



**UNIVERSITÀ DEGLI STUDI DI ROMA
TOR VERGATA**

FACOLTÀ DI INGEGNERIA

**CORSO DI LAUREA IN INGEGNERIA
DELL'AUTOMAZIONE**

A.A. 2005/2006

Tesi di Laurea di Primo Livello

**Filtro di Kalman esteso per la localizzazione del robot
mobile NOMAD XR4000**

RELATORE

Ing. F. Martinelli

CANDIDATO

Mateusz Gospodarczyk

Indice

Introduzione	1
1 Localizzazione	3
1.1 Generalità	3
1.1.1 Ricostruzione odometrica	4
1.1.2 Localizzazione assoluta	6
1.2 Descrizione del problema	7
1.3 Approcci probabilistici	8
2 Filtro di Kalman lineare a tempo discreto	10
2.1 Filtraggio Bayesiano	10
2.2 Principi generali del filtro di Kalman	12
2.3 L'algoritmo del filtro di Kalman lineare	16
2.3.1 Fase di inizializzazione	16
2.3.2 Fase di predizione	16
2.3.3 Fase di correzione	16
2.4 Cenni sulla taratura	18
3 Filtro di Kalman esteso	19
3.1 Principi generali del filtro	19

3.2	L'algoritmo del filtro di Kalman esteso	23
3.2.1	Fase di inizializzazione	23
3.2.2	Fase di predizione	23
3.2.3	Fase di correzione	24
3.3	Conclusioni	24
4	Nomad XR4000	26
4.1	Generalità	26
4.2	Il sistema olonomo Nomad XR4000	27
4.3	Hardware	28
4.3.1	Sistema Sensus 600 di orientamento	28
4.3.2	Sistema laser Sick Sensus 550	29
4.4	L'architettura XRDev	30
5	FKE applicato al NOMAD XR4000	31
5.1	Il modello cinematico	31
5.2	Il modello dell'osservatore	33
5.3	Localizzazione mediante il FKE	36
5.3.1	Fase di inizializzazione	37
5.3.2	Fase di predizione	37
5.3.3	Fase di correzione	38
5.4	Simulazioni	39
5.4.1	I risultati delle prove	41
6	Conclusioni e sviluppi futuri	50
6.0.2	Sviluppi futuri	50

Appendice A - Simulatore	52
Appendice B - Generazione del rumore Gaussiano	54
Bibliografia	56

Introduzione

Uno degli aspetti fondamentali della robotica mobile è stimare la posizione del robot all'interno dell'ambiente operativo. Tale stima può essere effettuata in maniera differente, a seconda del contesto applicativo in cui il robot viene immerso. Talvolta è sufficiente conoscerne la sola variazione temporale, ossia lo spostamento relativo ad una posizione precedente. Altre volte viene richiesta la conoscenza della posizione assoluta all'interno dell'ambiente, soprattutto nei compiti in cui il robot deve raggiungere locazioni specifiche, fornitegli secondo il sistema di riferimento globale.

In questa tesi viene affrontato e discusso il problema della localizzazione, mediante l'applicazione del filtro di Kalman esteso, del robot mobile NOMAD XR4000.

L'obiettivo della tesi è l'implementazione dell'algoritmo del filtro di Kalman esteso che riesca a stimare in maniera efficiente la posizione del robot in un ambiente noto.

Traccia dell'esposizione

In seguito viene presentato un breve sommario dei capitoli presenti in questa tesi, per una rapida consultazione:

- Nel primo capitolo della tesi viene affrontato e descritto in dettaglio il problema della localizzazione nonché del perché viene adoperato il filtro di Kalman esteso.
- Nel secondo capitolo viene descritto in dettaglio il filtro di Kalman per sistemi

lineari. In particolare viene data una giustificazione alle equazioni che descrivono tale filtro.

- Nel terzo capitolo viene descritto l'oggetto di questa tesi ossia il filtro di Kalman esteso per sistemi non lineari. In particolare viene data una giustificazione alle equazioni che descrivono tale filtro, equazioni che risultano da un'estensione del filtro di Kalman lineare .
- Nel quarto capitolo viene descritto il robot mobile NOMAD XR4000 ed in particolar modo:
 - Il sistema di guida olofono, caratteristico di questo robot;
 - Il sensore laser;
 - La bussola digitale;
- Nel quinto capitolo, il più importante, viene mostrato come il FKE viene adattato, ed impiegato, nella localizzazione del robot mobile NOMAD XR4000, il tutto corredato da simulazioni.
- Il sesto capitolo è destinato alle conclusioni sul lavoro svolto confrontando gli obiettivi prefissati con i risultati ottenuti.
- Nell'appendice A viene descritto il simulatore Nserver.
- Nell'appendice B viene descritta la funzione utilizzata nella generazione del rumore Gaussiano.

Capitolo 1

Localizzazione

La localizzazione realizza la capacità di un robot di conoscere la propria posizione e il proprio orientamento in un ambiente noto, a partire da dati sensoriali.

1.1 Generalità

A seconda di come viene codificata la posizione del robot e di quale metodo di misura della posizione viene usato, possiamo identificare diverse categorie di localizzazione.

Una prima suddivisione, quindi, può essere così fatta:

- Localizzazione dipendente dalla codifica della posizione:
 - Localizzazione in coordinate cartesiane;
 - Localizzazione su mappa topologica;
- Localizzazione dipendente dal metodo di misura e/o calcolo della posizione :
 - Localizzazione relativa:
 - * Ricostruzione Odometrica;
 - * Navigazione Inerziale;

- Localizzazione assoluta:
 - * Landmark naturali/artificiali;
 - * Boe attive/passive;
 - * Mappe.

In questa tesi viene affrontato il problema della localizzazione assoluta mediante l'uso combinato della bussola digitale e del raggio laser. Solitamente la localizzazione assoluta sfrutta l'odometria, descritta nel paragrafo seguente, in modo da diminuire lo spazio di ricerca del robot nell'ambiente.

1.1.1 Ricostruzione odometrica

L'odometria, nonostante sia affetta da errori sistematici e non, descritti di seguito, è il metodo più usato per la localizzazione dei robot mobili in quanto non ha alti costi di applicazione ed è facilmente integrabile con altri metodi di localizzazione che invece danno misure più accurate. L'idea di base della ricostruzione odometrica è quella del calcolo della nuova posizione del robot in base alla strada percorsa rispetto alla posizione precedente. Il calcolo avviene tramite l'integrazione nel tempo dell'informazione sul movimento. Per ricavare i valori da integrare, l'odometria utilizza degli Encoder, attaccati agli assi delle ruote o all'armatura del motore, che vanno a misurare la velocità di rotazione delle ruote e/o l'orientazione dello sterzo (basandosi sul principio che la rotazione compiuta da una ruota può essere tradotta in spostamento lineare).

Errori

Come già detto l'idea base del metodo della ricostruzione odometrica si basa sull'integrazione del movimento del robot nel tempo; questo metodo però porta all'accumularsi di errori, soprattutto di orientazione che causano a loro volta errori in posizione che

crescono proporzionalmente alla distanza percorsa. Oltre a questo tipo di problemi possiamo identificare altri due tipi di errori:

- Errori sistematici.
- Errori non sistematici.

Gli errori sistematici interessano tutti quegli errori dovuti alle diverse approssimazioni fatte nei calcoli visti prima, e possono dipendere da diversi fattori quali:

- diametri delle ruote diseguali;
- diametro medio delle ruote che differisce dal valore nominale;
- lunghezza dell'asse delle ruote che differisce dal valore nominale;
- disallineamento delle ruote;
- risoluzione finita degli encoder;
- campionamento finito degli encoder.

Gli errori non sistematici invece sono causati essenzialmente da fattori accidentali quali:

- movimento su terreno irregolare;
- movimento su oggetti imprevisti posti sul terreno;
- slittamento delle ruote dovuto a alte accelerazioni, rotazioni veloci, interazioni con corpi esterni, contatto non puntuale delle ruote col terreno.

E' importante sapere se abbiamo la prevalenza dei primi o dei secondi; ad esempio, su superfici lisce gli errori sistematici hanno un peso notevole perché si accumulano

costantemente e sono quelli che contribuiscono maggiormente alle inesattezze nelle varie misurazioni. Al contrario gli errori non sistematici sono predominanti se il robot si muove su superfici irregolari e sono difficilmente predicibili e/o correggibili. Infatti è possibile solo stimare gli errori sistematici e decidere di conseguenza con che frequenza correggerli con dei riferimenti assoluti. Sono stati creati diversi algoritmi che stimano l'errore della posizione del robot, tra i quali il più usato è quello in cui ogni posizione calcolata è circondata da una "error ellipse" che indica la zona di incertezza della posizione del robot, vedi figura . Tipicamente queste ellissi crescono

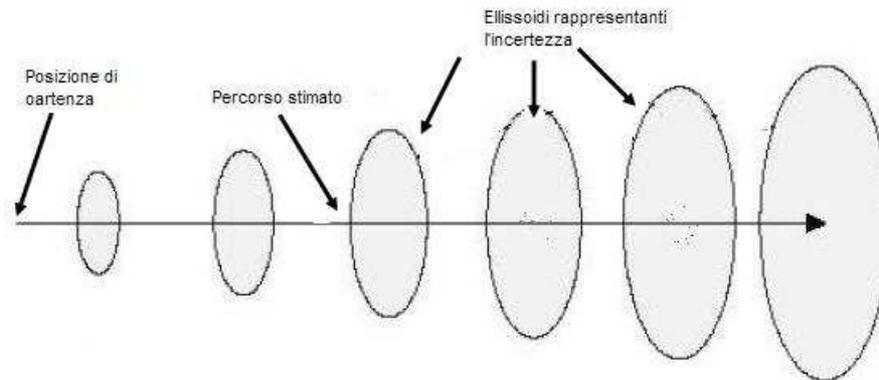


Figura 1.1: Error Ellipse

proporzionalmente alla distanza percorsa, fino a che una misura assoluta riporta la dimensione della stessa allo stato di partenza.

1.1.2 Localizzazione assoluta

La localizzazione assoluta affrontata in questa tesi si basa sulle mappe globali. Questo metodo permette al robot di localizzare la propria posizione in base alle percezioni sensoriali, e una mappa globale dell'ambiente nota a priori. I sensori esterni restituiscono le informazioni sull'ambiente in cui il robot si muove per essere successivamente

confrontate mediante tecniche appropriate con la posizione attuale del robot. In questo modo l'incertezza sulla posizione dipende soltanto dalle caratteristiche del sensore esteroceettivo utilizzato. Un modo per caratterizzare i sensori consiste nell'adoperare l'accuracy.

L'accuracy misura la massima deviazione della misura calcolata dal segnale in uscita rispetto al valor vero quindi fornisce in genere la prestazione peggiore del sistema. Il valore della misura in uscita sarà quindi vero a meno di un errore $\pm\Delta$ dipendente dalla precisione dello strumento.

1.2 Descrizione del problema

Quindi il problema principale nella localizzazione è individuare una corretta associazione tra i dati sensoriali, affetti da errori, e il modello dell'ambiente. Tale associazione, come mostrato nella figura 1.2, si articola in due fasi: la fase della predizione e dell'aggiornamento.

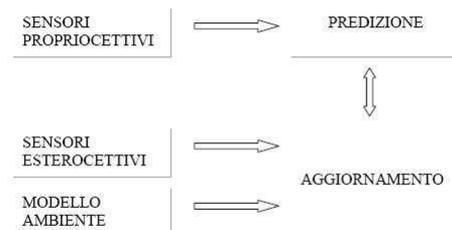


Figura 1.2: Problema della localizzazione.

Nella fase di predizione vengono adoperati i sensori propriocettivi, tali sensori forniscono informazioni sulla configurazione del robot, e sono indipendenti dall'esterno. Essi, ad esempio, sono in grado di fornire dati sulla posizione, sulla velocità, sull'o-

rientazione del robot. In questa categoria rientrano gli encoder e i giroscopi.

Invece nella fase dell'aggiornamento vengono adoperati i sensori esterocezionali, tali sensori sono in grado di effettuare misurazioni esterne al robot e quindi fornire dati sulla configurazione dell'ambiente esterno. I laser rangefinder, i sonar e le bussole sono esempi di questa categoria di sensori.

A causa del rumore, tipico dei processi di misura, la stima della posizione può solamente essere stimata. Una delle tecniche maggiormente utilizzate per stimare l'errore sulla posizione risulta essere il filtro di Kalman esteso, tale tecnica adoperata un approccio statistico.

1.3 Approcci probabilistici

I problemi di localizzazione possono essere visti come problemi di stima dello stato in un sistema dinamico discreto. Sebbene esistano molte tecniche specifiche per poter stimare lo stato del sistema a partire da un insieme di misurazioni, la maggior parte di queste non considera in maniera esplicita la natura rumorosa delle misurazioni. Tale rumore tipicamente è descritto da statistiche, il che porta a dover utilizzare metodi stocastici per affrontare il problema. In questa categoria di metodi di localizzazione includiamo:

- Filtro di Kalman e filtro di Kalman nella sua versione estesa.
- Markov Localization.

Questi approcci hanno come caratteristica particolare quella di non calcolare una posizione e una orientazione, ma di misurare una densità di probabilità che associa ad ogni punto dello spazio (x, y, θ) la probabilità che il robot si trovi nella configurazione corrispondente.

