

UNIVERSITÀ
DEGLI STUDI
DI PADOVA



DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Corso di laurea in ingegneria dell' informazione

Identificazione di modello di un veicolo

Relatore: Prof. Mattia Zorzi

Laureando: Giacomo Greco

Anno accademico 2021-2022

Data di laurea 23/09/2022

Indice

1)Introduzione	3
2)Sistema di coordinate del veicolo	3
2.1)Roll angle	4
2.2)Pitch angle	4
2.3)Yaw angle	5
3)Modello del veicolo	6
4)Calcolo dello yaw rate	7
5)Studio del modello	8
5.1) Individuazione dell' altro angolo	9
6)Creazione del dataset	10
6.1)Simulazioni per testare la validità del modello	10
6.2) Aggiunta del rumore	14
6.3) Fase di detrend	14
7)Approccio black box	16
7.1)Richiamo sulle trasformate zeta	17
7.2) Trasformata z di una sequenza ritardata	17
8)Identificazione del modello di veicolo	17
8.1)Trovare la matrici Θ a massima verosomiglianza e le matrici A e B	21
9)Scelta di n_a e n_b tramite minizzazione dell'errore	23
10) Conclusioni	26

1 Introduzione

Nella tesi è studiato la dinamica di un veicolo che sta attraversando una curva. Il modello descrive le equazioni di moto del veicolo, ovvero le velocità longitudinali e laterali della macchina. La conoscenza accurata del modello del veicolo è la parte determinante per permettere un controllo più preciso del veicolo stesso.

Attraverso lo studio dell'equazioni che descrivono la struttura del modello l'obiettivo è identificare un modello dinamico a tempo discreto della struttura $x(k+1) = Ax(k) + Bu(k)$

Il modello studiato è una chiara semplificazione della realtà, sia perché viene individuato un sistema lineare mentre il modello in sé non lo è, sia perché sono state omessi dei parametri per il calcolo delle traiettorie che descrivono il comportamento di un veicolo in curva.

Inoltre non vi è la componente di attrito che influenzerebbe la dinamica del veicolo. L'identificazione è stata attuata attraverso metodi studiati nei corsi del terzo anno, quali l'uso dello stimatore a massima verosimiglianza che equivale alla minimizzazione dell'errore di predizione.

2 Sistema di coordinate del veicolo

I sistemi di controllo che aiutano il conducente a evitare incidenti, o limitare i danni in caso di incidente, sono diventati onnipresenti nel passeggero e nelle macchine moderne. Ad esempio, le auto recenti in genere hanno un sistema di frenata antibloccaggio (ABS) che impedisce le ruote di bloccarsi durante una brusca frenata e possono disporre anche di un sistema di controllo elettronico (ESC), che stabilizza il movimento laterale del veicolo per prevenire slittamenti. Questi sistemi si basano su informazioni dello stato del veicolo e dintorni che vengono ottenute tramite sensori. Per una tipica autovettura con sistema ESC, misure necessarie includono l'angolo del volante, le velocità angolari delle ruote, accelerazione laterale e la velocità di rotazione attorno all'asse verticale del corpo fisso, noto come velocità di imbardata (yaw rate).

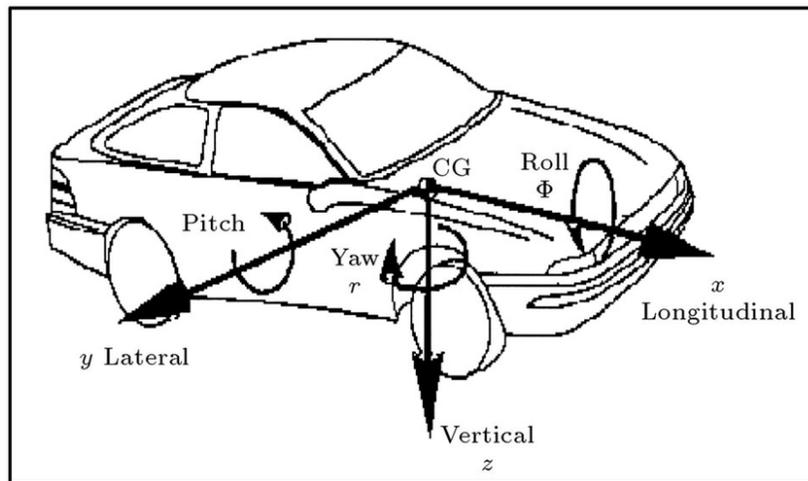


Figure 1: Degrees of freedom of the vehicle.

La figura 1 mostra i gradi di libertà del veicolo nel sistema di coordinate *Center of Gravity*.

Il sistema di coordinate *Center of gravity*, in cui l'origine è situata nel centro di gravità del veicolo, è il più importante.

$x_{CoG}, y_{CoG}, z_{CoG}$	Assi del sistema di coordinate Center of gravity
ψ	Yaw angle (rotazione attorno all'asse z)
χ	pitch angle (rotazione attorno all'asse y)
ϕ	roll angle (rotazione attorno all'asse x)

Table 1: Riassunto delle variabili del sistema di coordinate.

2.1 Roll angle

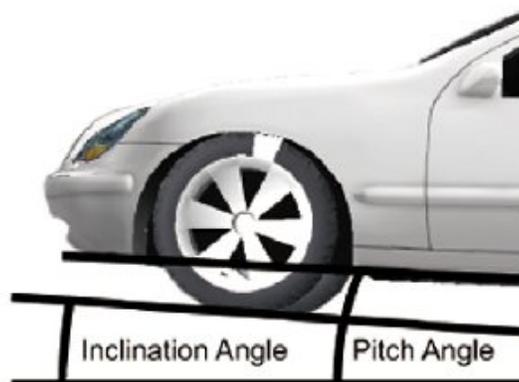
Roll è il movimento laterale di un'auto attorno all'asse longitudinale. Definisce quanto bene l'auto distribuisce il suo peso in curva. Il rollio avviene in risposta alle forze centrifughe. In una brusca manovra di svolta, esiste un rollio, ma se è eccessivo fa oscillare l'auto.



Figure 2: Vehicle rolling in a curve.

2.2 Pitch angle

È il movimento anteriore e posteriore di un'auto attorno all'asse laterale e trasversale che passa per il sistema del baricentro. Avviene in risposta alle forze di accelerazione e decelerazione. Quando un'auto sta decelerando (in frenata), il peso dell'auto viene trasferito sulle ruote anteriori, che tendono a inclinare in avanti la carrozzeria dell'auto, in caso di accelerazione l'auto si inclina verso l'indietro, trasferendo i pesi sulle ruote posteriori.

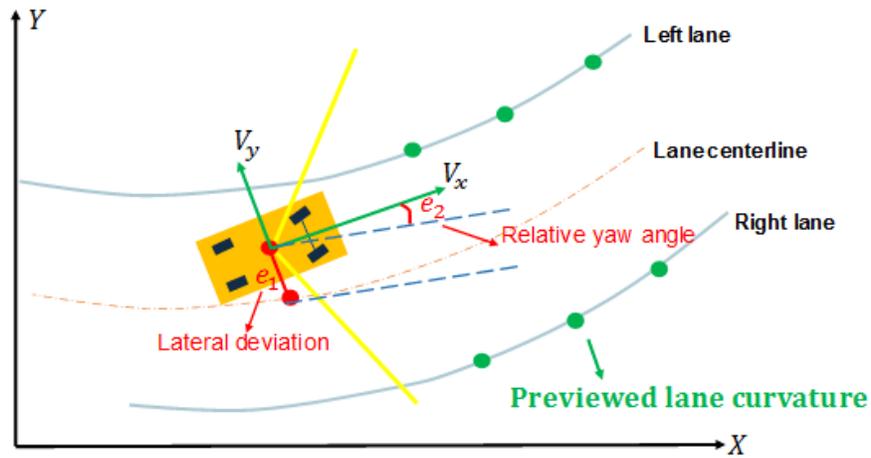


2.3 Yaw angle

L'imbardata è la rotazione dell'auto attorno all'asse verticale, che dipende dalle curve e talvolta dal vento laterale.

Ogni volta che c'è una manovra di svolta, le ruote dell'auto ruotano a velocità diverse.

Quando l'auto gira a sinistra, le ruote destre girano più velocemente delle ruote a sinistra. Questo perché le ruote di destra coprono la maggiore distanza rispetto alle ruote di sinistra che a sua volta creano una forza sbilanciante che influenza la svolta del veicolo.



Nel nostro modello semplificato, useremo lo yaw rate.

