

POLITECNICO DI MILANO

FACOLTÀ DI INGEGNERIA INDUSTRIALE

CORSO DI LAUREA IN INGEGNERIA MECCANICA



Algoritmo di identificazione Real-Time dei parametri modali per il controllo adattativo di una smart structure

Relatore: *Prof. Ferruccio Resta*

Correlatore: *Ing. Francesco Ripamonti*

Tesi di Laurea di:

Massimo Borroni

matr. 754725

Anno Accademico 2012-2013

Ottobre 2013

Indice

1	Introduzione	1
2	Stato dell'arte	5
2.1	Controllo Adattativo	5
2.1.1	Controlli Model based	6
2.1.2	Controlli Rule-Based	8
2.1.3	Controlli non Model-Based	8
2.2	Identificazione Real-Time	8
3	Algoritmo di identificazione RT	15
3.1	Subspace Tracking (ST)	16
3.2	Il modello ARMA	27
4	Identificazione RT - Prove sperimentali	33
4.1	Il Banco prova	33
4.2	Applicazione sperimentale di ST e ARMA	36
4.3	La gestione dei parametri identificati	47
4.4	Parametri modali identificati RT	52
4.4.1	Disturbo random 0-100 Hz	52
4.4.2	Disturbo random 0-50 Hz e rumore rosa	54
5	Logiche di controllo adattativo	59
5.1	Richiami sul controllo modale	61
5.2	Controllo ottimo LQR	63

5.3	Controllo LQR adattativo con 3 modi controllati	65
5.3.1	Forzante data dalla somma di 3 sinusoidi in risonanza	67
5.3.2	Cambio di forzante	69
5.3.3	Forzante fuori dalla risonanza	69
5.4	Controllo LQR adattativo con 17 modi controllati	70
5.4.1	Forzante random sul sistema modificato	72
5.4.2	Controllo Gain Scheduling su 17 modi	75
5.5	Controllo adattativo - Prove sperimentali	76
5.5.1	LQR con aggiornamento della matrice di stato	76
5.5.2	Gain Scheduling, 3 modi controllati	81
	Conclusioni	85
5.6	Sviluppi per il futuro	87
A	Algoritmo genetico	89
B	Approfondimenti sulla SVD	93

Elenco delle figure

1.1	Schema a blocchi di un generico sistema di controllo	1
1.2	Schemi di controllo	2
1.3	Configurazione base di un controllo adattativo	3
2.1	Schema a blocchi MRAC	7
2.2	Gain Scheduling	7
2.3	Semplice neurone a più ingressi	11
2.4	Rete Multi-layer	12
3.1	Prima sequenza di rotazioni di Givens	21
3.2	Seconda sequenza di rotazioni di Givens	23
3.3	Segnale a 3 componenti con rumore bianco	26
3.4	Uscita del Subspace Tracking	27
3.5	Modello di identificazione con ingresso $u(t)$, uscita $y(t)$ e rumore bianco (ARMAX)	28
3.6	Modello ARMA: in ingresso c'è solamente rumore bianco	28
4.1	Funzione di trasferimento filtro, 0-500 Hz	36
4.2	Piastra, lato A	37
4.3	Piastra, lato B	39
4.4	Segnale random	40
4.5	Segnale random, spettro di frequenze	41
4.6	Prima storia temporale in uscita dal ST	41
4.7	Prima storia temporale del ST elaborata dall'ARMA	42