Tale probabilità viene aggiornata ogni qualvolta vengano ricevuti nuovi dati sensoriali. In particolare il filtro di Kalman tiene in memoria i primi due momenti della densità di probabilità, ovvero media e covarianza, mentre la localizzazione di Markov tiene in memoria l'intera distribuzione di probabilità. Quindi la localizzazione di Markov se da un lato risulta uno dei metodi di localizzazione più robusto dall'altro però soffre di alcuni inconvenienti:

- richieste computazionali elevate: dovuto al passo di integrazione
- instabilità in ambienti dinamici: dovuta all'intenso utilizzo dell'assunzione di Markov nella derivazione matematica.

inconvenienti che portano a preferire il filtro di kalman in quanto ha una bassa complessità computazionale e risulta essere stabile in ambienti dinamici.

Capitolo 2

Filtro di Kalman lineare a tempo discreto

Il filtro di Kalman lineare a tempo discreto è un set di equazioni matematiche che forniscono una soluzione ricorsiva computazionalmente efficiente al metodo dei minimi quadrati. È tra gli strumenti matematici che possono essere utilizzati per il filtraggio bayesiano, discusso nel primo paragrafo. Viene spesso utilizzato nei sistemi lineari per ottenere una migliore valutazione di un dato ottenuto dalla lettura di più sensori, ognuno caratterizzato da un rumore di misura avente caratteristiche differenti nel tempo. Nel seguito della trattazione il filtro di Kalman lineare a tempo discreto verrà denominato filtro di Kalman lineare o FKL per brevità della trattazione.

2.1 Filtraggio Bayesiano

Il problema del filtraggio può essere espresso come stima dello stato x di un sistema dinamico discreto avendo a disposizione:

- La conoscenza analitica della funzione di transizione f_t e la conoscenza statistica del rumore dello stato ω_t .
- La conoscenza analitica della funzione di uscita h_t e la conoscenza statistica del rumore di osservazione ω_t .
- la realizzazione dell'uscita del sistema $z_{1:t}$ fino al tempo t .

Un filtro bayesiano è uno strumento matematico il cui scopo è stimare l'evoluzione dello stato del sistema, date le misurazioni disponibili, ossia stimare la funzione densità di probabilità:

$$p(x_{0:t} | z_{1:t}) \quad (2.1.1)$$

Spesso si è interessati a stimare la densità marginale dello stato corrente, chiamata anche distribuzione filtrata:

$$p(x_t | z_{1:t}) \quad (2.1.2)$$

Una volta formalizzato il problema della stima dello stato come un problema di filtraggio bayesiano, quello che resta da fare è trovare una formulazione matematica che ci permetta di risalire alla densità di probabilità (2.1.1) attraverso la conoscenza del sistema e delle osservazioni che abbiamo. Per prima cosa abbiamo bisogno di modellare sia l'evoluzione interna del sistema in oggetto, sia il modo in cui esso si presenta a noi attraverso le sue uscite osservabili. A questo scopo vengono introdotte due distribuzioni condizionate. La prima, $p(z_t | x_t)$, è il modello di osservazione e rappresenta la densità della misurazione dello stato z dato lo stato del sistema x . La seconda, $p(x_t | x_{t-1})$, è il modello d'evoluzione. A questo punto utilizzando la regola di Bayes è possibile ottenere una densità di probabilità a posteriori dello stato in maniera incrementale:

$$p(x_{0:t+1} | z_{1:t+1}) = p(x_{0:t} | z_{1:t}) \frac{p(z_{t+1} | x_{t+1})p(x_{t+1} | x_t)}{p(z_{t+1} | z_{1:t})} \quad (2.1.3)$$

Mentre per quanto riguarda la distribuzione filtrata:

$$p(x_{t+1} | z_{1:t+1}) = \eta p(z_{t+1} | x_{t+1}) \int p(x_{t+1} | x_t) p(x_t | z_{1:t}) dx_t \quad (2.1.4)$$

dove η è il fattore di normalizzazione. Solitamente la valutazione dell'equazione (2.1.4) avviene in due passaggi. Nella fase di predizione viene calcolato lo stato x_{t+1} a partire

dallo stato x_t , attraverso l'applicazione del modello di transizione. Successivamente l'osservazione z_t viene incorporata all'interno della funzione di densità di probabilità precedentemente calcolata, attraverso la fase di aggiornamento.

2.2 Principi generali del filtro di Kalman

Sia $x \in \mathfrak{R}^n$ lo stato di un processo lineare a tempo discreto che si voglia stimare. Il processo è descritto dall'equazione lineare stocastica:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (2.2.1)$$

Ove:

- w_{k-1} è il rumore del processo, assunto a media nulla, bianco e con distribuzione di probabilità normale:

$$p(w) \sim N(0, Q)$$

Q è la matrice di covarianza del rumore di processo, assunta simmetrica e semidefinita positiva.

- $A^{n \times n}$ è la matrice che mette in relazione lo stato all'istante $k - 1$ con lo stato all'istante k .
- u_{k-1} è l'ingresso del processo.
- $B^{n \times l}$ è la matrice che mette in relazione l'ingresso di controllo $u \in \mathfrak{R}^l$ con x .

Sia poi:

$$z_k = Hx_k + v_k \quad (2.2.2)$$

la misura del processo, con $z \in \mathfrak{R}^m$, ove:

- v_k è il rumore di misura, assunto a media nulla, bianco e con distribuzione di probabilità normale:

$$p(v) \sim N(0, R)$$

R è la matrice di covarianza del rumore di misura, assunta simmetrica e definita positiva.

- $H^{m \times n}$ è la matrice che mette in relazione lo stato del processo con la misura effettuata.

Infine, i rumori w_k e v_k sono considerati indipendenti tra di loro.

L'equazione (2.2.1) rappresenta un processo reale soggetto al rumore aleatorio w_k , di cui non si conosce il valore, quindi tale quantità può anche essere trascurata a patto, però, di introdurre $\hat{x}^- \in \mathfrak{R}^n$, la stima dello stato eseguita a priori, e $\hat{x} \in \mathfrak{R}^n$, la stima dello stato eseguita a posteriori, cioè una volta nota la misura z_k . Una volta introdotte codeste variabili possiamo definire l'errore di stima a priori ed a posteriori nonché le relative matrici di covarianza. L'errore di stima a posteriori risulta:

$$e_k^- = x_k - \hat{x}_k^- \quad (2.2.3)$$

mentre quello a priori è:

$$e_k = x_k - \hat{x}_k \quad (2.2.4)$$

La covarianza dell'errore stimato a priori risulta:

$$P_k^- = E[e_k^- e_k^{-T}] \quad (2.2.5)$$

mentre la covarianza dell'errore stimato a posteriori è:

$$P_k = E[e_k e_k^T] \quad (2.2.6)$$

Introducendo a questo punto la 2.2.7

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H_k \hat{x}_k^-) \quad (2.2.7)$$

dove la stima a posteriori risulta come combinazione lineare di \hat{x}_k^- stimato a posteriori con una differenza, pesata secondo K , tra la misura z_k , reale, ed una previsione per la stessa calcolata secondo $H_k \hat{x}_k^-$. Il termine $z_k - H_k \hat{x}_k^-$ viene detto l'innovazione di misura o residuo.

La matrice $K^{n \times m}$ è detta “guadagno di Kalman”, per essa si ricerca un'espressione che minimizzi la matrice di covarianza dell'errore a posteriori. Tale espressione è determinata nel seguente modo:

1. Si esplicita la (2.2.7) in funzione della (2.2.4):

$$e_k = (x_k - \hat{x}_k^- - K(z_k - H_k \hat{x}_k^-))$$

2. Si riscrive la (2.2.6):

$$P_k = E[e_k e_k^T] = E[(x_k - \hat{x}_k^- - K(z_k - H_k \hat{x}_k^-))(x_k - \hat{x}_k^- - K(z_k - H_k \hat{x}_k^-))^T]$$

3. Si esplicita z_k e si sviluppano i prodotti:

$$\begin{aligned} P_k &= E[(x_k - \hat{x}_k^- - K(Hx_k + v_k - H_k \hat{x}_k^-))(x_k - \hat{x}_k^- - K(Hx_k + v_k - H_k \hat{x}_k^-))^T] \\ &= E[((I - K_k H)(x_k - \hat{x}_k^-) + K v_k)((I - K_k H)(x_k - \hat{x}_k^-) + K v_k)^T] \\ &= (I - K_k H)E[(x_k - \hat{x}_k^-)(x_k - \hat{x}_k^-)^T](I - K_k H)^T + K_k E[(v_k v_k^T)]K_k^T \\ &\quad + 2(I - K_k H)E[(x_k - \hat{x}_k^-)v_k^T]K_k^T \end{aligned}$$

4. Per l'ipotesi d'indipendenza $E[(x_k - \hat{x}_k^-)v_k^T] = 0$ la precedente equazione diventa:

$$P_k = (I - K_k H)P_k^- (I - K_k H)^T + K_k R K_k^T$$

5. Adesso dobbiamo calcolare la derivata della traccia di P_k rispetto a K_k :

$$\frac{\partial \text{tr}(P_k)}{\partial K_k} = \frac{\partial \text{tr}[(I - K_k H)P_k^- (I - K_k H)^T + \text{tr}(K_k R K_k^T)]}{\partial K_k}$$

per far ciò, si sfrutta la seguente relazione : $\frac{\partial ABA^T}{\partial A} = 2AB$. Ne consegue:

$$\frac{\partial \text{tr}(P_k)}{\partial K_k} = -2(I - K_k H)P_k^- H^T + 2K_k R_k = -2P_k^- H^T + 2K_k (HP_k^- H^T + R_k)$$

6. Infine imponendo che $\frac{\partial \text{tr}(P_k)}{\partial K_k} = 0$ otteniamo:

$$K_k = P_k^- H^T (HP_k^- H^T + R_k)^{-1}$$

Analizzando l'equazione appena trovata possiamo fare alcune considerazioni di natura qualitativa. In primo luogo, se la covarianza dell'errore di misura R_k tende a 0, il guadagno K dà sempre più peso al residuo e, di conseguenza, la misura z_k diviene via via più vera, mentre la previsione della misura sempre meno. Specificatamente

$$\lim_{R_k \rightarrow 0} K_k = H^{-1}$$

D'altra parte, al tendere a zero della covarianza dell'errore di stima a priori P_k^- , il guadagno K dà sempre meno peso al residuo e, di conseguenza, la misura attuale z_k è sempre meno vera, mentre la previsione della misura diventa sempre più veritiera. Specificatamente

$$\lim_{P_k^- \rightarrow 0} K_k = 0$$

.

2.3 L'algoritmo del filtro di Kalman lineare

Il FLK, di seguito descritto, si articola in 3 fasi :la fase di inizializzazione,di predizione e di correzione. Le tre fasi interagiscono secondo lo schema mostrato nella figura 1.1.

2.3.1 Fase di inizializzazione

Nella fase di inizializzazione del filtro bisogna inizializzare lo stato a posteriori \hat{x}_0^- e la matrice di covarianza \hat{P}_{k-1} con valori noti:

$$\hat{x}_{k-1} = \hat{x}_0$$

$$\hat{P}_{k-1} = \hat{P}_0$$

2.3.2 Fase di predizione

In questa fase viene predetto lo stato attuale \hat{x}_k^- e la covarianza dell'errore $P_k^- = E[e_k^- e_k^{-T}] = E[(x_k - \hat{x}_k^-)(x_k - \hat{x}_k^-)^T]$ mediante le equazioni:

$$\hat{x}_k^- = A_{k-1}\hat{x}_{k-1} + B_{k-1}u_{k-1}$$

$$P_k^- = A_{k-1}P_{k-1}A_{k-1}^T + Q_{k-1}$$

2.3.3 Fase di correzione

In questa fase viene stimato lo stato attuale \hat{x}_k mediante il vettore delle misure z_k . Una volta calcolata la stima dello stato, il processo viene ripetuto con la precedente stima a posteriori utilizzata per predire la nuova stima a priori, l'output del processo di predizione.

