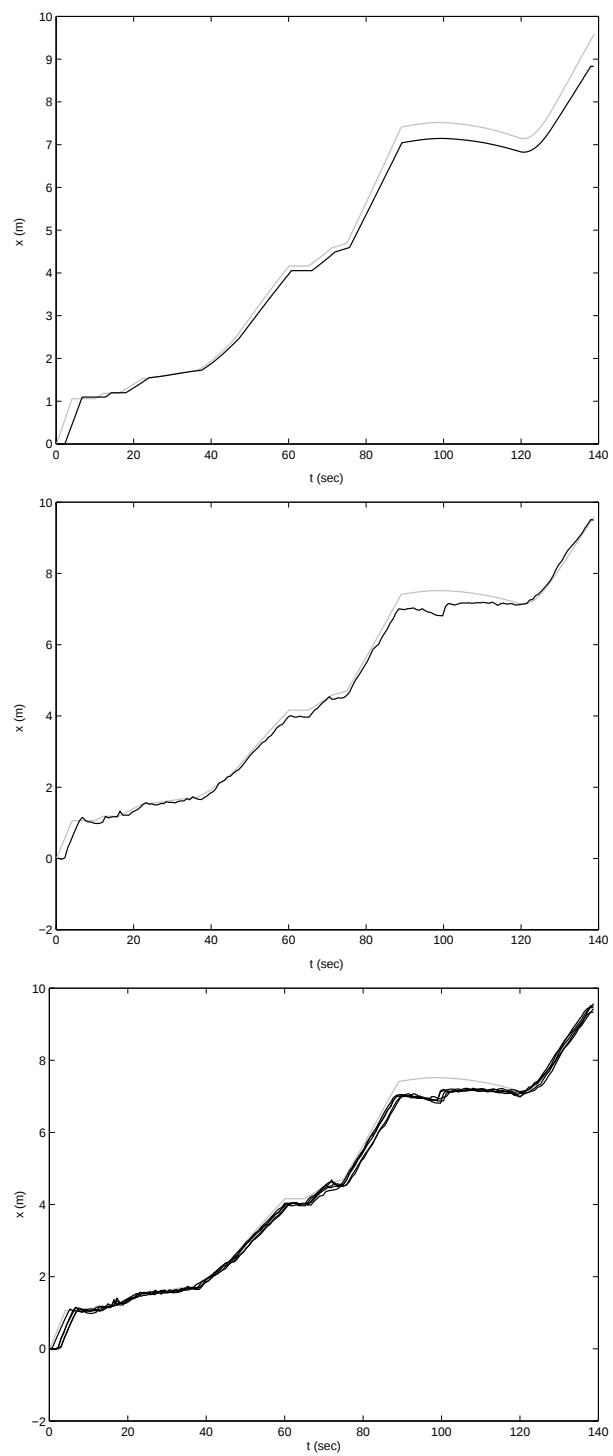


Figura 4.3: Andamento del robot rispetto a all'asse x : senza filtro, con il filtro, più risultati con il filtro.