4.8	Seconda storia temporale in uscita dal ST	42
4.9	Seconda storia temporale del ST elaborata dall'ARMA	43
4.10	Terza storia temporale del ST	43
4.11	Terza storia temporale del ST elaborata dall'ARMA	44
4.12	Scambio di frequenze identificate	44
4.13	Terza storia temporale del ST elaborata dall'ARMA - oblio adattativo .	45
4.14	Quarta storia temporale del ST elaborata dall'ARMA - senza oblio adattativo	45
4.15	Quarta storia temporale del ST elaborata dall'ARMA - oblio adattativo .	46
4.16	Esempio di contatore bidimensionale per le coppie frequenza-smorzamento identificati dall'ARMA(3,2)	48
4.17	Contatore bidimensionale: frequenze vicine	48
4.18	Contatore bidimensionale con griglia grossa e piccola	50
4.19	Contatore bidimensionale: caso particolare	51
4.20	Rumore Rosa, fft a valle del filtro elettronico	55
4.21	Curva di Rayleigh interpolata	57
4.22	Valori identificati real-time per una singola frequenza	58
5.1	Diagramma schema di controllo a 3 modi controllati	67
5.2	Somma di 3 sinusoidi in risonanza	68
5.3	Cambio di forzante	69
5.4	Forzante fuori risonanza	70
5.5	Diagramma del controllo LQR adattativo a 17 modi	72
5.6	Forzante random 0-100 Hz, matrice di stato modificata. Controllo LQR adattativo.	73
5.7	Forzante random 0-100 Hz, matrice di stato modificata. Controllo LQR non adattativo.	74
5.8	Forzante random 0-100 Hz, matrice di stato non modificata. Controllo LQR adattativo.	74
5.9	Diagramma controllo Gain Scheduling. 17 modi controllati.	76
5.10	Forzamento random 0-100 Hz con matrice di stato modificata. Controllo adattativo Gain Scheduling. 17 modi controllati	77

5.11	Prove sperimentali. Forzante random 0-100 Hz. Controllo adattativo . . .	79
5.12	Prove sperimentali. Forzante random 0-100 Hz. Controllo non adattativo	80
5.13	Prove sperimentali. Forzante random 0-100 Hz. Controllo adattativo . . .	82
5.14	Prove sperimentali. Forzante random 0-100 Hz. Controllo non adattativo	83
5.15	Prove sperimentali. Forzante random 0-50 Hz. Controllo adattativo . . .	83
5.16	Prove sperimentali. Forzante random 0-50 Hz. Controllo non adattativo	84
A.1	Sequenza di operazioni di un algoritmo genetico	90

Elenco delle tabelle

2.1	Riassunto metodi di identificazione	13
4.1	Caratteristiche smart structure	34
4.2	Caratteristiche patch piezoelettriche	34
4.3	Caratteristiche estensimetri	34
4.4	Caratteristiche accelerometri	35
4.5	Frequenze presenti nel segnale random	38
4.6	Scelta di m in funzione della frequenza minima e di f_s	46
4.7	Riassunto dei parametri caratteristici dell'algorithm di identificazione .	47
4.8	Parametri modali della piastra identificati offline	53
4.9	Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: random 0-100 Hz. Estensimetro n1.	53
4.10	Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: random 0-100 Hz. Accelerometro n1.	54
4.11	Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: random 0-50 Hz. Estensimetro n1.	55
4.12	Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: rumore rosa. Estensimetro n1.	56
5.1	Parametri modali della piastra identificati offline	60
5.2	Parametri modali della piastra identificati offline	73
5.3	Modi controllati	78

Abstract

A new method, for the Real-Time identification of the modal parameters, is presented. An ARMA model processes the signal decomposed with the Subspace tracking, and a statistical approach is used to achieve convergence to a value of natural frequency and damping ratio: it uses a dynamic discretization of the frequency range to identify. The real-time identification algorithm is tested to estimate the bounds of applicability limits, and it's used for the experimentation of different adaptive control logics aiming to reduce vibrations in a smart structure, represented a carbon fiber plate. A piezoelectric actuator introduces vibrations in the structure while data are acquired by three accelerometers and three strain gage given to input to the state observer. On the other side, real-time identification needs only one input channel. Three piezoelectric actuators are added to the plate in order to provide the control force. With the identification of the frequency of the disturbance force, it's possible to select the optimal gain matrix for the model-based controller, in order to minimize the control forces acting on a subset of the eigenmodes of the system. An adaptive LQR control logic is also presented. The algebraic Riccati equation is solved using, under certain assumptions, an updated state matrix: the update is performed with the modal parameters identified by the previously described algorithm. The same algorithm is used for a Gain scheduling control logic, allowing the choice of the optimal gain matrix considering the operating system conditions.