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H_k\hat{x}_k^-)$$

$$P_k = (I - K_k H_k) P_k^-$$

dove:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$$

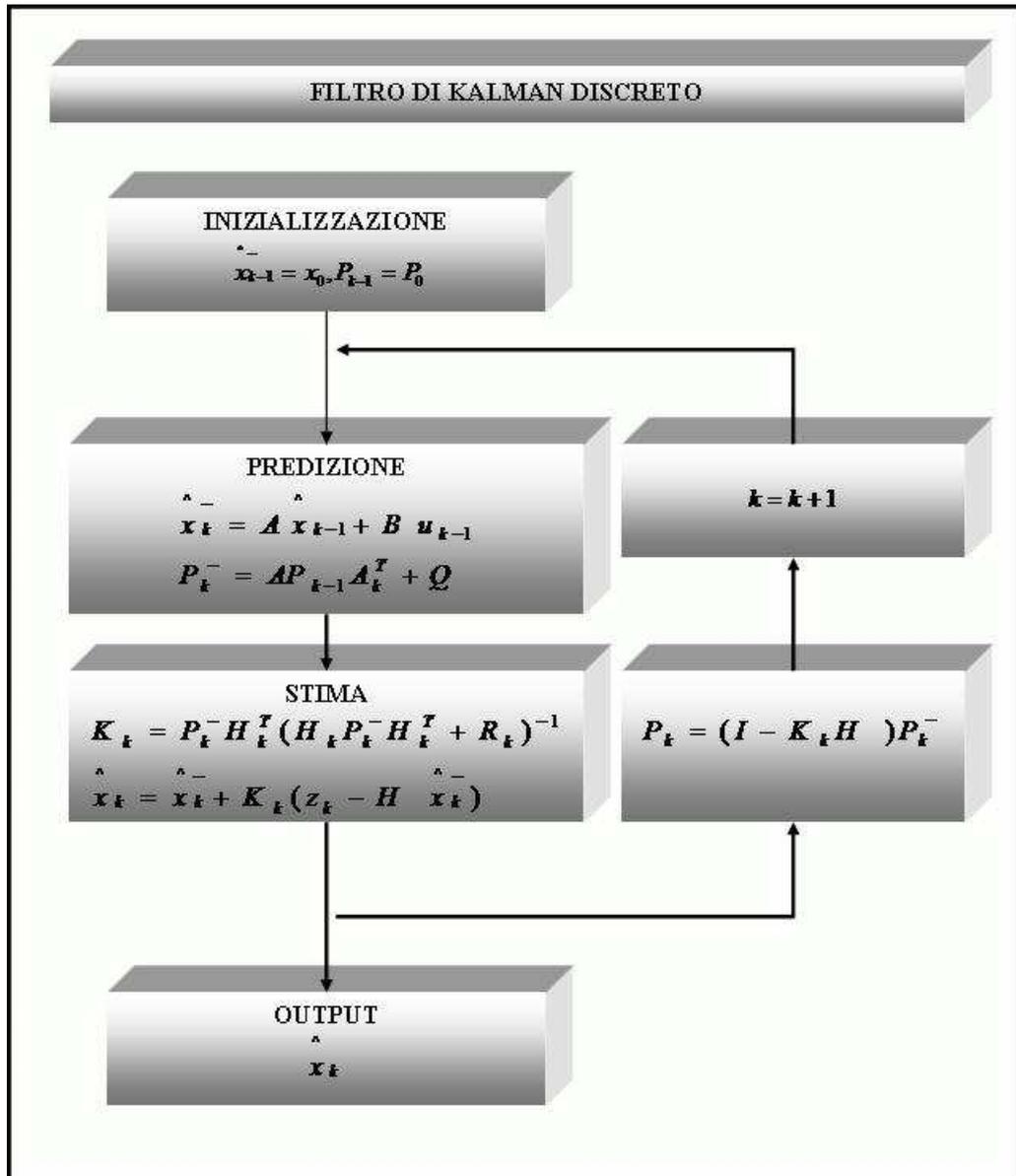


Figura 2.1: L'algoritmo FKL.

2.4 Cenni sulla taratura

I valori assunti dalle matrici R e Q , decisivi per il corretto funzionamento del filtro, sono spesso misurabili, a posteriori, prima di avviare il processo vero e proprio. Questo fatto è ragionevole per quanto riguarda la matrice R , in quanto è in genere possibile effettuare qualche misurazione in un contesto noto e valutare così la distribuzione di probabilità dell'errore. Mentre, per quanto concerne la matrice Q , il discorso risulta essere differente, in quanto l'errore di processo è una quantità meno deterministica che spesso viene introdotta semplicemente per inserire incertezza in modelli poco fedeli alla realtà.

Capitolo 3

Filtro di Kalman esteso

Il filtro di Kalman esteso, spesso abbreviato in letteratura con la sigla FKE (Filtro di Kalman Esteso) rappresenta senza dubbio l'algoritmo più noto per risolvere il problema di filtraggio nonlineare e non-gaussiano. Tale filtro rappresenta l'estensione del filtro di Kalman discreto nel caso di filtraggio non lineare e sfrutta il principio della linearizzazione delle funzioni di evoluzione e di misurazione usando l'espansione in serie di Taylor.

3.1 Principi generali del filtro

Assumendo un generico processo, con lo stato $x \in \mathfrak{R}^n$, descritto dall'equazione non lineare stocastica:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (3.1.1)$$

Ove lo stato attuale x_k è funzione dello stato passato x_{k-1} , del segnale di controllo u_{k-1} e del rumore di processo w_{k-1} . Sia poi:

$$z_k = h(x_k, v_k) \quad (3.1.2)$$

Ove la variabile, $z_k \in \mathfrak{R}^m$, rappresenta il vettore delle misure, funzione dello stato attuale x_k e del rumore di misura v_k . Le variabili casuali w_k e v_k sono assunte a media nulla, bianche, indipendenti e con distribuzione di probabilità normale:

$$p(v) \sim N(0, R)$$

$$p(w) \sim N(0, Q)$$

I rumori aleatori w_k e v_k possono essere trascurati, a patto che si approssimi lo stato del processo con \tilde{x}_k e della misura con \tilde{z}_k quindi le equazioni (3.1.1) e (3.1.2) assumeranno le forme:

$$\tilde{x}_k = f(\hat{x}_{k-1}, u_{k-1}, 0) \quad (3.1.3)$$

$$\tilde{z}_k = h(\tilde{x}_k, 0) \quad (3.1.4)$$

Adesso, andando a linearizzare le funzioni (3.1.1) e (3.1.2) attorno alla stima corrente di stato, mediante l'espansione in serie di Taylor (3.1.5), di primo ordine, perveniamo alle equazioni (3.1.6) e (3.1.7).

$$f(x + \Delta x) = f(x) + f^{(1)}\Delta x + \frac{f^{(2)}}{2!}\Delta x^2 + \dots + \frac{f^{(n)}}{n!}\Delta x^n + \dots \quad (3.1.5)$$

$$x_k = \tilde{x}_k + A_k(x_{k-1} - \hat{x}_{k-1}) + W_k w_{k-1} \quad (3.1.6)$$

$$z_k = \tilde{z}_k + H_k(x_k - \tilde{x}_k) + V_k v_k \quad (3.1.7)$$

In particolare:

- x_k e z_k sono rispettivamente lo stato attuale del processo e lo stato della misura stimata;
- \tilde{x}_k e \tilde{z}_k rappresentano rispettivamente lo stato attuale approssimato del processo e lo stato approssimato della misura stimata;
- \hat{x}_k è lo stato stimato a posteriori;

- w_k e v_k come detto in precedenza rappresentano le variabili casuali;
- A è la matrice Jacobiana delle derivate parziali di $f(\hat{x}_{k-1}, u_k, 0)$ rispetto ad x :

$$A_k = \frac{\partial f(\hat{x}_{k-1}, u_k, 0)}{\partial x}$$

- W è la matrice Jacobiana delle derivate parziali di $f(\hat{x}_{k-1}, u_k, 0)$ rispetto ad w

$$W_k = \frac{\partial f(\hat{x}_{k-1}, u_k, 0)}{\partial w}$$

- H è la matrice Jacobiana delle derivate parziali di $h(\hat{x}_k, 0)$ rispetto ad x :

$$H_k = \frac{\partial h(\hat{x}_k, 0)}{\partial x}$$

- V è la matrice Jacobiana delle derivate parziali di $h(\hat{x}_k, 0)$ rispetto ad v :

$$V_k = \frac{\partial h(\hat{x}_k, 0)}{\partial v}$$

Una volta linearizzate le funzioni possiamo definire l'errore che si commette nella fase di predizione e nella fase di misura a seguito delle approssimazioni introdotte, in particolare:

$$\tilde{e}_{x_k} = x_k - \tilde{x}_k \quad (3.1.8)$$

$$\tilde{e}_{z_k} = z_k - \tilde{z}_k \quad (3.1.9)$$

mediante le espressioni appena definite possiamo riscrivere l'equazioni (3.1.6) e (3.1.7) come segue:

$$\tilde{e}_{x_k} = A_k(x_{k-1} - \hat{x}_{k-1}) + \epsilon_k \quad (3.1.10)$$

$$\tilde{e}_{z_k} = H_k \tilde{e}_x + \eta_k \quad (3.1.11)$$

dove le variabili ϵ_k ed η_k rappresentano le nuove variabili casuali, assunte indipendenti, a media nulla, e rispettivamente, con matrici di covarianza WQW^T e VRV^T .

Le equazioni appena determinate sono lineari, quindi è possibile stimare la predizione dell'errore \tilde{e}_{x_k} mediante l'impiego del FKL. La stima così ottenuta, denominata \hat{e} può essere utilizzata per ottenere la stima dello stato stimato a posteriori \hat{x}_k :

$$\hat{x}_k = \tilde{x}_k + \hat{e}_k \quad (3.1.12)$$

Le variabili casuali \tilde{e}_x , ϵ_k e η_k hanno approssimativamente le seguenti distribuzioni di probabilità:

$$p(\tilde{e}_x) \sim N(0, E[\tilde{e}_{x_k} \tilde{e}_{x_k}^T])$$

$$p(\epsilon_k) \sim N(0, WQ_k W^T)$$

$$p(\eta_k) \sim N(0, VR_k V^T)$$

Il FKL usato per stimare \hat{e}_k risulta:

$$\hat{e}_k = K_k \tilde{e}_{z_k} \quad (3.1.13)$$

Sostituendo quindi la (3.1.13) in (3.1.12) ed esplicitando \tilde{e}_{z_k} possiamo riscrivere \hat{x}_k come segue:

$$\hat{x}_k = \tilde{x}_k + K_k \tilde{e}_{z_k} = \tilde{x}_k + K_k(z_k - \tilde{z}_k) \quad (3.1.14)$$

L'equazione adesso può essere utilizzata nella fase di stima nel filtro di Kalman esteso, con \hat{x}_k lo stato predetto e \tilde{z}_k la stima della misura.

3.2 L'algoritmo del filtro di Kalman esteso

Il FKE, di seguito descritto, si articola in 3 fasi: la fase di inizializzazione, di predizione e di correzione. E' da notare che la variabile \tilde{x}_k è stata sostituita con \hat{x}_k^- mentre la \tilde{z}_k con $h(\hat{x}_k^-, 0)$ per semplicità della trattazione. Nello schema di figura 2.1 è mostrata l'interazione delle tre fasi.

3.2.1 Fase di inizializzazione

Nella fase di inizializzazione del filtro si inizializza lo stato a posteriori \hat{x}_0^- e la matrice di covarianza \hat{P}_{k-1} :

$$\hat{x}_{k-1} = \hat{x}_0$$

$$\hat{P}_{k-1} = \hat{P}_0$$

3.2.2 Fase di predizione

In questa fase viene predetto lo stato attuale \hat{x}_k^- e la covarianza dell'errore P_k^- :

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0)$$

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T$$

3.2.3 Fase di correzione

In questa fase viene stimato lo stato attuale \hat{x}_k mediante il vettore delle misure z_k . Una volta calcolata la stima dello stato, il processo viene ripetuto con la precedente stima a posteriori, utilizzata per predire la nuova stima a priori, l'output del processo di predizione.

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - h(\hat{x}_k^-, 0))$$

$$P_k = (I - K_k H_k) P_k^-$$

dove

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}$$

3.3 Conclusioni

Poiché il filtro di Kalman esteso usa solo i termini del primo ordine, nell'espansione in serie di Taylor, per linearizzare il modello del sistema, spesso introduce grandi errori nelle stime statistiche della distribuzione a posteriori dello stato. Questo fatto è specialmente evidente quando i modelli sono fortemente non lineari: in questo caso l'assunzione di linearità locale viene a mancare dato che i termini di ordine superiore al secondo non sono più trascurabili. Nonostante questo difetto, il filtro di Kalman esteso è largamente diffuso e usato, sia grazie alla sua facilità di implementazione sia soprattutto perché filtri migliori, il filtro di Kalman unscented, filtri particellari, filtri polinomiali, sono stati introdotti in tempi relativamente recenti.