3 Modello del veicolo

Per un'auto che guida su una superficie orizzontale, la velocità longitudinale e la velocità laterale al CG (Center of Gravity) sono governate dalle equazioni di moto:

$$\begin{aligned}\dot{V}_x &= a_x + \dot{\psi} V_y \\ \dot{V}_y &= a_y - \dot{\psi} V_x .\end{aligned}$$

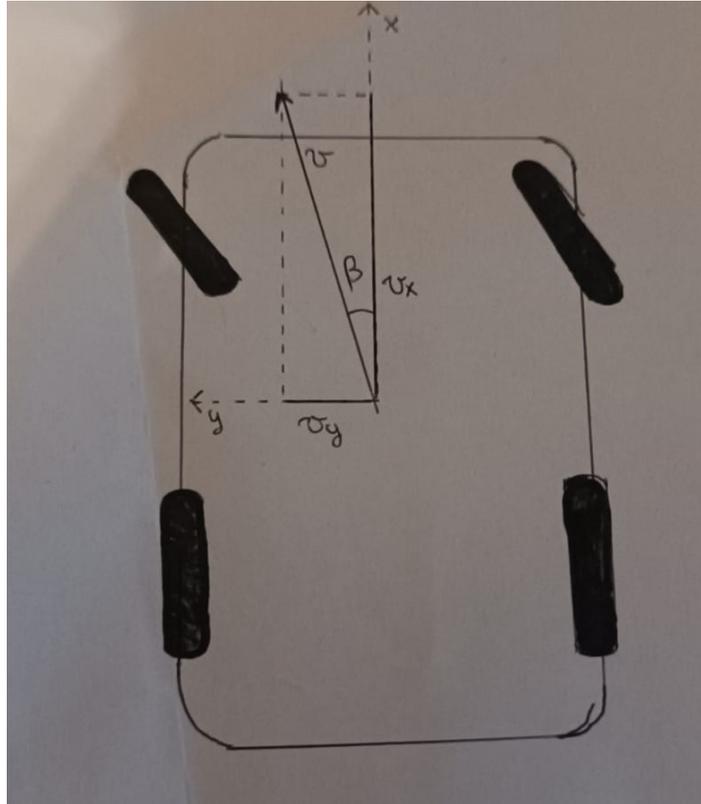


Figure 3: Illustrazione della velocità del veicolo, della velocità di imbardata, dello slittamento laterale del veicolo, angolo di slittamento del pneumatico e forze di attrito del pneumatico stradale. Il veicolo il vettore di velocità v al baricentro viene scomposto nel velocità longitudinale v_x , che è positiva nella direzione in avanti, e la velocità laterale v_y , che è positiva verso sinistra. L'angolo β tra l'asse x longitudinale e la velocità il vettore v è l'angolo di sbandamento laterale del veicolo, definito positivo in senso antiorario.

V_x e V_y sono le velocità longitudinali e laterali, a_x e a_y sono le accelerazioni longitudinali e laterali, e $\dot{\psi}$ è il tasso di imbardata. Ciascuno delle due equazioni include un termine di accelerazione e un termine che coinvolge la velocità di imbardata.

Le accelerazioni sono direttamente correlati alle forze che agiscono sul veicolo, imputabili principalmente all'attrito strada-pneumatico. I termini che coinvolgono la velocità di imbardata appaiono perché il sistema di coordinate in cui si calcola le velocità e le accelerazioni è solidale con l'auto, che ruota rispetto ad un sistema di coordinate inerziale.

4 Calcolo dello yaw rate

Le misure dello yaw rate accurate sono essenziali per il controllo dei sistemi della dinamica di un veicolo. solitamente lo yaw rate è misurato con un giroscopio a sensore.

Uno dei principali svantaggi dei giroscopi disponibili è il loro offset drift causato dalle brusche variazioni di temperatura. Quindi, per migliorare l' accuratezza della stima dello yaw rate, si combinano misure da sensori differenti.

Nelle curve, le ruote interne ed esterne del veicolo corrono con velocità diverse. Le ruote esterne percorrono una distanza maggiore rispetto alle ruote interne. La differenza di velocità può essere utilizzata per calcolare l'imbardata sull'asse posteriore e anteriore:

$$\dot{\psi}_f = \frac{(w_{fr}-w_{fl})r_{stat}}{b_f \cos\delta}$$

$$\dot{\psi}_r = \frac{(w_{fr}-w_{fl})r_{stat}}{b_r}$$

dove

w_{fr} è la velocità angolare della ruota anteriore destra;

w_{fl} è la velocità angolare della ruota anteriore sinistra;

w_{rr} è la velocità angolare della ruota posteriore destra;

w_{rl} è la velocità angolare della ruota posteriore sinistra;

r_{stat} è il raggio statico della ruota;

b_f è la larghezza della parte anteriore della macchina;

b_r è la larghezza della parte posteriore della macchina.

Il terzo fattore dello yaw rate $\dot{\psi}_r$ viene ricavato mediante il giroscopio a sensore.

Combinando i 3 fattori trovati , mediante una logica fuzzy , e con coefficienti di peso $h1$, $h2$ e $h3$ si ottiene il valore dello yaw rate. Se un segnale non è affatto affidabile viene pesato 0

5 Studio del modello

Nel nostro modello, consideriamo un'auto che sta girando a sinistra aventi le ruote frontali inclinate di un certo angolo α con l'asse x . L'angolo si forma tra la direzione della ruota e l'asse longitudinale.

Siccome il veicolo, sta girando a sinistra l'angolo α_{left} della ruota sinistra sarà un po' superiore rispetto all'angolo α_{right} della ruota destra.

Questa differenza dipende sostanzialmente dalle dimensioni della macchina, e dalla distanza dal punto di curvatura, come vedremo in seguito.

Le ruote dietro sono ferme e dritte, per semplicità.

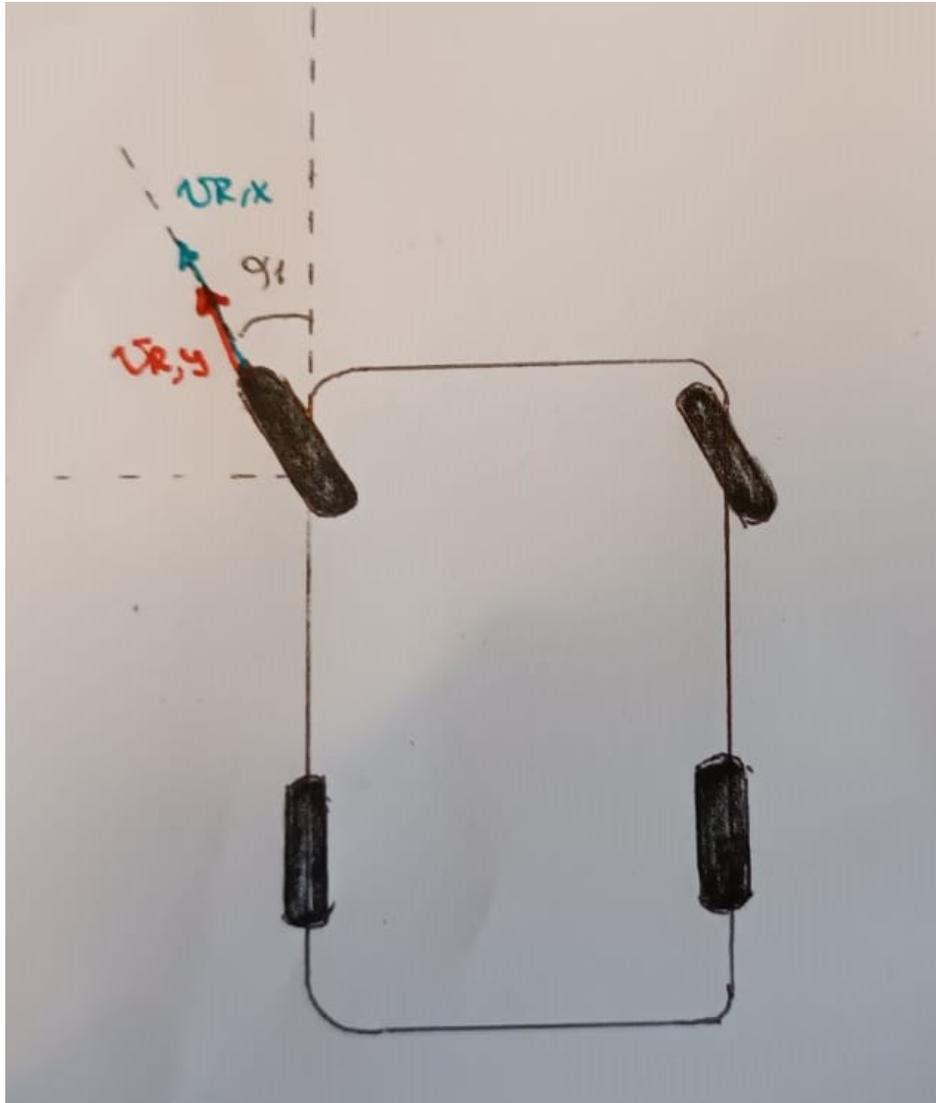


Figure 4: Veicolo in curva con le ruote anteriori inclinate.