ESTRATTO

Nella presente tesi viene presentato un nuovo metodo per l'identificazione Real-Time dei parametri modali di una smart structure. Un modello ARMA elabora il segnale scomposto dall'algoritmo Subspace Tracking, allo scopo di raggiungere la convergenza a un valore di frequenza propria e smorzamento adimensionale. Il sistema di identificazione presentato viene implementato sperimentalmente per testarne i limiti applicativi, e successivamente utilizzato per analizzare diverse logiche di controllo adattativo per ridurre le vibrazioni di una smart structure (piastra in fibra di carbonio). In particolare l'intero algoritmo viene utilizzato per l'identificazione del disturbo agente sul sistema e la conseguente scelta della migliore matrice dei guadagni per il controllore, nell'ottica di minimizzare la forze di controllo agendo solamente su un sottoinsieme dei modi di vibrare del sistema (i più sollecitati). Viene inoltre presentata la logica di controllo LQR adattativo, dove l'equazione algebrica di Riccati viene risolta utilizzando una matrice di stato che in certe condizioni può essere aggiornata con i parametri modali identificati dall'algoritmo appena descritto. Infine l'algoritmo di identificazione real-time viene utilizzato per una logica di controllo Gain Scheduling, permettendo la scelta della matrice dei guadagni più adatta alle condizioni in cui si trova il sistema.

Capitolo 1

Introduzione

In ingegneria, le vibrazioni sono un effetto indesiderato che porta a fenomeni come la diminuzione della vita a fatica del componente meccanico, e la generazione di rumore acustico causato da fenomeni vibratorii ad alta frequenza. A volte la presenza di vibrazioni è dovuta alla progettazione inadeguata di un prodotto, o all'uso di una macchina in un modo imprevisto [1]. Nell'ipotesi che il processo di progettazione sia stato eseguito correttamente, è possibile ugualmente che si manifestino vibrazioni indesiderate, che possono essere ridotte con un opportuno sistema di controllo. Nella teoria dei controlli, controllare un sistema significa fornire determinati ingressi in modo tale che il comportamento dinamico risultante soddisfi le specifiche imposte dal progettista. Le prestazioni del sistema controllato possono quindi essere valutate attraverso lo studio dell'anello di controllo.

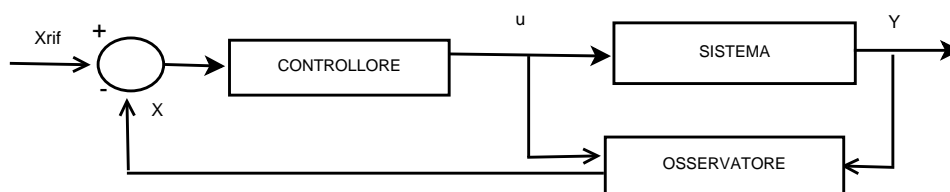
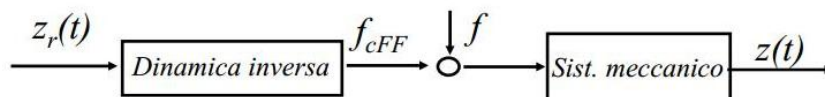