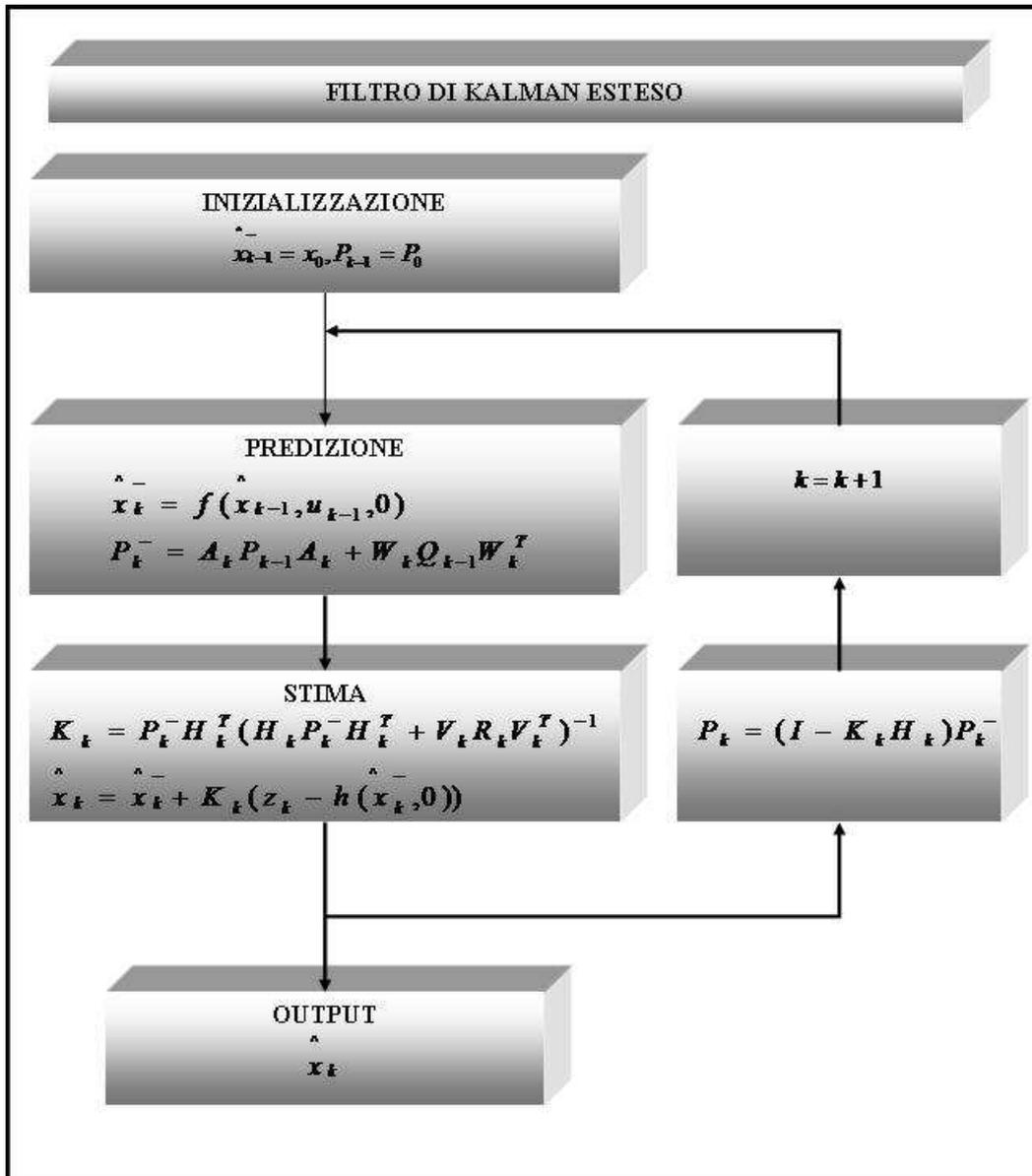


Figura 3.1: Algoritmo di FKE

Capitolo 4

Nomad XR4000

Questo capitolo è dedicato alla descrizione del robot NOMAD XR4000. Dapprima si darà una descrizione generale del robot e successivamente ci cercherà di descrivere in dettaglio la parte software, hardware ed il sistema di guida otonomo.

4.1 Generalità



Figura 4.1: Robot NOMAD XR4000

Il Nomad XR4000, mostrato nella figura, ha un sistema di guida otonomo, cioè è in grado di compiere traslazioni e rotazioni in modo completamente indipendente. Le quattro ruote di cui dispone sono comandate in modo indipendente l'una dall'altra e gli encoder di cui è dotato forniscono informazioni sul moto che il sistema di guida

filtra opportunamente per garantire una maggiore precisione dell'odometria. Inoltre questo robot è equipaggiato con un laser e un bussola digitale.

All'interno del robot è presente un calcolatore nel quale è installato un sistema operativo Linux Red Hat; esso è in grado di accedere ai dati forniti da tutti i sensori del robot tramite alcune librerie sviluppate appositamente per questo scopo (XRDEV).

Il linguaggio di programmazione usato per la loro implementazione è il C.

Infine il calcolatore può comunicare con l'esterno attraverso wireless, cavo ethernet o semplicemente tramite un normale monitor con presa seriale.

4.2 Il sistema oloonomo Nomad XR4000

Il sistema XR C8 Holonomic Drive System è un sistema di guida oloonomo a tre gradi di libertà (x, y, θ) senza restrizioni dovute allo spazio libero di operazione sul terreno, alle vibrazioni o alle complessità meccaniche. Questo risultato è stato raggiunto utilizzando due motori elettricamente indipendenti, uno per la sterzata e l'altro per il moto traslazionale, per ognuna delle quattro ruote che muovono il robot (vedi figura 4.2). L'utilizzo di queste quattro ruote lo rende un sistema ad otto assi sovravincolato. Per controllare questo sistema l'XR4000 utilizza uno speciale controller per il motore dotato di tre DSP e di un microcontrollore a 32-bit dedicato, in modo da controllare e stimare la posizione di questi otto assi.

IL nomad offre due modalità di controllo del movimento: la modalità joint e la modalità global. Nella modalità joint viene definito un sistema di riferimento solidale al robot, quindi tutti gli spostamenti saranno di tipo relativo, mentre nella modalità global viene definito un sistema di riferimento inerziale rispetto al quale il robot può muoversi, quindi tutti gli spostamenti saranno di tipo assoluto.

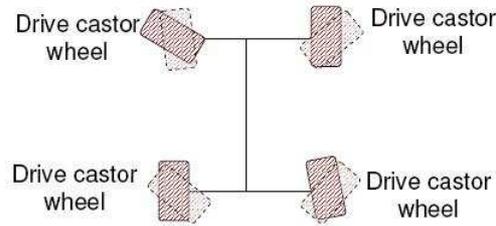


Figura 4.2: Sistema di guida oloonomo.

4.3 Hardware

Il Nomad XR4000, figura 4.3, è costituito da un computer basato su processore Intel Pentium III collegato tramite delle schede di interfacciamento, al sistema sensoriale standard (costituito da sensori sonar, sensori di prossimità ad infrarosso, e sensori tattili) al controller del motore e all'unità di potenza che fornisce alimentazione a tutte le apparecchiature. La strumentazione sensoriale del robot è inoltre completata da una bussola digitale e da un sistema laser di seguito descritti in quanto utilizzati come sensori esteroceettivi nella localizzazione .

4.3.1 Sistema Sensus 600 di orientamento

Il sistema Sensus 600 è basato su una bussola digitale KVH-C100 che è un componente industriale di grande affidabilità, compattezza e precisione. È in grado di misurare i decimi di gradi ed ha una accuratezza di 0.5° anche a latitudini vicine ai poli; anche grazie alle sue generose temperature di esercizio, da $-25C^\circ$ a $70C^\circ$, è stato impiegato in applicazioni critiche (come robot sottomarini) e in ambito militare.

Il KVH-C100 si basa su una bussola a flussometro digitale capace di misure molto precise anche se montata non in orizzontale, o in ambienti con forti disturbi elettro-

magnetici. La bussola è montata esternamente in un contenitore metallico blindato e collegata alla scheda di interfacciamento *Nomadic compass interface* da cui, tramite una piattina, è connessa alla seconda porta seriale della scheda madre. Il sistema è configurato per comunicare a 9600 baud, 8 bit di dati, nessun bit di parità e 1 bit di stop.

4.3.2 Sistema laser Sick Sensus 550

Il Sensus 550, figura , è un sistema di telemetria basato sul sensore Sick elettro-ottico LMS-200. Esso fornisce 180° di rilevamento planare ad incrementi di 0.5°, complessivamente 360 campioni. Ogni piano completo di rilevazione avviene con frequenza 20Hz.

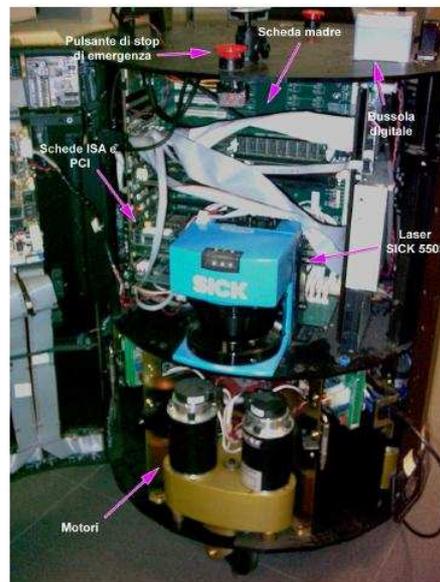


Figura 4.3: L'interno del NOMAD XR4000

4.4 L'architettura XRDev

XRDev è una architettura multi processo destinata alle applicazioni robotiche composta dalle seguenti librerie ed eseguibili:

- DRobot - È il demone caricato all'avvio del robot, addetto all'ascolto dei comandi impartiti dall'utente, i quali vengono poi diretti all'hardware presente sul Nomad. È possibile estendere questo programma, in modo che l'utente possa aggiungere altre caratteristiche, senza avere il codice sorgente.
- Nhostclient.a - È la libreria di funzioni utilizzate da un programma client per comandare un robot da qualsiasi computer sulla rete. Per utilizzare questa caratteristica è sufficiente che l'utente, nella compilazione del programma, lo linki a questa libreria, ed esegua il programma normalmente da qualsiasi macchina sulla rete; la libreria si prenderà cura della comunicazione attraverso la rete, per raggiungere e impartire comandi al robot.
- Ngui - E' l'interfaccia grafica che fornisce una rappresentazione grafica dell'attività del robot e dei dati dei sensori, nonché permette all'utente di controllare il robot (tramite un joystick a schermo oppure con semplici comandi).

Una applicazione XRDev è composta da differenti processi: i processi del robot, dell'utente, i processi di interfaccia, etc., che comunicano attraverso la rete. Non c'è un limite (se non quello imposto dall'efficienza e dal carico sulla rete e sulla CPU) al numero di robot che possono essere controllati con XRDev e non c'è un limite al numero di programmi che li controllano. Inoltre ogni programma, che può controllare più robot contemporaneamente, può essere eseguito ovunque sulla rete.

Capitolo 5

FKE applicato al NOMAD XR4000

In questo capitolo verrà mostrato come il FKE viene applicato nella localizzazione del robot mobile NOMAD XR4000. Innanzitutto verranno definiti il modello cinematico ed il modello dell'osservatore. Il primo descrive il movimento del robot e viene usato nella fase di predizione, mentre il secondo analizza i valori misurati dai sensori esterocezionali in funzione di un ipotetico stato e viene adoperato nella fase di stima. Successivamente verrà mostrato come vengono adattate le equazioni del FKE al problema della localizzazione (vedi capitolo 3). Infine il paragrafo finale è destinato allo studio delle simulazioni.

5.1 Il modello cinematico

La posizione che il robot occupa all'istante k , in un sistema di riferimento inerziale, può essere rappresentata mediante il vettore:

$$S_k = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} \quad (5.1.1)$$

con $S \in \mathbb{R}^3$.

Mentre il modello cinematico che descrive lo spostamento del robot è descritto da equazioni non lineari stocastiche:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \end{bmatrix} + \Delta T \begin{bmatrix} V_{k-1} \cos(\theta_k) \\ V_{k-1} \sin(\theta_k) \\ \omega_{k-1} \end{bmatrix} \quad (5.1.2)$$

In particolare:

- Le variabili x_k , y_k e θ_k definiscono la configurazione che il robot assume all'istante k in un sistema di riferimento inerziale (vedi figura 5.1). Si noti inoltre che θ_k è relativo all'asse X positivo.
- Le variabili x_{k-1} , y_{k-1} e θ_{k-1} sono relative alla configurazione passata.
- ΔT è semplicemente l'intervallo di campionamento.
- V_{k-1} e ω_{k-1} sono gli ingressi del sistema in questione e rappresentano rispettivamente la velocità lineare ed angolare. Quindi definendo con u_{k-1} il vettore degli ingressi si otterrà

$$u_{k-1} = [V_{k-1} \ \omega_{k-1}]$$

Al vettore u_k è associato il vettore dei rumori $\delta u_{k-1} = [\delta V_{k-1} \ \delta \omega_{k-1}]$ la cui media si suppone essere costante. Si noti inoltre che tale rumore si ipotizza essere scorrelato e di natura gaussiana, con varianze costanti σ_V^2 e σ_ω^2 . E' possibile quindi riscrivere le componenti del vettore u_{k-1} :

$$V_{k-1} = \bar{V}_{k-1} + \delta V_{k-1} \quad (5.1.3)$$

$$\omega_{k-1} = \bar{\omega}_{k-1} + \delta \omega_{k-1} \quad (5.1.4)$$

Come detto in precedenza il modello cinematico viene adoperato nella fase di predizione, e restituisce la posizione stimata mediante l'uso dell'odometria. Denotando quindi con ΔL e $\Delta \theta$ rispettivamente la posizione e l'angolo restituiti dagli encoder e riconsiderando la (5.1.2) è possibile fare le seguenti posizioni:

$$\Delta L = V_{k-1} \Delta T$$

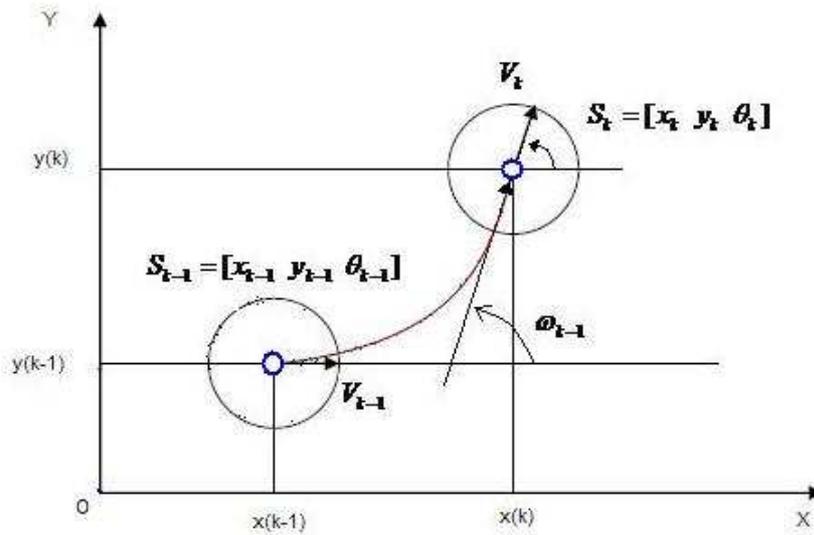


Figura 5.1: Spostamento del robot.

$$\Delta\theta = V_{k-1}\Delta T$$

Quindi le espressioni $V_{k-1}\Delta T \cos(\theta_k)$ e $V_{k-1}\Delta T \sin(\theta_k)$ rappresentano rispettivamente Δx e Δy .