Si hanno i valori delle velocità iniziali V_{x0} e V_{y0} oppure la velocità iniziale della macchina V_0 e si calcola $V_{x0} = V_0 \cos(\alpha)$ e $V_{y0} = V_0 \sin(\alpha)$.

Per calcolare lo yaw rate, sono necessarie le velocità angolari delle ruote destra e sinistra.

La velocità della ruota sinistra ad esempio, che si vede nel disegno, è la somma tra il vettore blu $V_{rl,x}$ e il vettore rosso $V_{rl,y}$.

Utilizzando le regole goniometriche si ha che

$$V_{rl,x} = V_x \cos(\alpha_{left}) ,$$

$$V_{rl,y} = V_y \sin(\alpha_{left}) ,$$

$$V_{rl} = V_{rl,x} + V_{rl,y} .$$

Ripetendo l' analogo per la ruota destra, utilizzando l' angolo della ruota destra al posto dell' angolo della ruota sinistra, si trova la velocità della ruota destra.

Usando le relazioni $w_{fr} = V_{rr}/r_{stat}$ e $w_{fl} = V_{rl}/r_{stat}$ si trovano le velocità angolari delle ruote anteriori.

Si noti che lo yaw rate è calcolato utilizzando la differenza di angoli tra la ruota sinistra e destra.

Siccome nel nostro modello, le ruote posteriori sono considerate diritte non c'è differenza di angolo tra le ruote destra e sinistra e quindi non teniamo in considerazione lo yaw rate della parte posteriore.

Lo yaw rate quindi si trova mediante la relazione

$$\psi = \frac{w_{fr} - w_{fl} r_{stat}}{b_f \cos \delta}$$

dove si è impostato l' angolo del volante $\delta = 0$.

5.1 Individuazione dell' altro angolo

Conoscendo l' angolo α_{left} che nel nostro modello è fissato $\alpha_{right} = 30^\circ$, si può trovare il valore di α_{left} , importante per trovare ψ , utilizzando la distanza tra le ruote anteriori e posteriori, come si vede in figura.

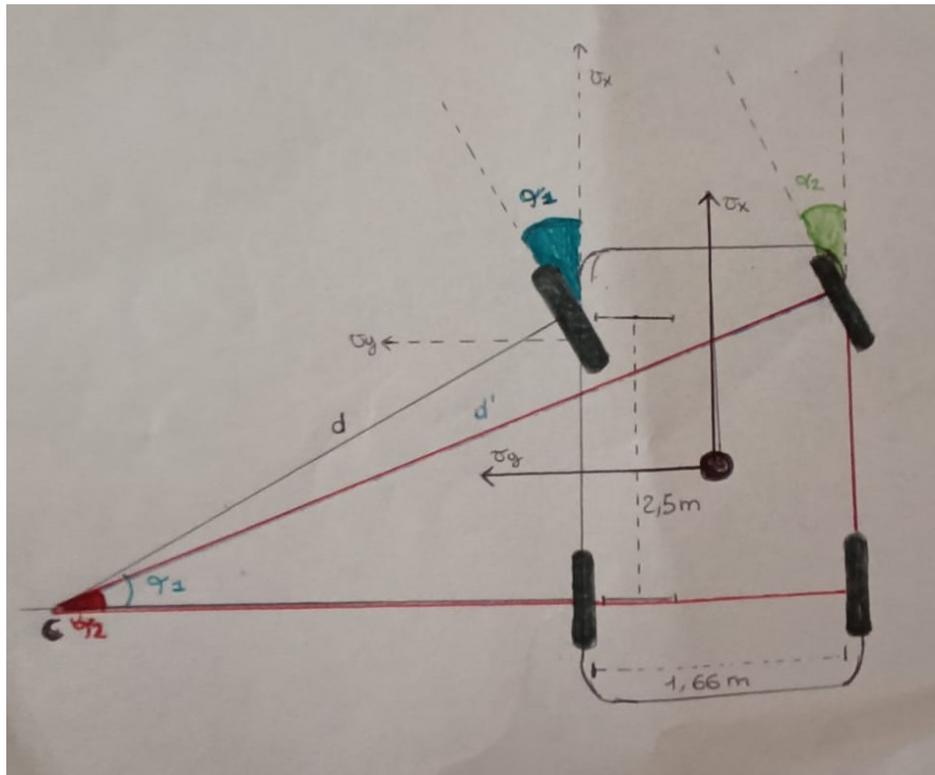


Figure 5: L'angolo della ruota viene individuato tramite i triangoli che forma con il punto di curvatura

Come si vede dalla figura, l' angolo α_{left} corrisponde all' angolo più grande del triangolo mentre l' angolo α_{right} corrisponde all' angolo più piccolo.

Sapendo la distanza tra le due ruote, che nel nostro caso è $t = 2.5m$ possiamo ricavare la distanza tra la ruota più vicina alla curva e il punto di curvatura mediante la relazione $t = d\sin\alpha_{left}$ quindi

$$d = \frac{t}{\sin\alpha_{left}} = \frac{2.5}{\sin 30} = 5 \text{ m}$$

Ora , per trovare la lunghezza che va dal punto C alla curva posteriore, e che chiameremo l , applichiamo la relazione $l = d\cos\alpha_{left} = 5\cos 30 = 4.33$ m e poi sommiamo questa lunghezza a b_r , larghezza della parte posteriore per ottenere la base del triangolo rosso ($l' = l + b_r = 4.33 + 1.66 = 6$ m)

A questo punto abbiamo le relazioni $t' = t = d\sin\alpha_{right}$ e $l' = d\cos\alpha_{right}$ possiamo ricavare che $\tan\alpha_{right} = \frac{t}{l'}$ quindi $\alpha_{right} = \arctan\frac{2.5}{6} = 22.62^\circ = 0.4$ rad

6 Creazione del dataset usando ode45

Per poter identificare il modello è necessario trovare i valori delle uscite in corrispondenza ai valori degli ingressi trovando quindi un dataset, ovvero una collezione di dati.

Si ricordi che le equazioni del modello sono

$$\dot{\mathbf{V}}_x = \mathbf{a}_x + \psi \mathbf{V}_y$$

$$\dot{\mathbf{V}}_y = \mathbf{a}_y - \psi \mathbf{V}_x$$

Si ricordi che si sta studiando il comportamento di un veicolo in curva. $a_x(t)$ e $a_y(t)$ sono le accelerazioni e sono gli ingressi del sistema, infatti si possono immaginare come la forza che l' uomo fornisce dall' esterno per premere il pulsante dell' acceleratore. $v_x(t)$ e $v_y(t)$ sono invece i valori delle uscite, che si trovano risolvendo l'equazione differenziale associata. Sapendo che

$$\psi = \frac{(w_{fr} - w_{fl})r_{stat}}{b_f \cos(\delta)}$$

si possono effettuare varie simulazioni su Matlab per testare inanzitutto **la validità del modello**.

Nelle simulazioni, come detto, l'angolo della ruota sinistra, visto precedentemente, è fissato a 30° e quindi l'angolo destro, ricavato nel paragrafo precedente, è 22.6° . Attraverso la funzione **ode45** di Matlab si può risolvere l'equazione differenziale associata e mostrare graficamente l'andamento in funzione del tempo dei segnali $v_x(t)$ e $v_y(t)$. Per fare ciò sono necessari, oltre alla struttura del modello, anche le condizioni iniziali e l'intervallo di tempo in cui vengono effettuate le misurazioni. Il tempo di simulazione utilizzato è $[0, 20]$ secondi, e il periodo di campionamento è $T = 100$ millisecondi, quindi c'è un campione ogni 100 millisecondi e così il numero di campioni è $N = 20/0.1 + 1 = 201$. Si aggiunge 1 perchè al tempo $t = 0$ vi è il primo campione (in Matlab gli array iniziamo con l'indice 1).

6.1 Simulazioni per identificare e testare il modello

La velocità iniziale V_0 , in direzione della ruota sinistra è di 50 km/h . Quindi si possono trovare le condizioni iniziali come $V_{x0} = V_0 \cos(30^\circ) = 43 \text{ km/h}$ e $V_{y0} = V_0 \sin(30^\circ) = 25 \text{ km/h}$

Per identificare e testare il modello, si utilizzano degli ingressi realistici. Per esempio, il fatto che l' uomo schiacci il pedale dell' acceleratore può essere approssimato a una funzione gradino in cui il valore dell' accelerazione passa da 0 alla costante nell'istante in cui si schiaccia il pedale.

Come primo esempio, quindi, si definisce $a_x(t) = 5[1(t - 2)]$, ovvero un gradino che passa da 0 a 5 km/h al tempo $t = 2$ secondi. $a_y(t)$ per il momento è impostata a 0.