Figura 1.1: Schema a blocchi di un generico sistema di controllo

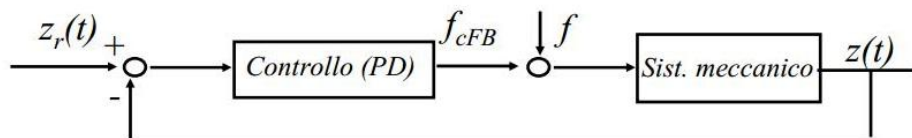
Gli schemi di controllo consolidati sono essenzialmente due: il controllo in feedforward e il controllo in feedback. Adottando un controllo di tipo feedforward, si risolve un problema di dinamica inversa, calcolando le azioni di controllo sulla base di un'uscita di

riferimento desiderata. Il vantaggio di questo sistema, nel caso di sistemi non controllati stabili, è che non ne viene modificata la stabilità. Uno svantaggio consiste nel fatto che è necessario disporre di un modello matematico del sistema sufficientemente preciso, al fine di ottenere prestazioni accettabili. Con un controllo di tipo feedback, il segnale di riferimento viene confrontato con la variabile in uscita dal sistema. La differenza tra valore desiderato e valore misurato, viene usata per il calcolo della forza di controllo che verrà esercitata sul sistema. Tale differenza viene dunque moltiplicata per il guadagno scelto dal progettista [2]. Infine, nel caso di controllo adattativo, il guadagno della legge di controllo subisce variazioni nel tempo.

- *Controllo in FEED FORWARD*



- *Controllo in FEED BACK*



- *Controllo in FEED FORWARD + FEED BACK*

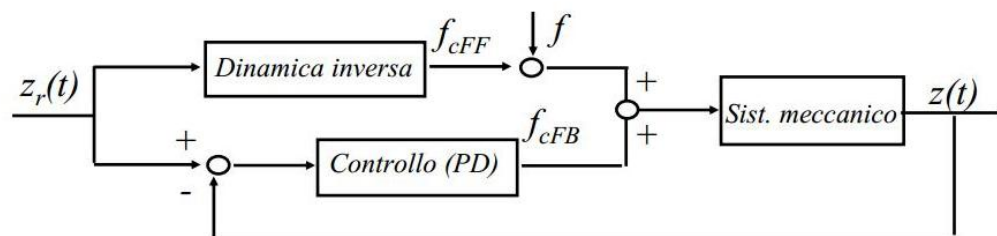


Figura 1.2: Schemi di controllo

Il controllo adattativo è un insieme di tecniche per la regolazione automatica dei controllori in tempo reale, al fine di ottenere o mantenere un livello desiderato delle prestazioni del sistema di controllo quando i parametri del modello dinamico sono sconosciuti o variano nel tempo. I sistemi di controllo ad alte prestazioni possono richiedere infatti una regolazione precisa del controllore, e in tal senso le tecniche di controllo adattativo forniscono un approccio sistematico per la regolazione automatica dei parametri.

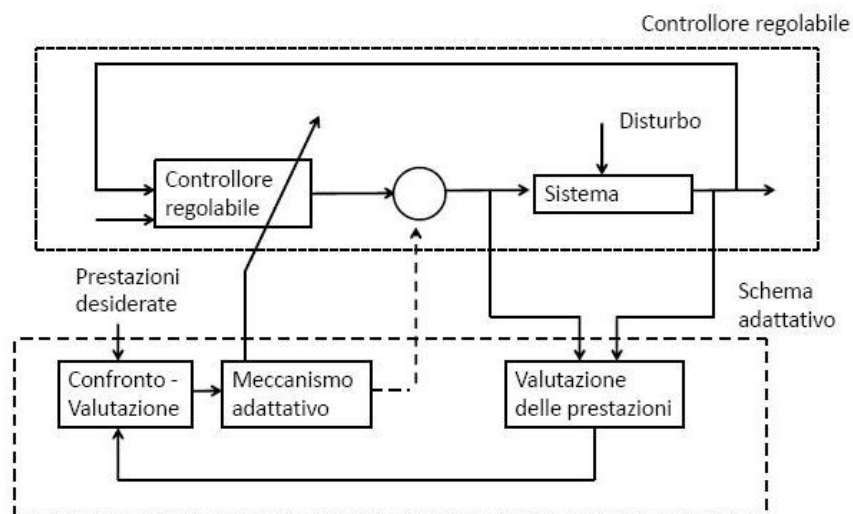


Figura 1.3: Configurazione base di un controllo adattativo

In figura 1.3 è rappresentata una configurazione tipica di un sistema di controllo adattativo.