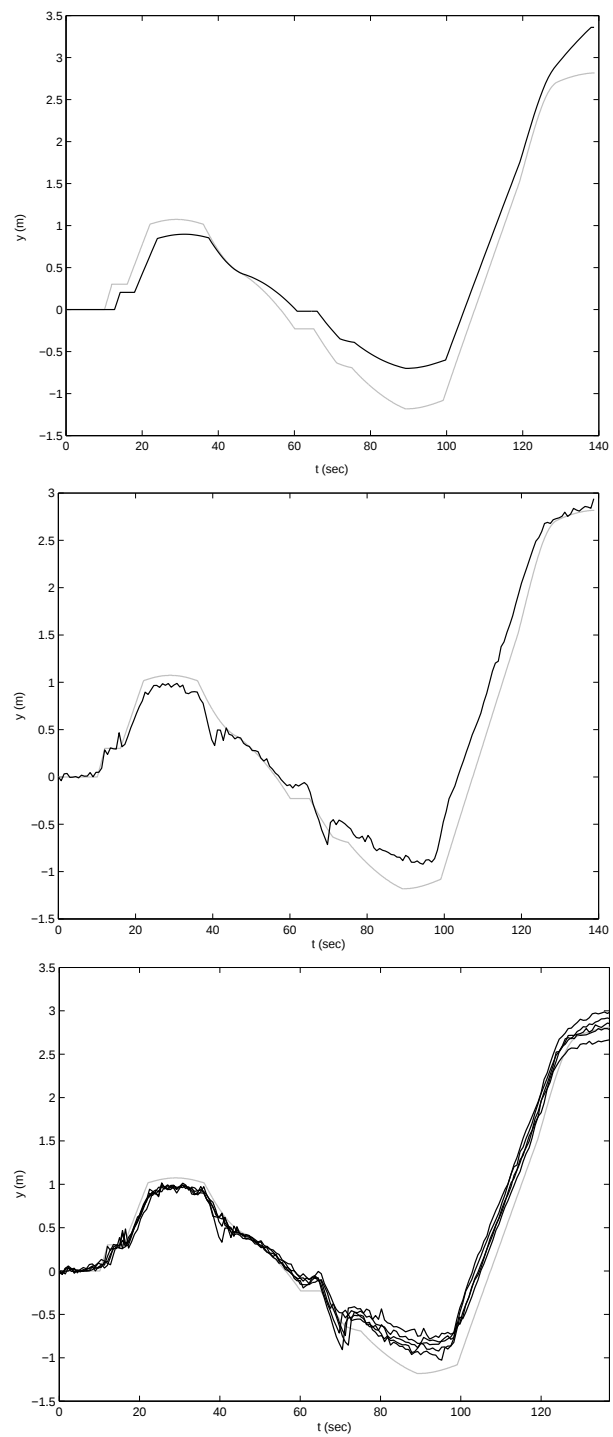


Figura 4.4: Andamento del robot rispetto all'asse y : senza filtro, con il filtro, più risultati con il filtro.