Le velocità invece sono

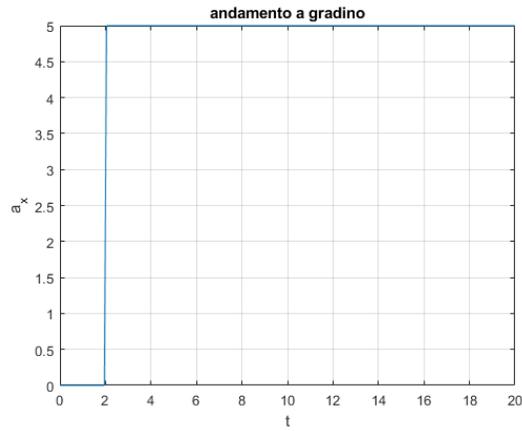


Figure 6: $a_x(t) = 5[1(t - 2)]$.

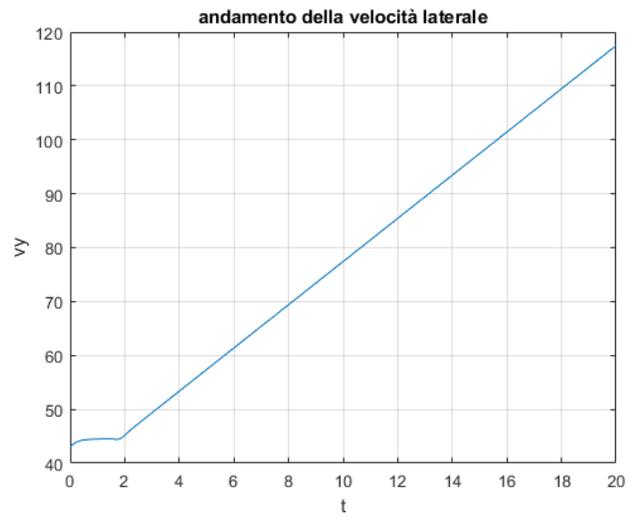
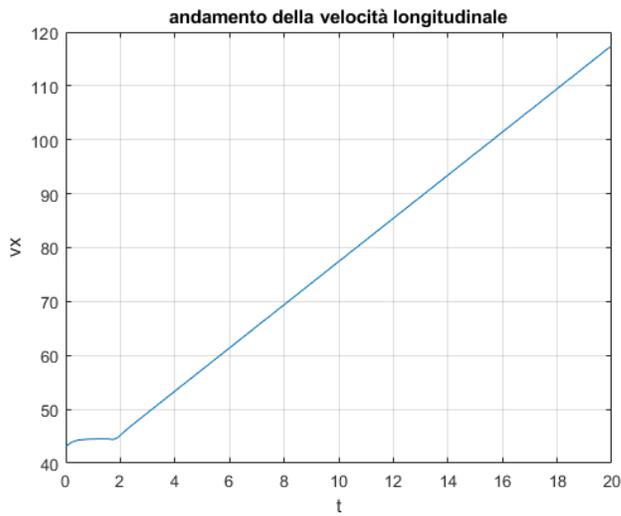


Figure 7: uscite in risposta all' ingresso a gradino.

Come si vede, fino a due secondi le velocità laterali e longitudinale sono circa costanti, poi vi è un'accelerazione costante che fornisce alle velocità un'andamento lineare. Si noti che le velocità raggiungono in poco tempo alti valori perchè in questo modello non vi è la componente di attrito che dissiperebbe energia, quindi le velocità tendono ad esplodere con un andamento lineare.

Se invece, si ipotizza più realisticamente, che il passaggio dell' accelerazione da 0 a una costante, non avvenga istantaneamente, ma avviene gradualmente si può ipotizzare $a_x(t) = 1 - \exp(-t/3)$

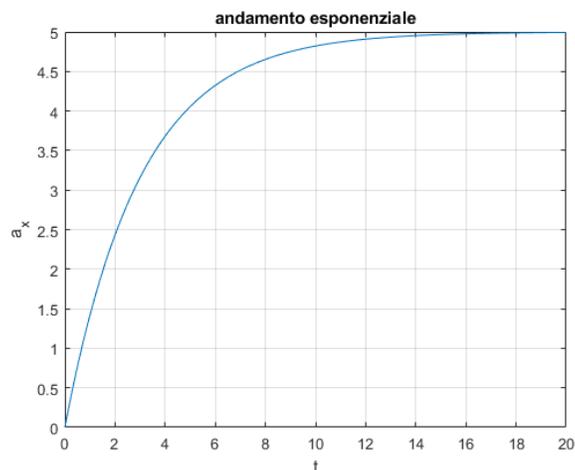


Figure 8: $a_x(t) = 1 - \exp(-t/3)$.

che ricorda il processo di carica di un condensatore. La costante di tempo è $T = 3$ secondi ricavabile dalla forma $e^{-\frac{t}{T}}$ quindi la funzione arriva al valore massimale circa in $5T = 15$ secondi, da lì in poi l' accelerazione diventa una costante, esattamente come nel caso prima.

Le velocità in questo caso sono

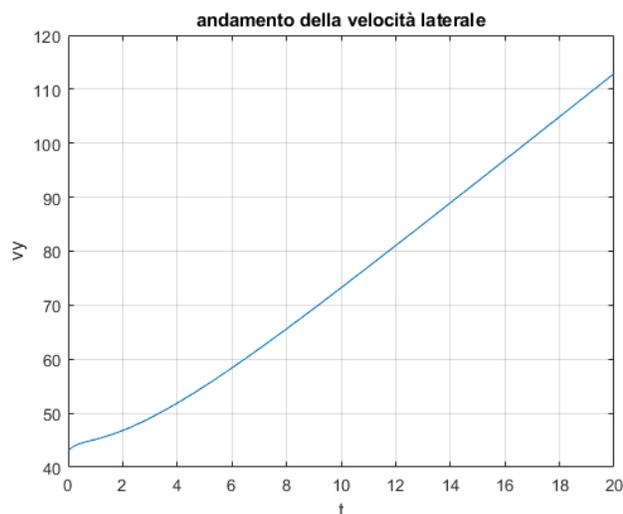
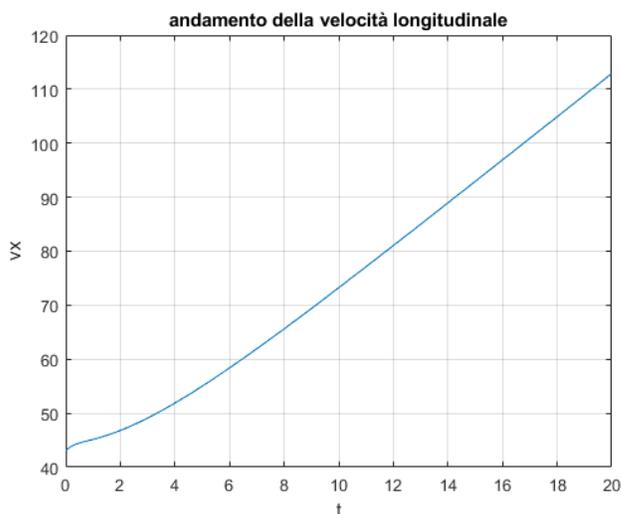


Figure 9: Uscite in risposta all'esponenziale negativo.

Qui, le velocità hanno un' andamento un pò meno rapido del caso precedente, perchè la funzione a_x , che determina la pendenza del grafico delle velocità è meno crescente del caso precedente.

Per poter identificare e testare il modello in maniera più consistente è necessario **eccitare il modello**. L' eccitazione consiste nel far "lavorare" il modello a **frequenze diverse**.

Più riesco a eccitare il modello, migliore sarà l' identificazione attraverso il dataset.

Quindi il segnale gradino non è sufficiente, perchè se è vero che durante il passaggio da 0 alla costante il modello lavora a moltissime frequenze, è anche vero che dopo la salita il modello viene eccitato solo da un segnale costante. Quindi l' informazione viene concentrata solo in un piccolissimo istante di tempo e questo non va bene. Per rendere il dataset più realistico e permettere al modello di lavorare a più frequenze possibili, si possono usare segnali periodici come onde quadre, onde trapezoidali oppure onde triangolari. Usiamo, quindi, come ingressi $a_x(t)$ un' onda quadra che inizia a 5 e ha semiperiodo $T/2 = 2$ secondi e come $a_y(t)$ un' onda quadra che passa dal valore $\frac{-1}{8}a_x$ a $\frac{-1}{10}a_x$ sempre in un semi-periodo $T/2 = 2$ secondi.