5.2 Il modello dell'osservatore

Il FKE nella fase di correzione confronta le misure reali fornite dai sensori esterocezionali con le misure stimate dal modello dell'osservatore. Quindi, avendo adoperato come sensori esterocezionali la bussola digitale ed il laser, il modello dell'osservatore sarà descritto mediante le equazioni alle differenze:

$$z_k = h(S_k, 0) + v_k \tag{5.2.1}$$

Dove la variabile casuale v_k è assunta a media nulla, bianca, indipendente e con distribuzione di probabilità normale :

$$p(v) \sim N(0, Q)$$

Mentre $h(S_k, p_t)$ è una funzione non lineare avente, nel caso in esame la forma:

$$h(S_k, 0) = \begin{pmatrix} d_k \\ \alpha_k \end{pmatrix} = \begin{pmatrix} \sqrt{(p_x - x_k + A \cos(\theta_k))^2 + (p_y - y_k + A \sin(\theta_k))^2} \\ \theta_k \end{pmatrix} \quad (5.2.2)$$

In particolare:

- d_k rappresenta la distanza stimata tra il robot, situato in una ipotetica configurazione, e il punto dell'ambiente che sarebbe intercettato dal laser durante un ipotetico rilevamento relativo alla suddetta configurazione;
- A , invece, come mostrato nella figura 5.2, rappresenta la distanza del laser dal centro del robot;
- α_k rappresenta l'angolo stimato che il robot assume all'istante k . La semplicità dell'espressione è dovuta essenzialmente alla scelta, come dispositivo esterocettivo, della bussola digitale.

Essendo la bussola preposta a misurare il campo magnetico terrestre, essa non ha bisogno di ulteriori riscontri esterni;

- Infine p_t è un vettore di componenti p_x e p_y , tali componenti rappresentano (vedi figura 5.2) la stima del punto dell'ambiente rettangolare che il raggio laser colpirebbe se il robot si trovasse in quella posizione;

Di seguito, per completezza della trattazione, vengono mostrate le fasi che portano al calcolo del vettore p_t , a tal proposito risulta utile tener presente la figura 5.3:

1. Per ogni k si calcolano gli angoli delle quattro rette che congiungono il centro del robot di coordinate (x_k, y_k) con i quattro vertici del rettangolo (X_i, Y_i) con $i \in 0, 1$ come mostrato in figura, tali angoli sono calcolati rispetto all'asse X del sistema di riferimento inerziale.

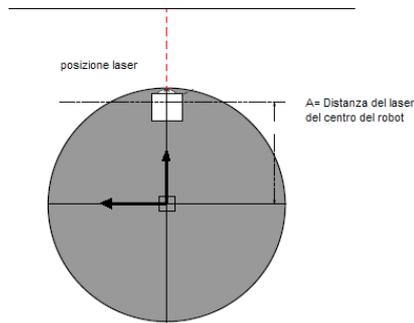


Figura 5.2: Distanza del laser dal centro del robot

2. Una volta noti gli angoli delle quattro rette, possiamo mediante una semplice operazione di confronto determinare a quale quadrante appartiene la retta del laser.
3. Una volta individuato il quadrante, conosciamo anche il segmento del quadrilatero che il raggio laser dovrebbe colpire se il robot si trovasse in quella configurazione.
4. Ricordando che l'equazione della retta è descritta da $y = mx + q$ dove m è il coefficiente angolare e q il termine noto, possiamo mediante il sistema di equazioni (5.2.3) calcolare il punto d'intersezione delle due rette, quindi il punto della parete che dovrebbe essere colpito dal raggio laser.

$$\begin{aligned} y_r &= m_r x_r + q_r \\ y_f &= m_f x_f + q_f \end{aligned} \quad (5.2.3)$$

dove i pedici r ed f si riferiscono rispettivamente alla retta che rappresenta il laser e alle rette che compongono il quadrilatero. Essendo il quadrilatero composto da due segmenti verticali e due orizzontali l'espressione (5.2.3) si semplifica notevolmente.

Specificatamente per segmenti orizzontali avremo

$$\begin{aligned} p_y &= y_r - y_f \\ p_x &= \frac{p_y}{m} \end{aligned} \quad (5.2.4)$$

mentre per i segmenti verticali

$$\begin{aligned} p_x &= x_r - x_f \\ p_y &= p_x m \end{aligned} \quad (5.2.5)$$

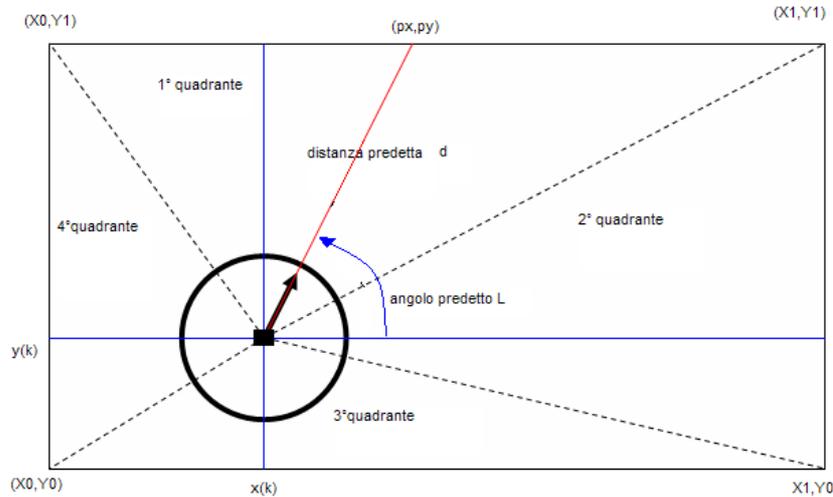


Figura 5.3: Stime del vettore p_t .

Una volta definiti i modelli sui cui andrà ad operare il FKE possiamo finalmente esplicitare tutti i termini di cui è composto tale filtro. Quindi il prossimo paragrafo risulta essere un'applicazione di tutta la teoria vista in precedenza ad un caso reale quale la localizzazione del NOMAD XR4000.

5.3 Localizzazione mediante il FKE

In questo paragrafo verrà mostrato come le equazioni del filtro di Kalman esteso vengono adattate al problema della localizzazione del robot NOMAD XR4000.

5.3.1 Fase di inizializzazione

In questa fase viene inizializzato il vettore \widehat{S}_{k-1} e le matrici di covarianza P , Q ed R . Quasi sempre il vettore \widehat{S}_{k-1} è inizializzato con valori noti mentre per quanto concerne le matrici P , Q ed R i valori iniziali possono essere solo stimati. Infine possiamo esplicitare gli elementi delle matrici:

$$P = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} \quad Q = \begin{bmatrix} \sigma_V^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix} \quad R = \begin{bmatrix} \sigma_d^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix} \quad (5.3.1)$$

dove per l'ipotesi di indipendenza gli elementi fuori dalle diagonali sono posti a 0.

5.3.2 Fase di predizione

In questa fase, mediante le equazioni (5.3.4) e (5.3.3) viene predetta rispettivamente la posizione \widehat{S}_{k-1} e la matrice di covarianza P_k^- . Per il calcolo del vettore \widehat{S}_{k-1} si procede nel seguente modo:

- Si esplicita il vettore dei rumori del sistema (5.1.2):

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} + \Delta T \begin{bmatrix} (\overline{V}_k + \delta V_k) \cos(\theta_k) \\ (\overline{V}_k + \delta V_k) \sin(\theta_k) \\ \overline{\omega}_k + \delta \omega_k \end{bmatrix} \quad (5.3.2)$$

- Si trascura il vettore dei rumori e si approssima lo stato S_k con il vettore

$$\widehat{S}_k: \quad \begin{bmatrix} \widehat{x}_{k+1}^- \\ \widehat{y}_{k+1}^- \\ \widehat{\theta}_{k+1}^- \end{bmatrix} = \begin{bmatrix} \widehat{x}_k \\ \widehat{y}_k \\ \widehat{\theta}_k \end{bmatrix} + \Delta T \begin{bmatrix} \overline{V}_k \cos(\theta_k) \\ \overline{V}_k \sin(\theta_k) \\ \overline{\omega}_k \end{bmatrix} \quad (5.3.3)$$

Mentre per quanto concerne la covarianza ‘a priori’ essa viene calcolata mediante l'equazione:

$$P_k^- = \nabla f_{S_k} P_{k-1} \nabla f_{S_k} + \nabla f_{w_k} Q_{k-1} \nabla f_{w_k} \quad (5.3.4)$$

Confrontando l'espressione appena trovata con $P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T$ è possibile identificare ∇f_{S_k} come la matrice Jacobiana delle derivate parziali rispetto allo stato (la matrice A_k) mentre ∇f_{w_k} è la matrice Jacobiana delle derivate parziali rispetto al vettore dei rumori (la matrice W_k).

Tenendo presente l'espressione (5.3.2) possiamo esplicitare ∇f_{S_k} e ∇f_{w_k} :

5. La matrice Jacobina delle derivate parziali del sistema $f(\bullet, \bullet)$ rispetto allo stato diventa:

$$\nabla f_{S_k} = \frac{\partial f(S)}{\partial S} \Big|_{S=\hat{S}_{k-1}, w=\overline{w}_{k-1}} = \begin{bmatrix} \frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} & \frac{\partial f_x}{\partial \theta} \\ \frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y} & \frac{\partial f_y}{\partial \theta} \\ \frac{\partial f_\theta}{\partial x} & \frac{\partial f_\theta}{\partial y} & \frac{\partial f_\theta}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta TV_k \sin(\theta_k) \\ 0 & 1 & \Delta TV_k \cos(\theta_k) \\ 0 & 0 & 1 \end{bmatrix} \quad (5.3.5)$$

6. La matrice Jacobina delle derivate parziali del sistema $f(\bullet, \bullet)$ rispetto al rumore risulta:

$$\nabla f_{w_k} = \frac{\partial f(S)}{\partial w} \Big|_{S=\hat{S}_{k-1}, w=\overline{w}_{k-1}} = \begin{bmatrix} \frac{\partial f_x}{\partial V} & \frac{\partial f_x}{\partial \varphi} \\ \frac{\partial f_y}{\partial V} & \frac{\partial f_y}{\partial \omega} \\ \frac{\partial f_\theta}{\partial V} & \frac{\partial f_\theta}{\partial \omega} \end{bmatrix} = \begin{bmatrix} \cos(\theta_k) & 0 \\ \sin(\theta_k) & 0 \\ 0 & 1 \end{bmatrix} \quad (5.3.6)$$

5.3.3 Fase di correzione

In questa fase dell'algoritmo la posizione predetta viene aggiornata mediante l'ausilio dei sensori esteroceettivi dall'equazione $\hat{S}_k = \hat{S}_k^- + K_k(z_k - h(\hat{S}_k^-, 0))$, dove:

- $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$ è il guadagno di Kalman;
- mentre H_k è la matrice Jacobina delle derivate parziali del modello dell'osservatore $h(\hat{S}_k^-)$ rispetto allo stato S_k valutato per $S_k = \hat{S}_k^-$ risulta :

$$H_k = \frac{\partial h(S)}{\partial S} \Big|_{S=\hat{S}_k^-} = \begin{bmatrix} \frac{\partial d}{\partial x} & \frac{\partial d}{\partial y} & \frac{\partial d}{\partial \theta} \\ \frac{\partial \alpha}{\partial x} & \frac{\partial \alpha}{\partial y} & \frac{\partial \alpha}{\partial \theta} \end{bmatrix} \quad (5.3.7)$$

Dove:

$$h_{11} = \frac{\partial d}{\partial x} = \frac{-p_x + x_k - A\cos(\theta_k)}{\sqrt{(p_x - x_k + A\cos(\theta_k))^2 + (p_y - y_k + A\sin(\theta_k))^2}}$$

$$h_{12} = \frac{\partial d}{\partial y} = \frac{-p_y + y_k + A\sin(\theta_k)}{\sqrt{(p_x - x_k + A\cos(\theta_k))^2 + (p_y - y_k + A\sin(\theta_k))^2}}$$

$$h_{13} = \frac{\partial d}{\partial \theta} = A \frac{(p_x - x_k)\sin(\theta_k) + (p_y - y_k)\cos(\theta_k)}{\sqrt{(p_x - x_k + A\cos(\theta_k))^2 + (p_y - y_k + A\sin(\theta_k))^2}}$$

$$h_{21} = \frac{\partial \alpha}{\partial x} = 0$$

$$h_{22} = \frac{\partial \alpha}{\partial y} = 0$$

$$h_{23} = \frac{\partial \alpha}{\partial \theta} = 1$$

Una volta effettuata la correzione sulla posizione l'algoritmo viene ripetuto, come mostrato nello schema della figura 5.4. La posizione appena stimata viene utilizzata per predire la nuova posizione a priori mentre la matrice di covarianza dello stato viene utilizzata per predire la nuova matrice di covarianza a priori mediante l'espressione $P_k = (I - K_k H_k) P_k^-$.