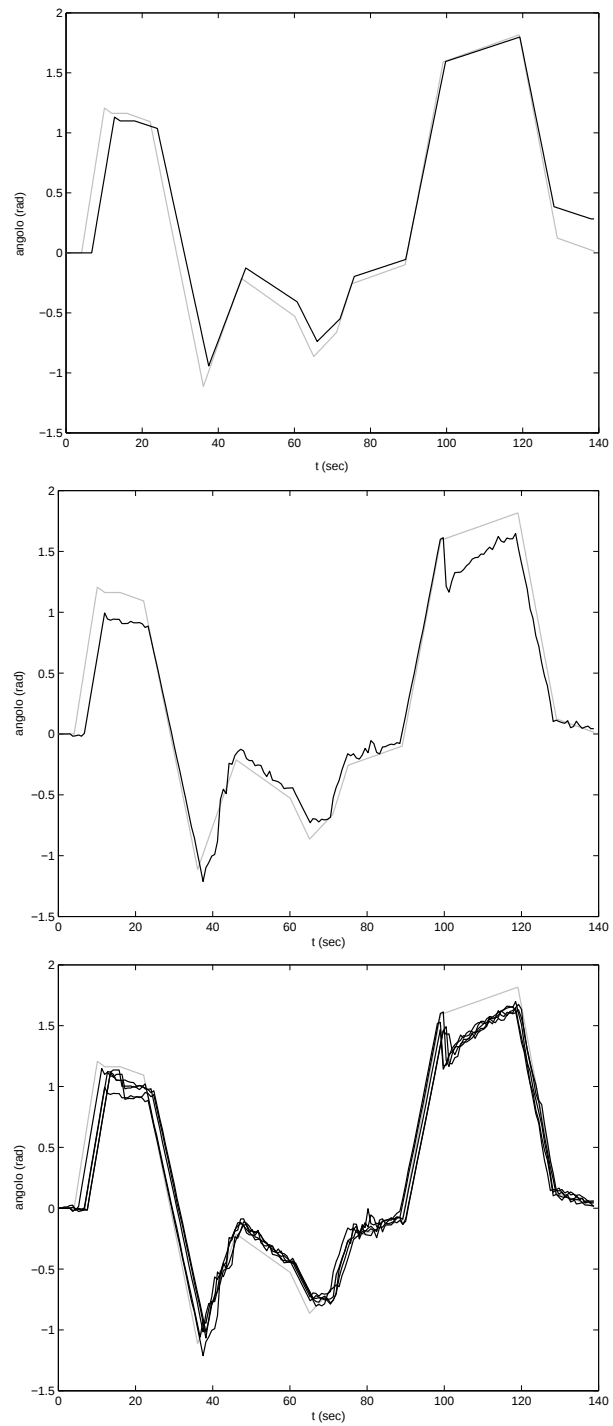


Figura 4.5: Andamento del robot rispetto all'orientamento: senza filtro, con il filtro, più risultati con il filtro.

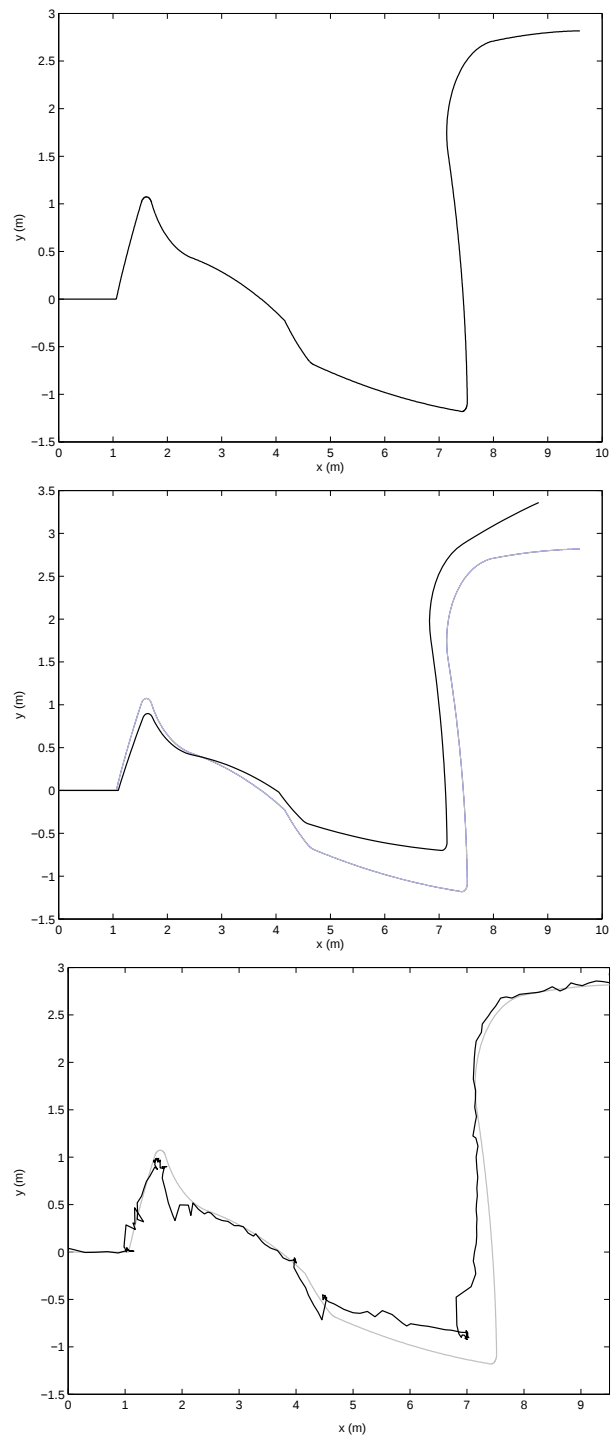


Figura 4.6: Percorso del robot: traiettoria, localizzazione senza filtro, localizzazione con filtro.