Le onde sono come in figura 10:

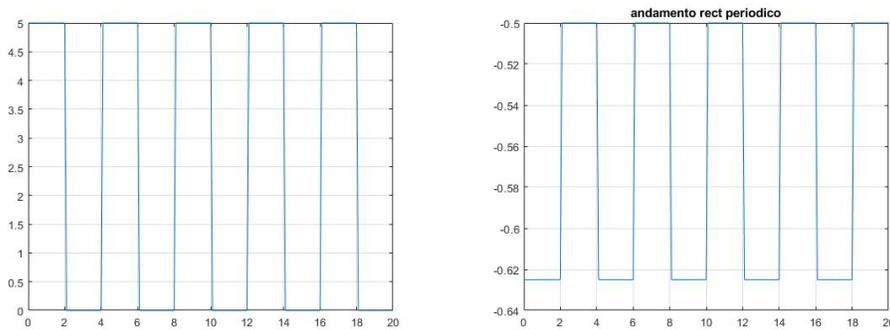


Figure 10: Onde quadre.

$a_y(t)$ è stata definita negativa per simulare la forza centrifuga che porta il veicolo verso l' esterno in curva. Ovviamente questa accelerazione non deve essere tanto elevata perchè altrimenti il veicolo sbanderebbe davvero.

Le velocità in risposta alle accelerazioni sono:

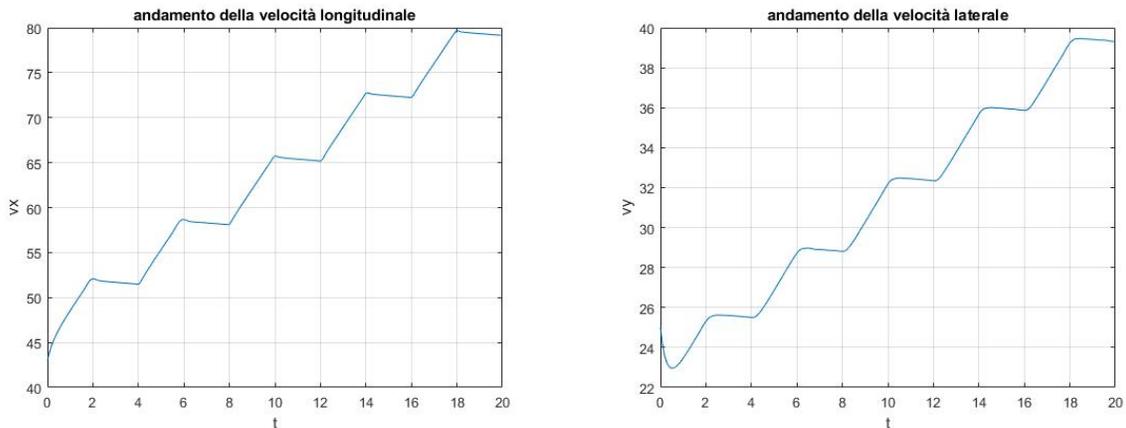


Figure 11: Uscite in risposta alle onde quadre.

6.2 Aggiunta del rumore

Introduciamo del rumore al segnale . Il rumore è l'insieme di segnali imprevedibili e indesiderati che si sovrappongono al segnale utile, solitamente si intendono segnali di origine aleatoria provenienti dall'interno e vengono descritti in termini probabilistici.

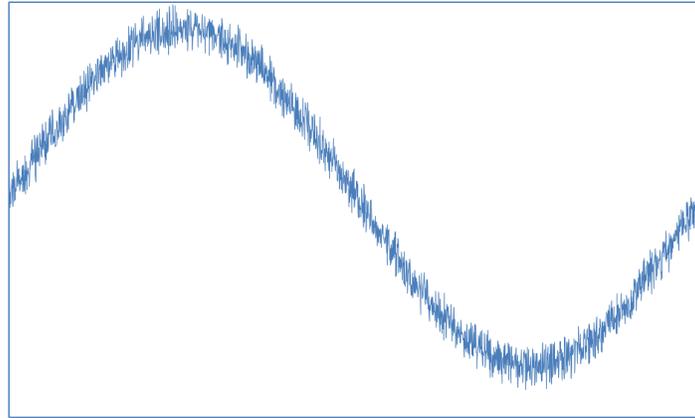


Figure 12: Esempio di funzione sinusoidale affetta da rumore.

Nel caso studiato, il rumore è stato aggiunto ai segnali $a_x(t)$, $a_y(t)$, $v_x(t)$, $v_y(t)$ tramite una variabile aleatoria gaussiana di valori casuali di media 0 e varianza 0.1^2 .

Il rumore è stato introdotto per simulare gli errori di misura, dovuti al fatto che gli strumenti non riescono a misurare con estrema precisione i dati.

6.3 Fase di detrend

Per fare l'identificazione, i segnali di ingresso e di uscita devono avere media nulla. Dalla teoria dei controlli, infatti, un sistema **non lineare** viene linearizzato attorno al suo punto di equilibrio.

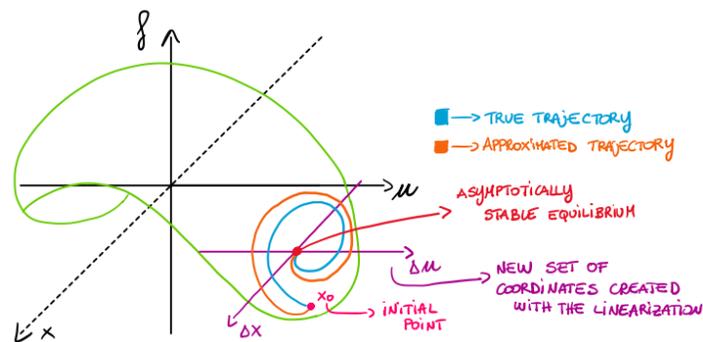


Figure 13: L'origine del sistema linearizzato è il punto di equilibrio su cui si effettua la linearizzazione.

Siccome si ha un modello non lineare e si vuole realizzare un'identificazione dinamica del modello tramite una funzione di trasferimento, si sta realizzando una linearizzazione.

Se si vuole che il valore di y e di u coincidano con i valori del sistema linearizzato è necessario attuare il detrend ai segnali, ovvero renderli a media nulla.

In questo modo l'origine del sistema di equilibrio diventa $(0,0)$ (si veda Figura 13) e i valori del sistema non linearizzato coincidono con i valori del sistema linearizzato a meno di approssimazioni dovute alla linearizzazione.

Per i segnali d'ingresso presenti nella figura 10 trovare la media è facile. Ad esempio, si nota subito che la media di $a_x(t)$ è 2.5. Eseguendo in Matlab le operazioni $a_{xm}(t) = a_x(t) - \text{mean}(a_x(t))$ e $a_{ym}(t) = a_y(t) - \text{mean}(a_y(t))$ si ottengono i segnali di ingresso a media nulla desiderati.

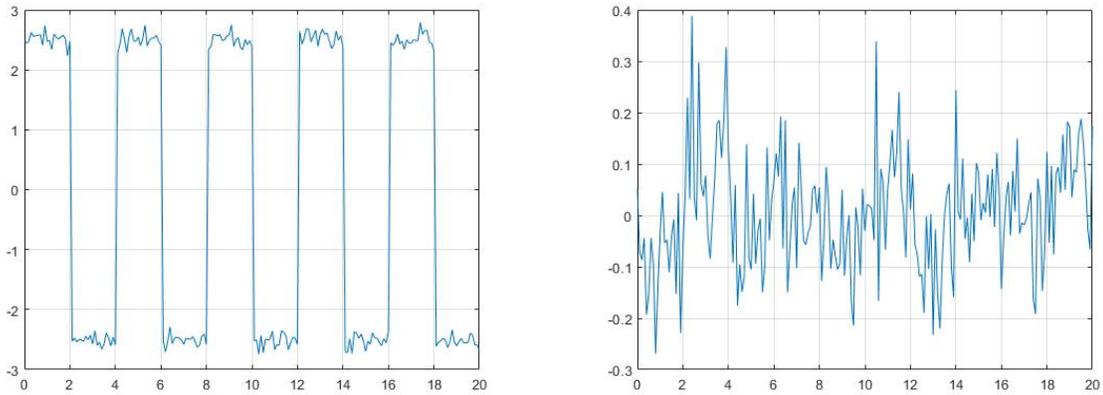


Figure 14: Segnali di ingresso a media nulla e con rumore.

Per i segnali di uscita $v_x(t)$ e $v_y(t)$ il processo di detrending è un pò più complicato. Infatti i segnali presenti nella Figura 13 hanno un'andamento simile a una funzione lineare. Quindi per renderli a media nulla, è necessario toglier loro l'andamento lineare.

In Matlab, questa operazione viene fatta dal comando `detrend(data,1)` dove quell'1 indica appunto un polinomio del primo grado.