5.4 Simulazioni

Per comprendere appieno le potenzialità derivanti dall'applicazione del FKE si ha la necessità di confrontare la traiettoria stimata con quella reale ma ciò risulta impossibile in un contesto sperimentale in quanto, per conoscere la traiettoria reale, bisogna impiegare tecniche particolari. Un modo pratico e veloce per aggirare tale ostacolo

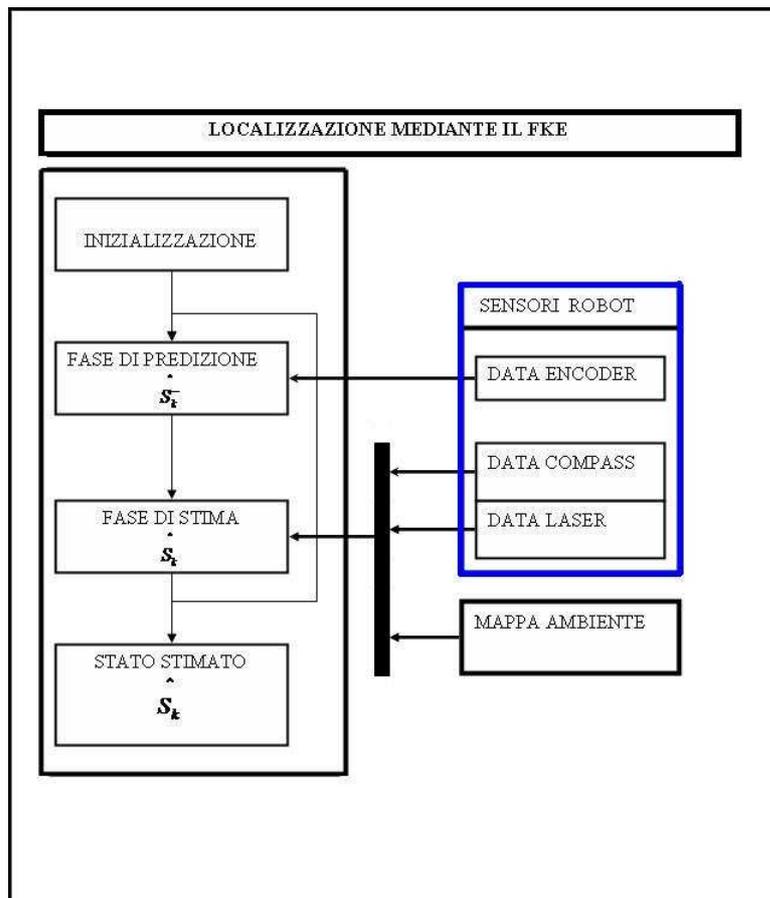


Figura 5.4: Localizzazione mediante l'applicazione del FKE

consiste nel simulare la dinamica del robot reale al calcolatore mediante l'aggiunta del rumore gaussiano a media nulla in tutte le misure effettuate. Inoltre nelle misure odometriche è stato messo in conto l'errore sistematico tipico del robot reale.

5.4.1 I risultati delle prove

Di seguito vengono riportati alcuni grafici che mostrano il comportamento del filtro di Kalman esteso in alcuni casi significativi.

In tutte le figure, di seguito mostrate vengono utilizzate le seguenti convenzioni:

- Le linee blu rappresentano le variabili reali.
- Le linee rosse rappresentano a seconda del grafico le stime odometriche o le stime mediate il FKE.
- Le linee verdi rappresentano le misure dei sensori esteroceettivi.

Iniziamo col considerare le figure 5.5, 5.6, 5.7 e 5.8 le quali si riferiscono ad una rotazione di 45 gradi in senso antiorario, in dettaglio:

- Nella figura 5.5 viene mostrata la stima dell'angolo mediante misure odometriche e l'angolo reale. In particolare possiamo notare come l'angolo desiderato diverge progressivamente nel tempo dall'angolo reale, tale divergenza è dipendente dell'azione combinata dell'errore sistematico con il rumore gaussiano.
- Nelle figure 5.6 e 5.7 viene mostrata la stima dell'angolo mediante il filtro di Kalman esteso, in più nella figura 5.7 vengono mostrate le misure della bussola digitale.
- La figura 5.8 è semplicemente un ingrandimento della 5.7 reso indispensabile a causa dell'elevato numero dei cicli di stima che non permettono di evidenziare la stima dell'angolo mediate il FKE nelle figure 5.6 e 5.7. In essa è possibile

notare come l'angolo stimato insegue molto fedelmente l'angolo reale anche in presenza delle misure rumorose della bussola digitale.

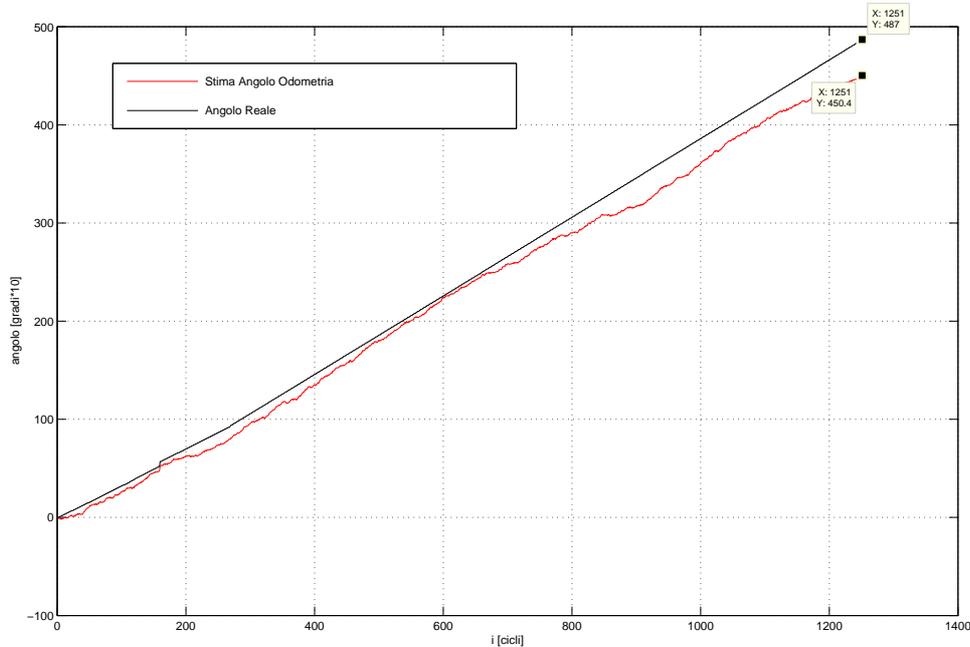


Figura 5.5: Stima dell'angolo mediante l'odometria.

Nelle figure seguenti vengono mostrate le simulazioni inerenti ad una traslazione che porta il robot dalla posizione iniziale di coordinate (0 m,0 m) nella posizione finale (0.8 m,0.4 m). In particolare le figure 5.9, 5.10, 5.11 e 5.12 si riferiscono alla proiezione della traiettoria sull'asse X mentre le figure 5.13, 5.14, 5.15 e 5.16 si riferiscono alla proiezione della traiettoria sull'asse Y. Iniziamo col considerare in dettaglio le figure 5.9, 5.10, 5.11 e 5.12:

- Nella figura 5.9 viene mostrata la stima della coordinata X mediante misure odometriche e la coordinata X reale. In particolare possiamo notare come la X desiderata diverge progressivamente nel tempo dalla X reale. Tale divergenza è dipendente dall'azione combinata dell'errore sistematico con il rumore gaussiano.

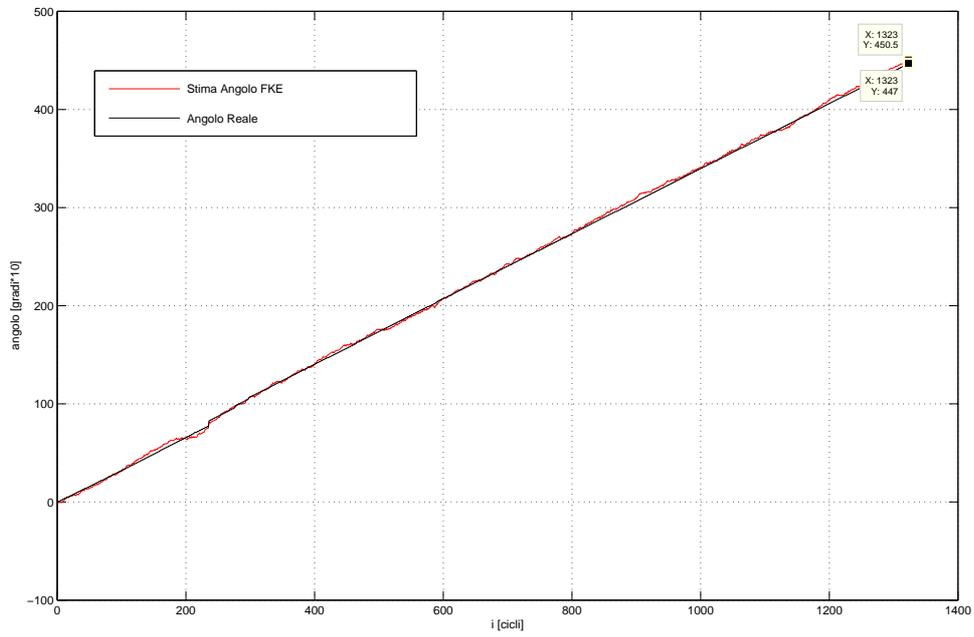


Figura 5.6: Stima dell'angolo mediante il FKE.

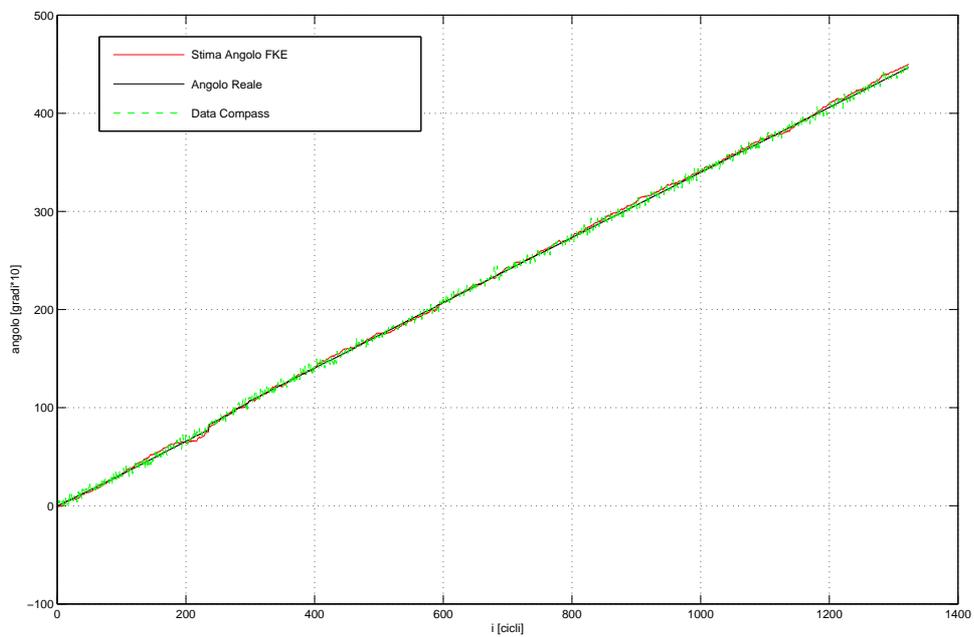


Figura 5.7: La figura 5.6 con le misure della bussola digitale.

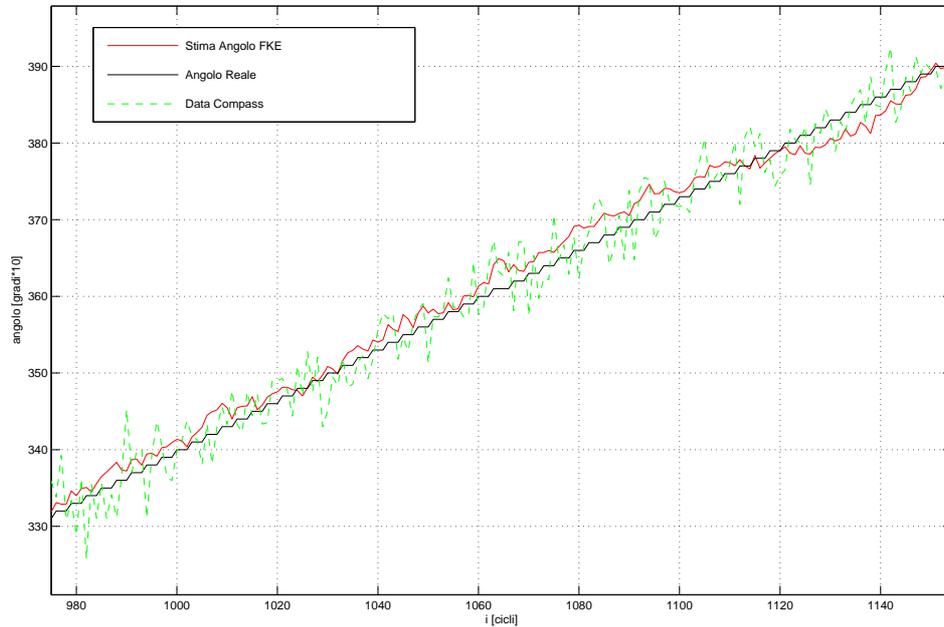


Figura 5.8: Ingrandimento figura 5.7

- Nelle figure 5.10 e 5.11 viene mostrata la stima della coordinata X mediante l'uso del filtro di Kalman esteso, in più nella figura 5.11 vengono mostrate le misure del raggio laser proiettate sull'asse X .
- Nella figura 5.12 viene invece mostrato un ingrandimento della 5.11 reso indispensabile a causa dell'elevato numero dei cicli di stima che non permettono di evidenziare la stima della coordinata X mediante il FKE nelle figure 5.10 e 5.11. In essa è possibile notare come la coordinata X stimata insegue molto fedelmente X reale grazie soprattutto alle misure poco rumorose del raggio laser.