Conclusioni

La fusione delle informazioni dei sensori ultrasonici con i dati odometrici, mediante filtro di Kalman esteso, risolve il problema della localizzazione, portando ad una sostanziale coincidenza lo stato stimato con quello effettivo. L'impiego del filtro di Kalman permette il funzionamento dell'algoritmo anche con misure fortemente rumorose come quelle utilizzate nelle varie simulazioni: il robot oscilla intorno ad un punto medio.

I limiti dell'approccio considerato sono costituiti proprio dalle oscillazioni dovute agli errori di misura: la stima non diverge dalla posizione effettiva ma diviene nervosamente variabile e attuare un controllo in funzione di questa non è facile. Per migliorare la risposta complessiva è necessario agire sulle caratteristiche hardware del robot:

- sensori acustici con maggiore risoluzione;
- maggiore rapidità di rilevamento e lettura degli stessi (il robot si muove mentre si prelevano le misure sonar);
- maggiore capacità di calcolo per migliorare la qualità della discretizzazione delle equazioni del moto riducendo l'intervallo di campionamento (ora $T = 0.9s$);
- numero più elevato dei sensori sonar per aumentare la risoluzione angolare.

Le maggiori vibrazioni sulla stima dello stato, relativamente alle coordinate cartesiane, sono dovute alla rumorosità della misura dei sensori ultrasonici. La stima dell'orientamento è fortemente dipendente dalla bassa risoluzione angolare degli stessi. Agendo su quantità e qualità dei sensori acustici si dovrebbe ottenere un notevole miglioramento della risposta dell'algoritmo.

La qualità della risposta, infine, è dovuta anche alle notevoli approssimazioni lineari effettuate. Il filtro viene applicato su un modello fortemente non lineare, linearizzato e trattato come un sistema lineare non stazionario, come si nota dai calcoli sulle relative matrici di covarianza.

Appendice A

L'algoritmo

```
#include <time.h>
#include "matrix.h"
#include <iostream>
#include <stdlib.h>
#include "Nclient.h"
#include "utility.h"
#include <unistd.h>

#define T 0.9      //periodo di un ciclo in secondi
#define inch2cm 2.54      // da inch a cm
#define grad2rad 0.017453      // da gradi a rad
#define inch2m 0.0254      // da inch a m

#ifndef _NO_NAMESPACE
using namespace std;
using namespace math;
#define STD std
#else
#define STD
#endif

#ifndef _NO_TEMPLATE
typedef matrix<double> Matrix;
#else
typedef matrix Matrix;
#endif

int main ()
{
    /* Definizione delle principali variabili*/

    double gauss_dis[16];
```

```

Matrix STATO_s;           // stima
Matrix STATO_p;           // predizione
Matrix P_s;               // stima
Matrix P_p;               // predizione
Matrix H;
Matrix R;                  (...)

STATO_s.SetSize(3,1);
STATO_p.SetSize(3,1);
P_s.SetSize(3,3);
P_p.SetSize(3,3);
H.SetSize(16,3);
R.SetSize(16,16);         (...)

/* Inizializzazione delle variabili */           (...)
/* Connessione al robot e inizializzazione */     (...)
/* Inizializzazione del generatore casuale */     (...)

/* Inizializzazione ciclo di calcolo */

cout << "Ciclo di calcolo..." <<endl;
gs();
STATO_s(0,0) = inch2m*static_cast<double>(0.1*State[STATE_CONF_X]);
STATO_s(1,0) = inch2m*static_cast<double>(0.1*State[STATE_CONF_Y]);
STATO_s(2,0) = grad2rad*static_cast<double>(0.1*State[STATE_CONF_STEER]);

/* Ciclo di calcolo */

clock_t start, end;
start = clock();

for(;;)
{

    /* FASE DI PREDIZIONE */

    Vel[0] = 0.1*inch2m*static_cast<double>(State[STATE_VEL_TRANS]);
    Vel[1] = 0.1*grad2rad*static_cast<double>(State[STATE_VEL_STEER]);

    //...dello stato
    STATO_p(0,0) = STATO_s(0,0) + T*Vel[0]*cos(STATO_s(2,0)+Vel[1]*(T/2));
    STATO_p(1,0) = STATO_s(1,0) + T*Vel[0]*sin(STATO_s(2,0)+Vel[1]*(T/2));
    STATO_p(2,0) = STATO_s(2,0) + T*Vel[1];