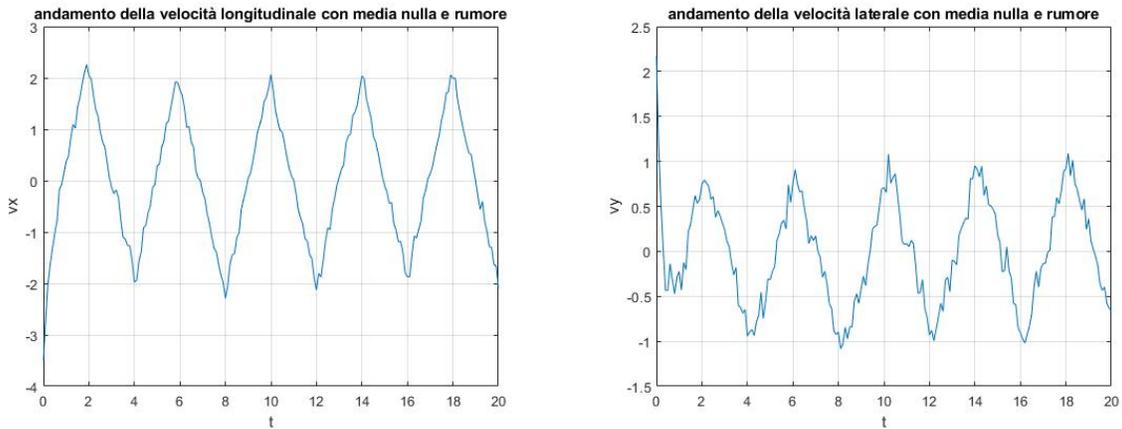


Figure 15: Segnali di uscita a media nulla e con rumore.

Ora abbiamo il dataset per l'identificazione.

7 Approccio black-box

Avendo trovato il dataset , tramite la funzione dell'ode45, possiamo definire

$$u^N = [u(1), u(2), u(3), \dots, u(N)] \text{ e}$$

$$y^N = [y(1), y(2), y(3), \dots, y(N)]$$

dove $1,2,3,\dots,N$ sono i campioni del vettore t e u^N e y^N sono rispettivamente i valori degli ingressi e delle uscite negli istanti di campionamento.

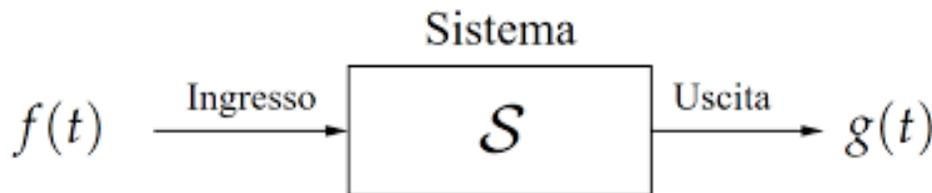
Quindi, nel caso studiato

$$u(t) = \begin{bmatrix} a_x(t) \\ a_y(t) \end{bmatrix} \text{ e } y(t) = \begin{bmatrix} v_x(t) \\ v_y(t) \end{bmatrix}$$

Si noti, quindi che ogni singola componente $u(1), u(2), u(3), \dots, u(N)$ oppure $y(1), y(2), y(3), \dots, y(N)$ è composta da due elementi.

Si ricordi che le simulazioni avvengono nell' intervallo di tempo $[0, 20]$ secondi e che il periodo di campionamento è $T = 0.1$ secondi.

Tramite questi vettori, l' obiettivo è **trovare un modello dinamico a tempo discreto** che spieghi quello che ha misurato. Usiamo un ' approccio black box e ci concentriamo sui modelli LTI.



In questo caso $y(t) = G(z)u(t) + e(t)$ dove

$$y(t) = \begin{bmatrix} v_x(t) \\ v_y(t) \end{bmatrix} \text{ è l' uscita, quindi è una matrice } 2 \times 1;$$

$G(z)$ è la matrice di trasferimento a tempo discreto, è una matrice 2×2 ;

$$u(t) = \begin{bmatrix} a_x(t) \\ a_y(t) \end{bmatrix} \text{ è l'ingresso, è una matrice } 2 \times 1;$$

$e(t)$ è l' errore di misura o rumore;

E' importante osservare che la formula sopra è una notazione simbolica, in realtà sarebbe il prodotto tra le trasformate oppure la convoluzione nel dominio del tempo

7.1 Richiamo sulla trasformata zeta

La trasformata Z è un operatore che associa ad una funzione $f(k)$ di variabile discreta (o, equivalentemente, ad una sequenza numerica) una funzione $F(z)$ di variabile complessa definita come segue

$$F(z) = Z(f(k)) = f(0) + f(1)z^{-1} + f(2)z^{-2} + \dots = \sum_{k=0}^{\infty} f(k)z^{-k}$$

La trasformata Z è una serie di potenze nella variabile (complessa) z^{-1}

La trasformata Z si declina anche in una forma differente (bilatera) in cui l'indice inferiore della sommatoria vale $-\infty$

$$\sum_{k=-\infty}^{\infty} f(k)z^{-k}$$

La trasformata Z unilatera ha però proprietà che la rendono maggiormente conveniente per lo studio dei sistemi dinamici a tempo discreto. Si noti che le espressioni della Z trasformata unilatera e bilatera coincidono per segnali causali, cioè segnali $f(k)$ identicamente nulli per $k < 0$.

7.2 Trasformata z di una sequenza ritardata

Data una sequenza $x(k)$, identicamente nulla per $k < 0$, si consideri un intero $m > 0$ e la sequenza definita come $y(k) = x(k - m)$. Ciò corrisponde ad uno scorrimento verso destra di m istanti temporali del grafico.

Si dimostra che $Y(z) = Z(x(k - m)) = z^{-m}X(z)$ se la sequenza è identicamente nulla per $k < 0$.

8 Identificazione del modello di veicolo

Nel caso studiato $G(z) = \sum_{k=0}^{\infty} G(k)z^{-k}$ dove $G(k)$ è una matrice 2×2 e costituisce la risposta impulsiva del sistema. Infatti si ha che

$$y(t) = G(z)u(t) = \sum_{k=0}^{\infty} G(k)u(t - k) = G(k)_{k \geq 0} * u(t),$$

cioè l'uscita è data dalla convoluzione tra il segnale in ingresso e la risposta all'impulso del sistema, dove la trasformata Z di quest'ultima fornisce la funzione di trasferimento del sistema.

Il task di determinare la struttura di G_k e quindi di $G(z)$ viene chiamato **problema di identificazione del modello**. Per fare ciò, prima di tutto fissiamo la struttura di $G(z)$.

Si prende $G(z) = A(z)^{-1}B(z)z^{-1}$, dove con z^{-1} introduco un ritardo

Si definisce $A(z) = 1 + \sum_{k=1}^{n_a} A(k)z^{-k}$ e $B(z) = \sum_{k=0}^{n_b-1} B(k)z^{-k}$

$A(z), B(z), G(z)$ vengono dette matrici di polinomi perchè sono matrici 2×2 di polinomi di variabile z .

A questo punto, $y(t) = A(z)^{-1}B(z)z^{-1}u(t)$ e quindi $A(z)y(t) = B(z)u(t-1) + e(t)$ dove $e(t)$ è un rumore la cui presenza sta ad indicare che la relazione tra y e u non è esattamente descritta da $G(z)$.

$$\begin{aligned} \text{Quindi } y(t) + \sum_{k=1}^{n_a} A(k)y(t-k) &= \sum_{k=0}^{n_b-1} B(k)u(t-k-1) + e(t) \\ \text{ovvero } y(t) &= -\sum_{k=1}^{n_a} A(k)y(t-k) + \sum_{k=0}^{n_b-1} B(k)u(t-k-1) + e(t). \end{aligned}$$

Il modello è un caso particolare di un modello noto in letteratura con il nome di modello **ARX** (AutoRegressive model with eXternal input), la cui struttura generale è come vista sopra.

Supporremo sempre di avere $n_a \geq n_b$. Come già detto a suo tempo, n_a in queste condizioni misura la complessità del sistema. Diremo che n_a è l'ordine del sistema.

Si introduce il delay z^{-1} per avere l'uscita correlata al passato di y e solo del passato di u . Se non si introducesse il delay avrei l'uscita y correlata anche al presente di u e non solo al suo passato. Il presente di u si può ricavare mediante il suo passato ma è abbastanza complicato, e non varrebbe la formula della regressione lineare. In alternativa si potrebbe ridefinire la struttura del modello ARX in altro modo.