Infine:

- Nella figura 5.13 viene mostrata la stima della coordinata Y mediante misure odometriche e la coordinata Y reale. In particolare possiamo notare come la Y

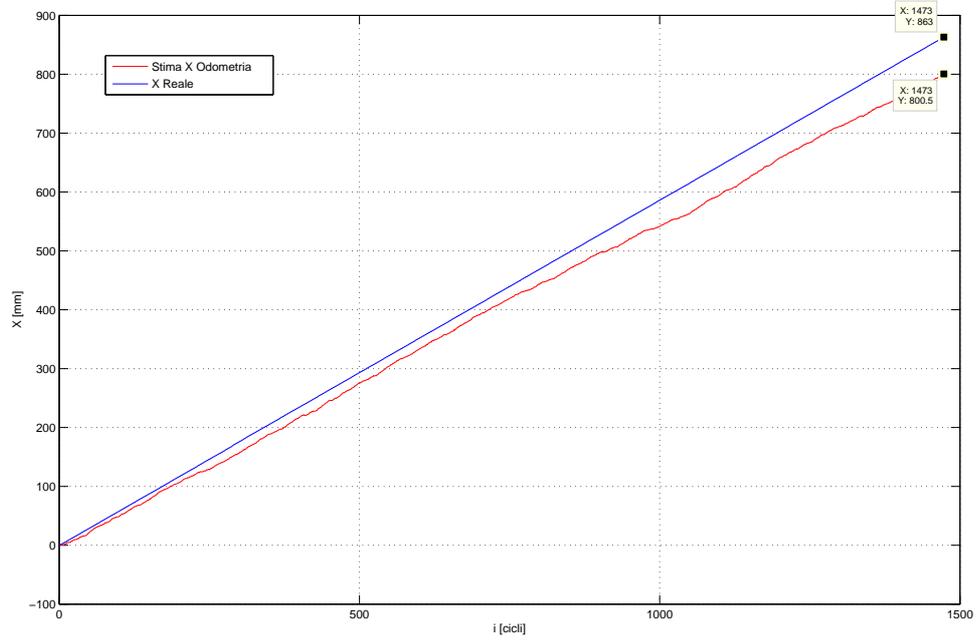


Figura 5.9: Stima della coordinata X mediante l'odometria.

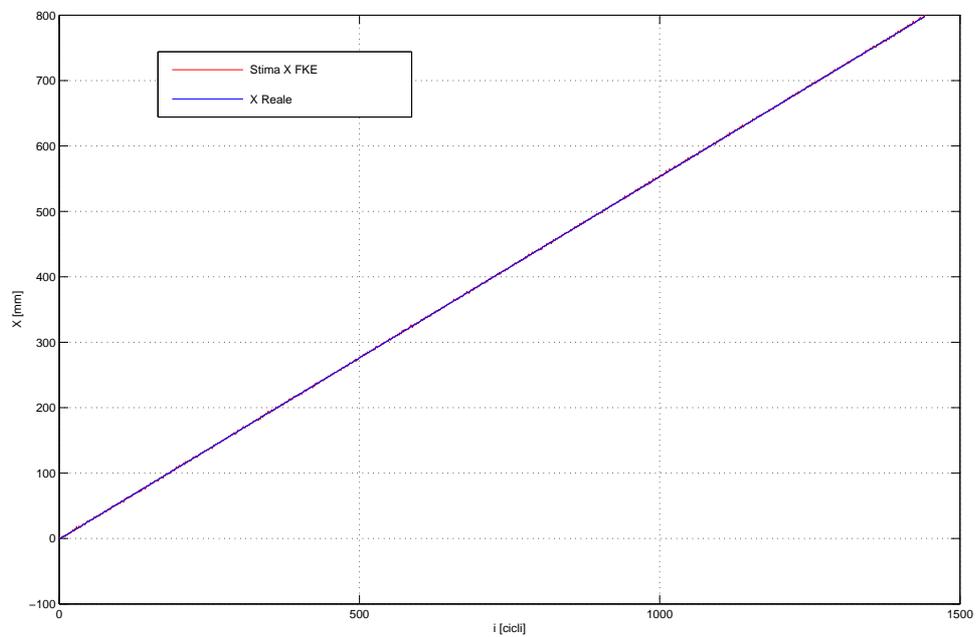


Figura 5.10: Stima della coordinata X mediante il FKE.

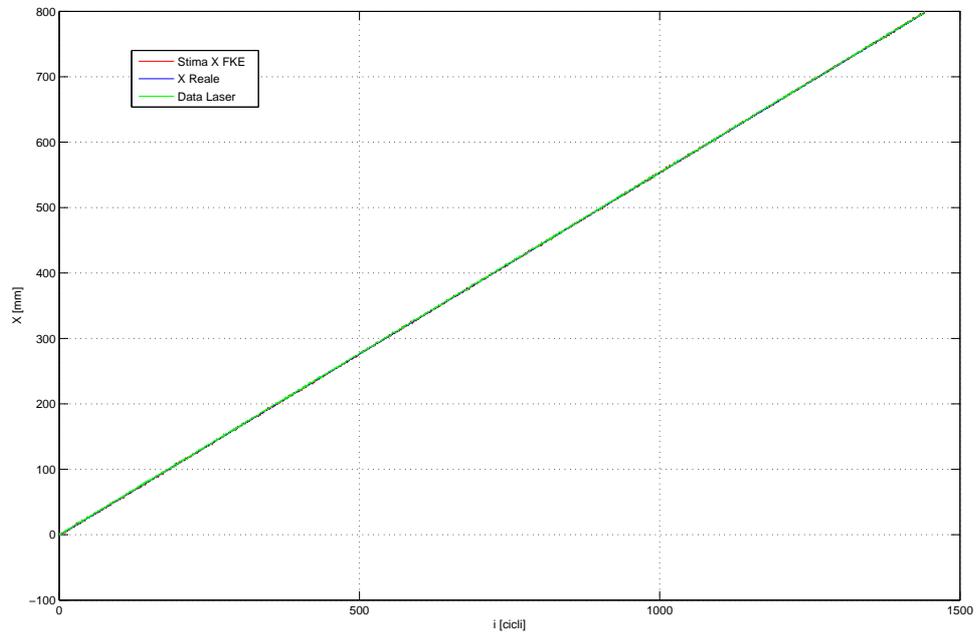


Figura 5.11: La figura 5.10 con le misure laser.

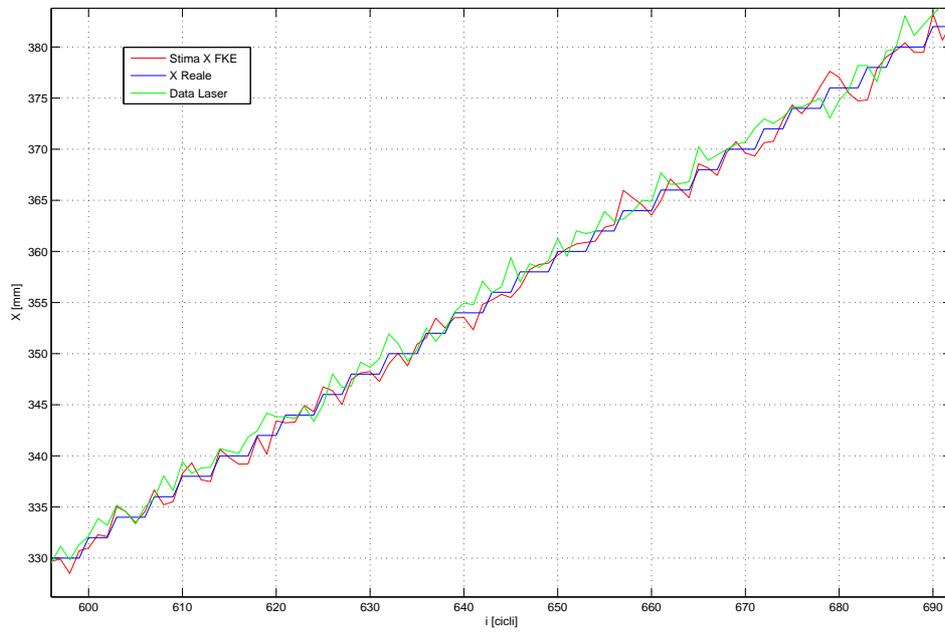


Figura 5.12: Ingrandimento figura 5.11.

desiderata diverge progressivamente nel tempo dalla Y reale. tale divergenza è dipendente dell'azione combinata dell'errore sistematico con il rumore gaussiano.

- Nelle figure 5.14 e 5.15 viene mostrata la stima della coordinata Y mediante l'uso del filtro di Kalman esteso, in più nella figura 5.15 vengono mostrate le misure del raggio laser proiettate sull'asse Y .
- Nella figura 5.16 viene invece mostrato un ingrandimento della 5.15 reso indispensabile a causa dell'elevato numero dei cicli di stima che non permettono di evidenziare la stima della coordinata Y mediante il FKE nelle figure 5.14 e 5.15. In essa è possibile notare come la coordinata Y inseguie molto fedelmente la Y reale grazie soprattutto alle misure poco rumorose del raggio laser.

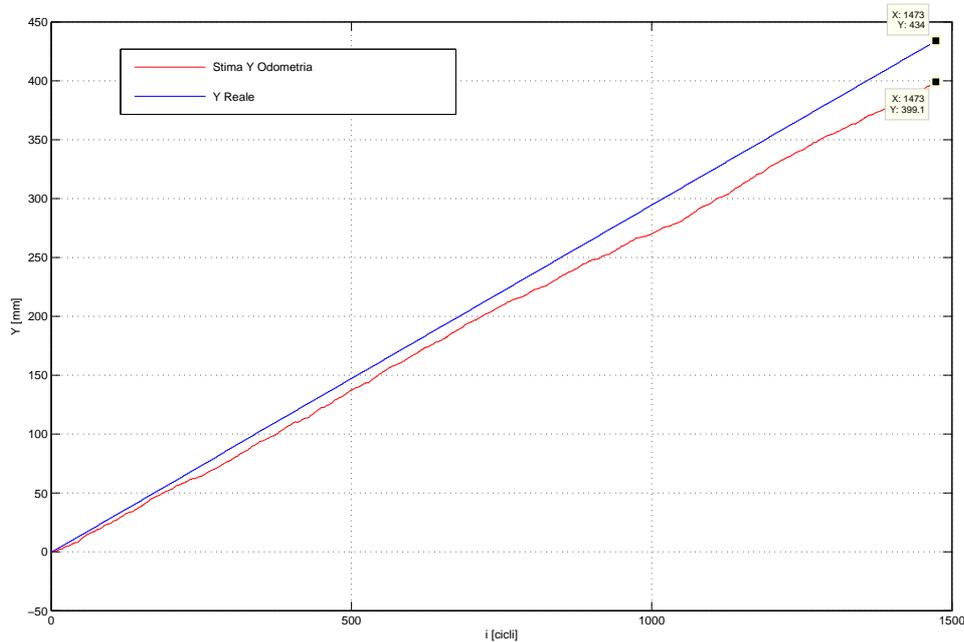


Figura 5.13: Stima della coordinata Y mediante l'odometria.