```

```

//...della matrice F(k)
F(0,0) = 1;
F(0,2) = -T*Vel[0]*sin(STATO_s(2,0) + Vel[1]*(T/2));
F(1,1) = 1;
F(1,2) = T*Vel[0]*cos(STATO_s(2,0) + Vel[1]*(T/2));
F(2,2) = 1;

//...della matrice P(k+1|k)
P_p = F*P_s*(~F) + Q;

//...del valore misurato dai sensori
long int sonar_p[45];
int laser[965];
predict_sensors(static_cast<int>((10/inch2m)*STATO_p(0,0)),
                static_cast<int>((10/inch2m)*STATO_p(1,0)),
                static_cast<int>((10/grad2rad)*STATO_p(2,0)),
                static_cast<int>((10/grad2rad)*STATO_p(2,0)),sonar_p,laser);
for (int i=0; i<16; i++)
{
    Y_pred(i,0) = inch2m*static_cast<double>(sonar_p[STATE_SONAR_0+i]);
}

/* FINE DELLA FASE DI PREDIZIONE */

/* Si attende che passi il tempo T */

double cpu_time_used;
double T_rest;
//for(int i= 0; i<100000000; i++);
end = clock();
cpu_time_used = (static_cast<double>(end - start)) / CLOCKS_PER_SEC;
T_rest = T - cpu_time_used;
// cout << T_rest << " *" << endl;
// cout << cpu_time_used << endl;
if (T_rest<0)
{
    cout << endl << "Cycle time not respected! (too slow)" << endl;
    WaitForEnterKey();
}
else
usleep(static_cast<unsigned int>(T_rest*1000000));

/* FASE DI STIMA */

start = clock();

```

```

//...della matrice H (per il calcolo a tempo k+1 basata sui primi k dati)
double ang_sonar = grad2rad*22.5;    //angolo tra due sensori sonar in radianti.

for(int i=0; i<16; i++)
{
    predict(STATO_p(0,0),STATO_p(1,0),STATO_p(2,0),i,&d,&dhdx,&dhdy,&dhda,PUNTI);
    H(i,0) = cos(ang_sonar*i + STATO_p(2,0));
    H(i,1) = sin(ang_sonar*i + STATO_p(2,0));
    H(i,2) = 0;
}

//...della Y_err
gs();
for(int i=0; i<16;i++)
{
    gauss_dis[i]=0.05*gauss();
    Y_misur(i,0)=inch2m*static_cast<double>(State[STATE_SONAR_0+i])+gauss_dis[i];
}
for (int i=0; i<16; i++)
{
    if( Y_misur(i,0)-Y_pred(i,0) <= 0.6 && Y_misur(i,0)-Y_pred(i,0) >= -0.6 )
    {
        Y_err(i,0) = Y_misur(i,0)-Y_pred(i,0);
    }
    else
    {
        Y_err(i,0) = 0;
    }
}

//...della matrice K
K = P_p*(~H)*((R + H*P_p*(~H)).Inv());

//...della matrice P_s (aggiornamento al passo (k+1|k+1)
P_s = (I-K*H)*P_p;

//...dello stato STATO_s (aggiornamento al passo (K+1|K+1)
STATO_s = STATO_p - K*Y_err;

dp( static_cast<long int>((10/inch2m)*STATO_s(0,0)),
    static_cast<long int>((10/inch2m)*STATO_s(1,0))    );
da( static_cast<long int>((10/grad2rad)*STATO_s(2,0)),
    static_cast<long int>((10/grad2rad)*STATO_s(2,0))    );

/* FINE DELLA FASE DI STIMA */
/* Risultati */ (...)