Adesso che abbiamo trovato la struttura del modello l'obbiettivo è quello di trovare $(A_k)_1^{n_a}$ e $(B_k)_0^{n_b-1}$ utilizzando il dataset u^N, y^N .

$$y(t) = -[A_1, A_2, \dots, A_{n_a}] \begin{bmatrix} y(t-1) \\ y(t-2) \\ \vdots \\ y(t-n_a) \end{bmatrix} + [B_0, B_1, \dots, B_{n_b-1}] \begin{bmatrix} u(t-1) \\ u(t-2) \\ \vdots \\ u(t-n_b) \end{bmatrix} + e(t)$$

$$y(t) = [-A_1, -A_2, \dots, -A_{n_a}, B_0, B_1, \dots, B_{n_b-1}] \begin{bmatrix} y(t-1) \\ y(t-2) \\ \vdots \\ y(t-n_a) \\ u(t-1) \\ u(t-2) \\ \vdots \\ u(t-n_b) \end{bmatrix} + e(t)$$

Usando la proprietà di trasposizione ed il prodotto tra due matrici si ottiene

$$(y(t))^T = [(-y(t-1))^T \dots (-y(t-n_a))^T (u(t-1))^T \dots (u(t-n_b))^T] \begin{bmatrix} (A_1)^T \\ \vdots \\ (A_{n_a})^T \\ (B_0)^T \\ \vdots \\ (B_{n_b-1})^T \end{bmatrix}.$$

Si definisce $\phi(t) = [(-y(t-1))^T \dots (-y(t-n_a))^T (u(t-1))^T \dots (u(t-n_b))^T]$. $\phi(t)$ ha dimensione $1 \times 2(n_a + n_b)$ in quanto ogni singola componente trasposte ha dimensione 1×2 e vi sono $n_a + n_b$ componenti che rendono $\phi(t)$ un vettore riga.

$$\text{Poi, si definisce } \Theta = \begin{bmatrix} (A_1)^T \\ \vdots \\ (A_{n_a})^T \\ (B_0)^T \\ \vdots \\ (B_{n_b-1})^T \end{bmatrix}.$$

In questo caso la matrice Θ ha dimensione $2(n_a + n_b) \times 2$ in quanto ogni singola matrice $(A_1)^T, \dots, (A_{n_a})^T, (B_0)^T, \dots, (B_{n_b-1})^T$ è una matrice 2×2 ed esse sono impilate una sotto l'altra per formare Θ . Si ottiene così $(y(t))^T = \phi(t)\Theta + (e(t))^T$. L'equazione fornisce il valore dell'uscita per ogni istante di tempo, o almeno quelli campionati.

$$\text{Impilando i vettori } \phi(t) \text{ per } t = 1, 2, \dots, N \text{ in modo da costruire la matrice } \Phi = \begin{bmatrix} \phi(1) \\ \phi(2) \\ \vdots \\ \phi(N) \end{bmatrix}$$

$$\text{e i vettori } y(t) \text{ per } t = 1, 2, \dots, N \text{ in modo da costruire la matrice } Y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix} \text{ si ottiene}$$

la formula di regressione lineare $Y = \Phi\Theta + E$, dove $E = \begin{bmatrix} e(1) \\ e(2) \\ \vdots \\ e(N) \end{bmatrix}$.

Dal corso di Sistemi e Modelli, sappiamo che lo stimatore a massima verosomiglianza è quello che minimizza la relazione

$$\Theta_{PEM}^{\hat{}} = argmin(\|Y - \Phi\Theta\|^2).$$

Il termine a destra dell'uguale è detto errore di predizione(prediction error).

Attraverso passaggi matematici si arriva alla relazione

$$\Theta_{PEM}^{\hat{}} = ((\Phi)^T\Phi)^{-1}(\Phi)^TY$$

dove pem sta per prediction error minimization.

8.1 Trovare la matrice Θ a massima verosomiglianza e le matrici A e B

Avendo trovato il modo per definire la struttura del modello il primo passo da fare è riuscire, attraverso il dataset, a trovare la forma del regressore, quindi trovare la matrice Φ , che si trova impilando le righe di ϕ , come visto precedentemente. Per ogni istante di tempo (negli istanti di campionamento) abbiamo

$\phi(t) = [(-y(t-1))^T \dots (-y(t-n_a))^T (u(t-1))^T \dots (u(t-n_b))^T]$. Per esempio se $n_a = 2$ e $n_b = 2$ allora

$\phi(4) = [(-y(3))^T, -y(2))^T, u(3))^T, u(2))^T]$. Come si può vedere la dimensione della matrice $\phi(4)$ è $1 \times 2(n_a + n_b)$ come si era detto precedentemente. Se però $n_a = 4$ e $n_b = 4$ e si vuole calcolare $\phi(1)$ ci saranno degli indici negativi. Si imposta il valore di $y(i)$ e $u(i)$ a 0 quando $i \leq 0$, perchè devo inizializzare le condizioni iniziali del sistema.

E' certo che si sta inserendo un errore perchè si inseriscono delle condizioni iniziali in generale errate, però essendo il modello BIBO-stabile, ci si "dimentica" delle condizioni iniziali quando il tempo tende a infinito. Questo giustifica il fatto di impostare $y(i)$ e $u(i)$ a 0 quando $i \leq 0$. Si noti quindi che per definizione $\phi(1) = \vec{0}$. Si costruisce la matrice Φ tramite l' algoritmo. Essa è una matrice triangolare superiore a blocchi. Infatti, se per esempio, $n_a = n_b = 2$

$$\Phi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_x(1) & v_y(1) & 0 & 0 & a_x(1) & a_y(1) & 0 & 0 \\ v_x(2) & v_y(2) & v_x(1) & v_y(1) & a_x(2) & a_y(2) & a_x(1) & a_y(1) \\ v_x(3) & v_y(3) & v_x(2) & v_y(2) & a_x(3) & a_y(3) & a_x(2) & a_y(2) \\ \vdots & \vdots \\ v_x(N-1) & v_y(N-1) & v_x(N-2) & v_y(N-2) & a_x(N-1) & a_y(N-1) & a_x(N-2) & a_y(N-2) \end{bmatrix}$$

Si trova poi

$$\Theta_{PEM} = (\Phi^T \Phi)^{-1} (\Phi^T Y)$$

Il prodotto $(\Phi^T \Phi)^{-1}$ forma una matrice quadrata di dimensioni $2(n_a + n_b)$. Il prodotto $(\Phi^T \Phi)^{-1} (\Phi)$ è quindi possibile perchè le colonne di $(\Phi^T \Phi)^{-1}$ coincidono con le righe di $(\Phi)^T$ e il loro prodotto forma una matrice rettangolare di dimensioni $2(n_a + n_b) \times N$.

Infine il prodotto tra $(\Phi^T \Phi)^{-1} (\Phi)$ e Y (che ha dimensioni $N \times 2$) forma una matrice rettangolare di dimensioni $2(n_a + n_b) \times 2$ come dalla teoria.

Attenzione: sia per costruire il regressore che per la matrice Y si sono usati i segnali a media nulla trovati dalla fase di detrend e con presenza di rumore.

La matrice Θ ha quindi le matrici $A_1^T, A_2^T, \dots, A_{n_a}^T, B_0^T, B_1^T, \dots, B_{n_b-1}^T$, ognuna avente dimensione 2×2 , impilate una sotto l' altra. Tornando all' esempio di prima con $n_a = n_b = 2$

$$\Theta = \begin{bmatrix} \Theta_{11} & \Theta_{12} \\ \Theta_{21} & \Theta_{22} \\ \Theta_{31} & \Theta_{32} \\ \Theta_{81} & \Theta_{82} \end{bmatrix}.$$

Per costruire le matrici $A_1, A_2, \dots, A_{n_a}, B_0, B_1, \dots, B_{n_b-1}$, si può implementare un ciclo dove A_1^T è composta dai primi quattro elementi di Θ , A_2^T dai secondi quattro, e così via fino a $B_{n_b-1}^T$ che sarà dato dagli ultimi quattro elementi.

$$\text{Nell' esempio } A_1^T = \begin{bmatrix} \Theta_{11} & \Theta_{12} \\ \Theta_{21} & \Theta_{22} \end{bmatrix}, A_2^T = \begin{bmatrix} \Theta_{31} & \Theta_{32} \\ \Theta_{41} & \Theta_{42} \end{bmatrix},$$

$$B_0^T = \begin{bmatrix} \Theta_{51} & \Theta_{52} \\ \Theta_{61} & \Theta_{62} \end{bmatrix}, B_1^T = \begin{bmatrix} \Theta_{71} & \Theta_{72} \\ \Theta_{81} & \Theta_{82} \end{bmatrix}.$$

Una volta ricavate $A_1, A_2, \dots, A_{n_a}, B_0, B_1, \dots, B_{n_b-1}$, si può ricavare la struttura come

$$y(t) = - \sum_{k=1}^{n_a} A(k)y(t-k) + \sum_{k=0}^{n_b-1} B(k)u(t-k-1) + e(t).$$

9 Scelta di n_a e n_b tramite minimizzazione dell'errore

Fino ad ora, il modello è stato definito utilizzando n_a e n_b generici. In questa sezione, si vuole capire come scegliere questi due parametri, quindi si vuole determinare il grado del sistema ARX.