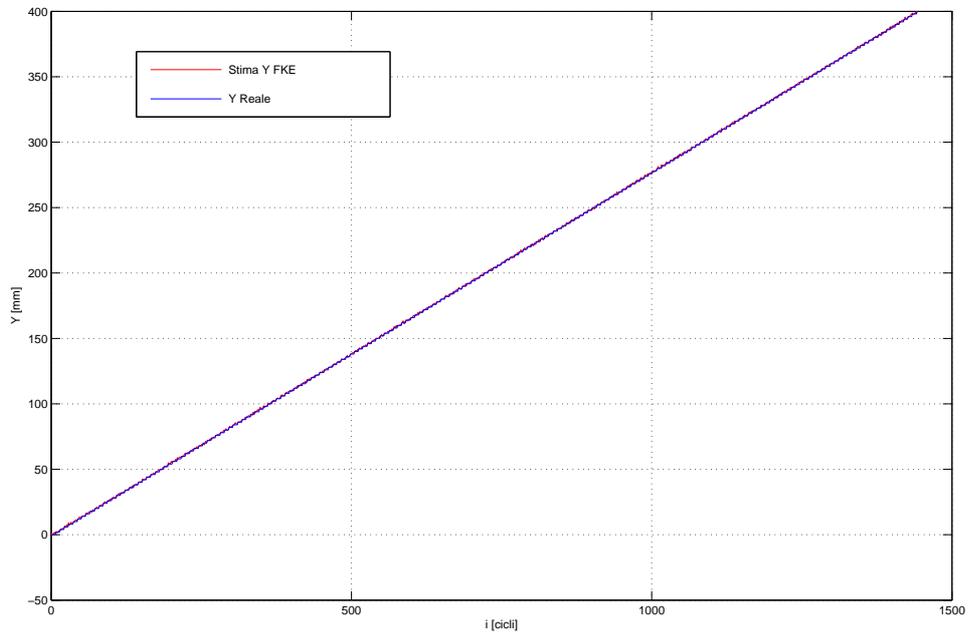


Figura 5.14: Stima della coordinata Y mediante il FKE.

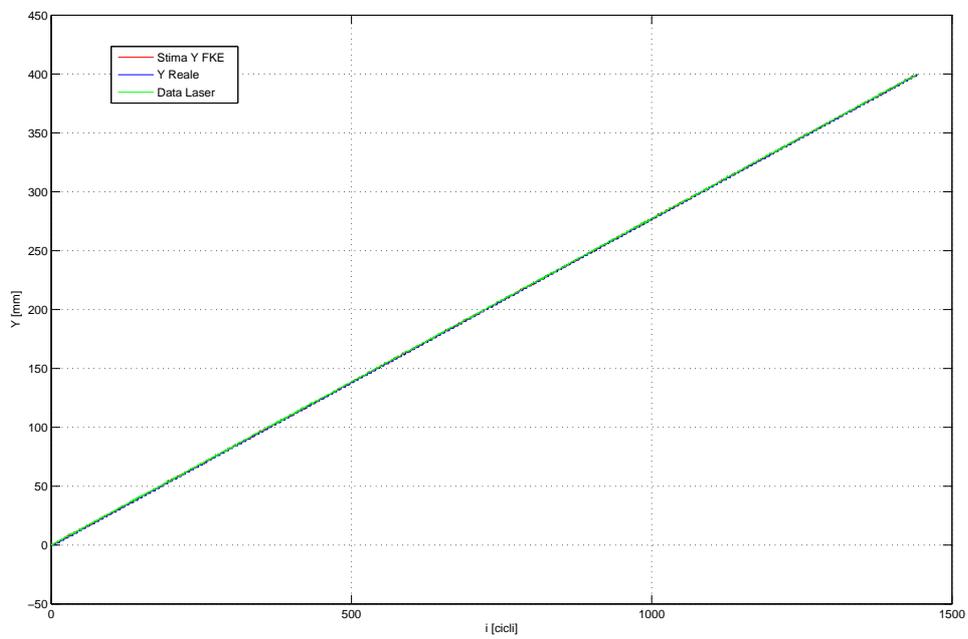


Figura 5.15: La figura 5.14 con le misure laser.

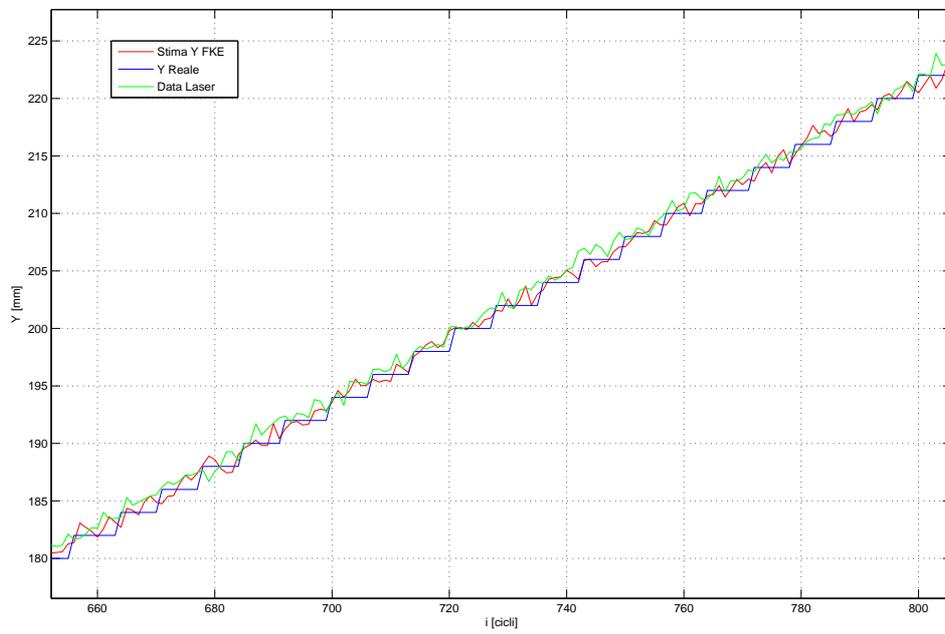


Figura 5.16: Ingrandimento figura 5.15

Capitolo 6

Conclusioni e sviluppi futuri

In questo lavoro è stato presentato ed affrontato il problema della localizzazione mediante il filtro di Kalman esteso del robot NOMAD XR4000. Dai risultati ottenuti nel precedente capitolo possiamo concludere che l'obiettivo proposto all'inizio della tesi è stato raggiunto appieno ossia il filtro di kalman esteso si è dimostrato una soluzione efficiente che risolve il problema della localizzazione. Tale risultato in prevalenza dev'essere attribuito alla scelta dei sensori esteroceettivi, in particolare avendo scelto come sensore esteroceettivo proposto alle misure angolari la bussola digitale e come sensore esteroceettivo proposto alle misure metriche il laser abbiamo ottenuto delle misure affidabili e molto precise specialmente per l'angolo che ha un notevole impatto sulla localizzazione del robot.

6.0.2 Sviluppi futuri

L'obiettivo preposto in questa tesi riguarda lo sviluppo di uno dei mattoni della robotica mobile, fondamentale per poter sviluppare applicazioni robotiche complesse. Quindi avendo risolto il problema della localizzazione gli sviluppi futuri potrebbero essere:

- Lo studio delle traiettorie complesse;

- Simultanea localizzazione e costruzione di mappe per squadre di robot mobili cooperanti;
- Il montaggio del braccio robotico SCORBOT sulla piattaforma del NOMAD XR4000 estendendo in questo modo lo spazio di lavoro da 2D in 3D. Un applicazione di tale fusione potrebbe ad esempio rimuovere gli ostacoli imprevisti o previsti dalla traiettoria.

Appendice A

Simulatore

Come simulatore è stato utilizzato il programma Nserver, sviluppato dalla Nomadic. Nserver è un programma che genera un'interfaccia grafica elaborata, vedi figura 6.1, che permette di vedere i dati provenienti dai sensori, la posizione del robot e la mappa dello spazio in cui si muove il robot.

Per poter eseguire i programmi, scritti in C, sul simulatore, è indispensabile inclu-

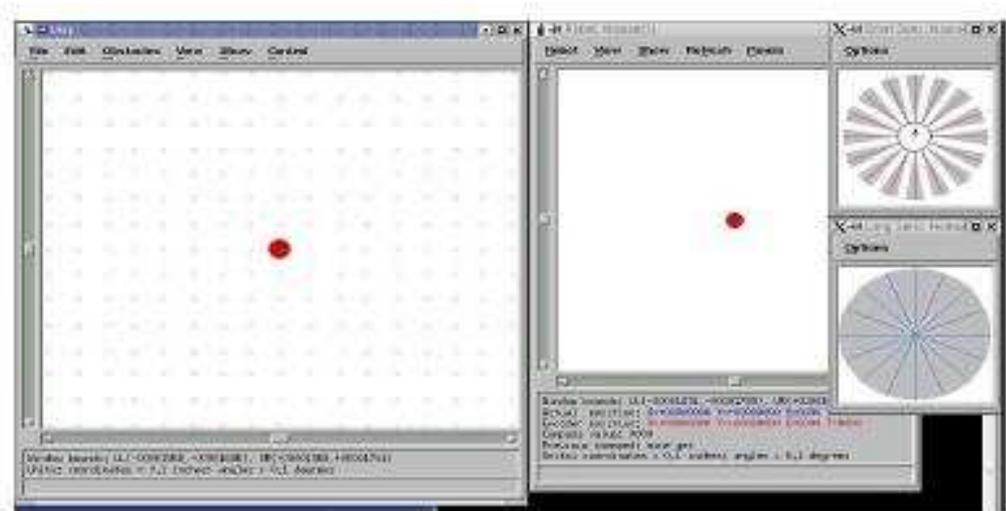


Figura 6.1: L'interfaccia grafica del programma Nserver

dere nel file sorgente il file header Nclient.h e nella fase di compilazione il file oggetto Nclient.o. Una volta creato il file oggetto è indispensabile, se vogliamo lanciare l'ese-

guibile che l'applicazione Nserver sia attiva , il motivo di ciò è illustrato nella figura 6.2.

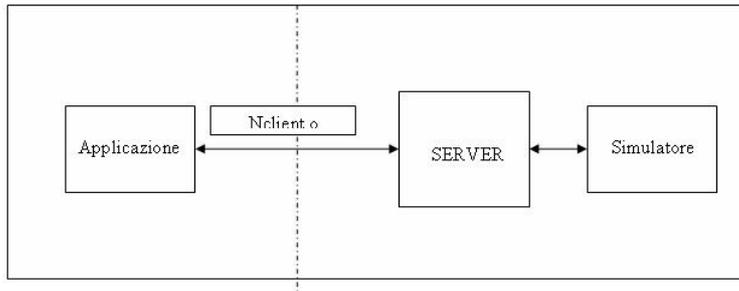


Figura 6.2: Programmazione in modalità client

Appendice B

Generazione del rumore Gaussiano

Sia $f_0[m]$, $m = 0, 1, 2, \dots, N - 1$ il vettore delle misure senza rumore; il vettore delle misure reali $f[m]$ differisce da $f_0[m]$ per quantità che variano in modo casuale, quindi il vettore $f[m]$ risulta:

$$f[m] = f_0[m] + w[m]$$

dove $w[m]$ è la realizzazione di una variabile aleatoria con le seguenti proprietà:

- per m diverso da m' , $w[m]$ e $w[m']$ sono realizzazioni di variabili aleatorie indipendenti;
- $w[m]$ è la realizzazione di una variabile aleatoria Gaussiana (o normale) con media 0 e varianza σ^2 ; quindi la sua densità di probabilità è data da:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (6.0.1)$$

dove $x = w[m]$.

Di seguito è mostrata la funzione `gaussdev()` che implementa l'equazione (6.0.1). L'implementazione si basa sul generare una coppia di numeri, ognuno con distribuzione uniforme in $[0,1]$.

Essendo il rumore gaussiano additivo basta aggiungere ad ogni valore del segnale il risultato della chiamata alla funzione `gaussdev` opportunamente scalato. Infatti

```
float gausdev ()
{
  float v1,v2,r2,fac;
  do {
    v1=2.0*(float)random()/RAND_MAX-1.0;
    v2=2.0*(float)random()/RAND_MAX-1.0;
    r2=v1*v1+v2*v2;
  } while (r2>=1.0 || r2 == 0.0);
  fac=sqrt(-2.0*log(r2)/r2);

  return v2*fac;
}
```

Figura 6.3: Funzione per la generazione di numeri con distribuzione di probabilità gaussiana

possiamo cambiare la deviazione standard del rumore moltiplicando il risultato della funzione per la deviazione standard voluta.

Bibliografia

- [1] J. Borenstein, H. R. Everett, and L. Feng, “*Where am I? Sensors and Methods for Mobile Robot Positioning*” Edited and compiled by J. Borenstein, April 1996
- [2] Greg Welch - Gary Bishop, “*An Introduction to the Kalman Filter*”, 2004
- [3] Agostino Martinelli, “*Lectures on Mobile Robot Localization*”, 2004
- [4] Raj Madhavan - Kingsley Fregene - Lynne E. Parker, “*Distributed Heterogeneous Outdoor Multi-robot Localization*”, 2002
- [5] Robert Grover Brown - Patrick Y.C. Hwang, “*Introduction to random signals and applied Kalman filtering*”, 1997
- [6] Hans-Joachim von der Hardt, Didier Wolf, René Husson, “*The Dead Reckoning Localization System of the Wheeled Mobile Robot ROMANE*”, Proceedings of the 1996 IEEEISICEIRSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems
- [7] Rudy Negenborn, “*Robot Localization and Kalman Filters On finding your position in a noisy world*”, A thesis submitted to the Institute of Information and Computing Sciences in partial fulfillment of the requirements for the degree of Master of Science, specialized in Intelligent Systems, September 1, 2003
- [8] Emanuele Crisostomi, “*Filtraggio e stima dello stato nei sistemi dinamici non lineari*”, Laurea Specialistica in Ingegneria dell’Automazione, Università di Pisa,

Sessione di Laurea del 04/03/2005

- [9] Fabrizio Romanelli - Luciano Spinello, "*Guida al NOMAD XR4000*",
- [10] Nomadic Technologies, inc, "*Language Reference Manual*", 1997