```

Generazione del rumore gaussiano

47

Appendice C

La funzione predict

```
double P1x = 0;
double P1y = 0;
double P2x = 0;
double P2y = 0;
int nsegmenti = PUNTI[0];
double mp[nsegmenti];
double qp[nsegmenti];
double xp[nsegmenti];
double yp[nsegmenti];
double dist[nsegmenti];
int indice = 0;

// ...Calcolo dei punti xp,yp, per ogni segmento
for (int i=0; i<PUNTI[0]; i++)
{
    // ...retta approssimante il segmento
    P1x = (static_cast<double>(PUNTI[4*i+1]))*0.1*inch2m;
    P1y = (static_cast<double>(PUNTI[4*i+2]))*0.1*inch2m;
    P2x = (static_cast<double>(PUNTI[4*i+3]))*0.1*inch2m;
    P2y = (static_cast<double>(PUNTI[4*i+4]))*0.1*inch2m;

    if (fabs(P1x-P2x) < 0.05 && fabs(P1y-P2y) > 0.1)
    {
        mp[i] = 999999;
        qp[i] = P1y-mp[i]*P1x;
    }
    else
    {
        mp[i]= (P1y-P2y) / (P1x-P2x);
        qp[i]= (P2y*P1x - P2x*P1y) / (P1x-P2x);
    }
}
```

```

        yp[i]= -(-mp[i]*cos(a+a0*n_sonar)*y+mp[i]*sin(a+a0*n_sonar)*x+
                +qp[i]*sin(a+a0*n_sonar))/(mp[i]*cos(a+a0*n_sonar)+
                -sin(a+a0*n_sonar));
        xp[i]= -(qp[i]*cos(a+a0*n_sonar)-y*cos(a+a0*n_sonar)+
                +sin(a+a0*n_sonar)*x)/(mp[i]*cos(a+a0*n_sonar)+
                -sin(a+a0*n_sonar));

        // ...appartenenza del punto al segmento
        if (yp[i] < min_dub(P1y,P2y) || yp[i] > max_dub(P1y,P2y))
        {
        if(xp[i] < min_dub(P1x,P2x) || xp[i] > max_dub(P1x,P2x))
        {
            xp[i] = 999999;
            yp[i] = 999999;
        }
    }

    //...scelta del semipiano di intersezione valido
    if (cos(a+a0*n_sonar)*(xp[i]-x)<0 || sin(a+a0*n_sonar)*(yp[i]-y)<0)
    {
        xp[i] = 999999;
        yp[i] = 999999;
    }

    //...calcolo delle distanze
    if (xp[i] > 99999)
    dist[i] = 99999;
    else
    dist[i] = sqrt( pow((x+0.25*cos(a+a0*n_sonar)-xp[i]),2)+
                    +pow((y+0.25*sin(a+a0*n_sonar)-yp[i]),2));
    }

    //... la distanza minima
    indice = min_vet(dist,nsegmenti);
    cout << endl << dist[indice];

    //...risultati
    if (dist[indice]<99990)
    {
        *d = dist[indice];
        *dhdx = mp[indice]/( mp[indice]*cos(a+a0*n_sonar)-sin(a+a0*n_sonar));
        *dhdy = -1 / ( mp[indice]*cos(a+a0*n_sonar) - sin(a+a0*n_sonar) );
        *dhda = (qp[indice]*cos(a+a0*n_sonar) - y*cos(a+a0*n_sonar) +
                + mp[indice]*cos(a+a0*n_sonar)*x + sin(a+a0*n_sonar)*
                *mp[indice]*qp[indice] - mp[indice]*y*sin(a+a0*n_sonar) +
                + sin(a+a0*n_sonar)*pow(mp[indice],2)*x) /