Fino ad ora, si sono utilizzati i dati di training per fare identificazione. Questi dati sono utilizzati esclusivamente in fase di addestramento del modello. Ovvero il nostro modello ha imparato le relazioni tra le nostre U (le variabili di input) e Y (ovvero il target, l'output).

Per evitare l'overfitting, per avere quindi una reale capacità predittiva, diamo al nostro modello dei dati che non ha mai visto e gli diciamo di fare una previsione. Overfitting vuol dire essere in grado di predire perfettamente i dati utilizzati in fase di addestramento, ma non essere in grado di generalizzare su dati nuovi. Questa fase viene detta **fase di validazione**. Per fare questo, è necessario avere il dataset di validazione.

Si simula quindi l'equazione differenziale in Matlab tramite il comando `ode45` inserendo un intervallo di tempo che va da 0 a 40 secondi (o anche di più).

Si divide poi il dataset generato in due parti, la prima che serve per l'identificazione (quello utilizzato precedentemente), che serve per calcolare le matrici A_1, A_2, \dots, A_{n_a} ,

$B_0, B_1, \dots, B_{n_b-1}$, e che va da 0 a 20 secondi e la seconda parte, quella di validazione che va da 20 a 40 secondi che serve per calcolare i valori veri negli istanti di campionamento lungo il tempo di validazione.

La nuova funzione `ode45` utilizza gli stessi ingressi della precedente con l'estensione delle onde quadre, visibili in figura 14, da 0 a 40 secondi.

Le velocità in risposta a tali ingressi sono disegnate in Figura 16

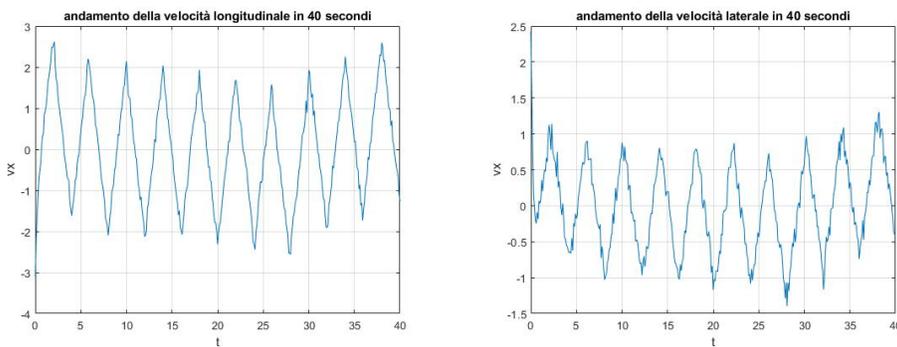


Figure 16: Uscite in 40 secondi

Si noti i primi venti secondi costituiscono i dati utilizzati per l'identificazione, i secondi 20 secondi i dati per la validazione.

Per distinguerli chiameremo questi dati y^v e u^v dove

$$u^v = \begin{bmatrix} a_x^v(t) \\ a_y^v(t) \end{bmatrix} \text{ e } y^v(t) = \begin{bmatrix} v_x^v(t) \\ v_y^v(t) \end{bmatrix}.$$

Il predittore ad un passo del modello considerato è

$$y^v(t|t-1) = - \sum_{k=1}^{n_a} A(k)y^v(t-k) + \sum_{k=0}^{n_b-1} B(k)u^v(t-k-1)$$

il quale fornisce una stima di $y^v(t)$ all'istante $t-1$.

Quindi per trovare i valori di n_a e n_b che meglio rappresentano il modello bisogna minimizzare l'errore

$$e = y^v(t) - y^v(t|t-1) .$$

Dati diversi valori di n_a e n_b si devono trovare quelli che minimizzano l'indice di costo dato dalla relazione

$$\sum_{t=1}^{N_v} \frac{1}{N_v} \|y^v(t) - y^v(t|t-1)\|^2$$

che costituisce l'errore quadratico medio, dato dalla differenza tra il valore del dato di validazione e il valore dato dalla previsione all'istante t dove N_v rappresenta il numero di campioni dei dati di validazione. Questa tecnica è chiamata **cross-validation**. La Cross Validation (o validazione incrociata) è infatti una tecnica statistica che permette di usare in modo alternato i dati sia per il train che per il test. Spesso viene chiamata k-fold cross validation perché diviso il dataset iniziale in una serie di porzioni uguali di dati (k-campi) e ne uso iterativamente un tot per il train e un tot per il test.

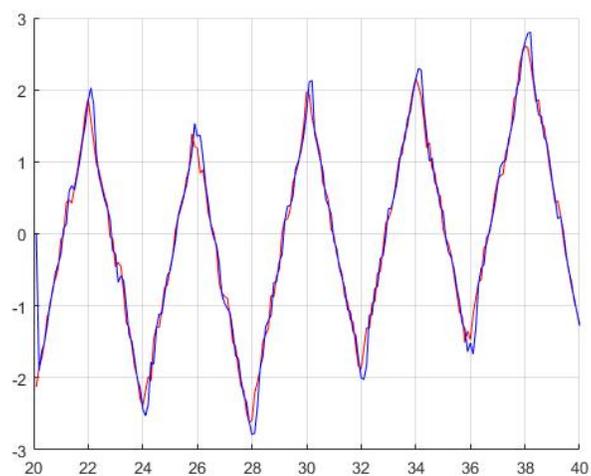


Per fare questo , si costruisce una funzione su Matlab in cui si calcola l'errore quadratico medio e si individua la coppia di valori n_a e n_b che minimizzano l'indice di costo. Il valore di n_a e n_b influisce sull'errore poichè genera le matrici $A_1, A_2, \dots, A_{n_a}, B_0, B_1, \dots, B_{n_b-1}$ che vengono utilizzate per calcolare $y^v(t|t-1)$.

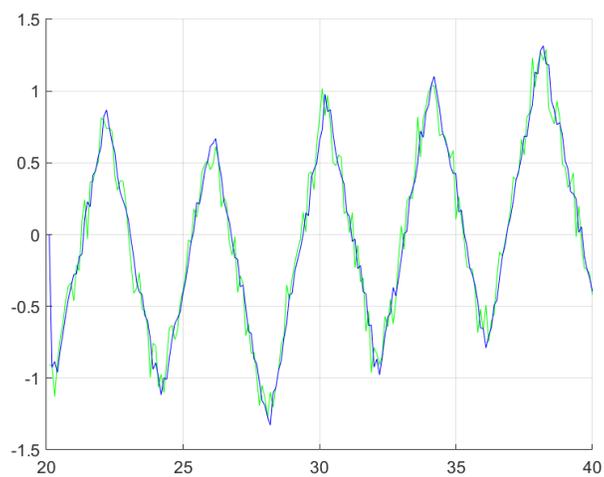
Dopo vari tentativi l'errore viene minimizzato quando $n_a = 3$ e $n_b = 2$ in cui vale circa 0.048.

Confrontando il comportamento di $y^v(t) = \begin{bmatrix} v_x^v(t) \\ v_y^v(t) \end{bmatrix}$ e $y_{predict}(t) = \begin{bmatrix} v_{xpredict}(t) \\ v_{ypredict}(t) \end{bmatrix}$

si ottengono i seguenti grafici



dove la funzione in rosso è $v_x^v(t)$ e la funzione in blu è $v_{xpredict}(t)$



dove la funzione in verde è $v_y^v(t)$ e la funzione in blu è $v_{ypredict}(t)$.

10 Conclusioni

Partendo da un modello che descrive il comportamento di un veicolo, nel suo centro di massa, mentre sta compiendo una curva si è identificata la l' equazione matematica del sistema dinamico a tempo discreto che meglio rappresenta il modello. Per fare questo è servita la fase di addestramento in cui sono state individuate le matrici A e B che costruiscono la struttura del modello ARX tramite l'uso dello stimatore a massima verosomiglianza.

Per individuare gli ordini corretti del modello ARX si è minimizzato la quantità l' errore quadratico medio di predizione. Si è ottenuto quindi che il modello è descritto verosimilmente da

$$y(t) = -A_1y(t-1) - A_2y(t-2) - A_3y(t-3) + B_0u(t-1) + B_1u(t-2) + e(t)$$

$$\text{dove } A_1 = \begin{bmatrix} -0.6994 & 0.0680 \\ 0.0059 & -0.4405 \end{bmatrix}, A_2 = \begin{bmatrix} -0.1201 & -0.0237 \\ -0.0657 & -0.0445 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} -0.1318 & -0.1166 \\ -0.1298 & -0.1380 \end{bmatrix},$$

$$B_0 = \begin{bmatrix} 0.0995 & -0.0982 \\ 0.279 & -0.214 \end{bmatrix}, B_1 = \begin{bmatrix} 0.0118 & -0.1021 \\ 0.0106 & 0.0118 \end{bmatrix}.$$