```

```

        (pow(mp[indice]*cos(a+a0*n_sonar),2)+
        -2*sin(a+a0*n_sonar)*mp[indice]*cos(a+a0*n_sonar)+
        +1-pow(cos(a+a0),2));
    }
else
{
    *d = 99999;
    *dhdx=0;
    *dhdy=0;
    *dhda=0;
}

return indice;
}

```

Bibliografia

- [1] G. Galati, M. Naldi, G. Pavan. *Teoria dei fenomeni aleatori: Fondamenti e prime applicazioni ai modelli stocastici*, volume I. Texmat, 2002.
- [2] G. Galati, M. Naldi, G. Pavan. *Teoria dei fenomeni aleatori: Modelli Stocastici e Metodi Statistici*, volume II. Texmat, 2002.
- [3] S. Lipschutz, M. Lipson. *Algebra Lineare*. Schaum's, 2001.
- [4] E. Fornasini, G. Marchesini. *Appunti di Teoria dei Sistemi*. Edizioni Libreria Progetto, 1994.
- [5] O. M. Grasselli. *Proprietà strutturali dei sistemi lineari e stazionari*. Pitagora Editrice, 1978.
- [6] H. Kwakernaak, R. Sivan. *Linear Optimal Control Systems*. Wiley-Interscience, 1972.
- [7] M. Bodson S. Sastry. *Adaptive Control: Stability, Convergence, and Robustness*, chapter I,II,III. 1994. <http://www.ece.utah.edu/~bodson/acscr/>.
- [8] O. M. Grasselli. Dispense sugli osservatori dello stato.
- [9] G. Welch, G. Bishop. An introduction to the kalman filter. SIGGRAPH, 2001.
- [10] F. Grandoni, A. Martinelli, F. Martinelli, S. Nicosia, P. Valigi. Sensor fusion for robot localisation. Università di Roma Tor Vergata.
- [11] Requisiti, specifiche e stato dell'arte degli algoritmi di fusione sensoriale. Technical report, Università di Roma Tor Vergata, 2000. Accordo di Programma ENEA-NURST L.95/95, Calcolo parallelo con applicazioni alla Robotica; Relazione 1.1: stato di avanzamento al 31.12.2000.
- [12] Definizione degli algoritmi per la gestione delle mappe. Technical report, Università di Roma Tor Vergata, 2000. Accordo di Programma ENEA-NURST L.95/95, Calcolo parallelo con applicazioni alla Robotica; Relazione 1.2: stato di avanzamento al 31.12.2000.

- [13] Sviluppo di algoritmi di fusione sensoriale e gestione delle mappe. Technical report, Università di Roma Tor Vergata, 2000. Accordo di Programma ENEA-NURST L.95/95, Calcolo parallelo con applicazioni alla Robotica; Relazione 1.3: stato di avanzamento al 31.12.2000.
- [14] Definizione e sviluppo algoritmi di pianificazione e nativazione. Technical report, Università di Roma Tor Vergata, 2001. Accordo di Programma ENEA-NURST L.95/95, Calcolo parallelo con applicazioni alla Robotica; Relazione 2.1: stato di avanzamento al 31.12.2001.
- [15] Sviluppo algoritmi e integrazione e prove di laboratorio. Technical report, Università di Roma Tor Vergata, 2001. Accordo di Programma ENEA-NURST L.95/95, Calcolo parallelo con applicazioni alla Robotica; Relazione 2.2: stato di avanzamento al 31.12.2001.
- [16] Technical report, Università di Roma Tor Vergata, 2002. Accordo di Programma ENEA-NURST L.95/95, Calcolo parallelo con applicazioni alla Robotica; Relazione sullo stato di avanzamento al 30.10.2002.
- [17] F. Romanelli. *Robot Nomad 150: nozioni di base*. Laboratorio di Automatica (F. Martinelli).