Qualora il sistema di controllo si basa su un modello numerico del sistema, è importante che quest'ultimo venga aggiornato nel caso in cui il sistema subisca delle variazioni nei parametri modali, cioè se le sue frequenze proprie e smorzamenti adimensionali sono cambiati: più è precisa la conoscenza delle matrici di stato del sistema, e più il controllo è efficiente.

Partendo da queste considerazioni, il presente lavoro di tesi è volto al controllo di vibrazioni in una smart structure costituita da una piastra in fibra di carbonio, sulla quale sono collegati degli attuatori piezoelettrici per fornire la forza di controllo. Sulla piastra sono collegati accelerometri ed estensimetri per la misurazione delle vibrazioni e la stima dello stato. La logica di controllo che si vuole implementare è di tipo adattativo e si basa sulla conoscenza del modello numerico della piastra. È quindi necessario sviluppare un sistema di identificazione Real-Time (RT) grazie al quale il controllore può inseguire le variazioni dei parametri modali. Dopo aver esposto lo stato dell'arte nel controllo adattativo e negli algoritmi di identificazione real-time (dando una maggior enfasi su quest'ultimo punto), verrà presentato un sistema di identificazione real-time, opportu-

namente modificato per poter essere utilizzato in una logica di controllo adattativo. Tale algoritmo verrà implementato sperimentalmente per testarne i limiti applicativi, variando tipologia di disturbo e posizione dell'unico sensore di acquisizione di cui l'algoritmo fa uso. In base ai risultati così ottenuti verranno presentate differenti logiche di controllo adattativo che fanno uso del sistema di identificazione precedentemente esposto. Le logiche di controllo verranno poi testate numericamente per diverse forzanti di disturbo. Infine si procederà ad una sperimentazione della logica di controllo ritenuta più adatta fra quelle presentate.

Capitolo 2

Stato dell'arte

In questo capitolo verranno inizialmente presentate le logiche di controllo adattativo presenti in letteratura, e successivamente verrà approfondito il problema dell'identificazione dei parametri modali di un sistema in tempo reale, di cui il controllo adattativo fa uso per inseguire le variazioni del sistema. Verrà quindi esposto con una maggior enfasi lo stato dell'arte relativo ai sistemi di identificazione real-time, enunciando pregi e difetti degli algoritmi presenti in letteratura.

2.1 Controllo Adattativo

Secondo la definizione di Astrom [3], un controllo adattativo è una logica nella quale è presente un controllore caratterizzato da guadagni che possono essere aggiornati nel tempo, con l'obiettivo che le prestazioni del sistema controllato non degradino in presenza della comparsa di disturbi o, soprattutto, di modifiche del sistema meccanico durante il funzionamento [4]. Esso si compone di una regola adattativa (o legge adattativa) e di una legge di controllo. Per regola adattativa si intende il meccanismo che rende possibile l'inseguimento nel tempo delle caratteristiche del sistema, mentre per legge di controllo si intende la logica con cui vengono calcolati i guadagni della forza di controllo. La combinazione tra leggi di controllo e regole adattative, dà luogo ad un'ampia classe di controllori adattativi con differenti proprietà. Una distinzione basilare è tra leggi di controllo che necessitano della conoscenza di un modello numerico

del sistema (Model Based) e controllori che non hanno la necessità di conoscerlo (non Model Based) [5]. Fra le leggi adattative si possono distinguere quelle che mirano a minimizzare una certa funzione di errore tra i parametri di riferimento misurati e quelli predetti (metodi di sensibilità) e quelle basate sulla stabilità di Lyapunov. In queste ultime il problema di progettazione di una legge adattativa è formulato come un problema di stabilità [5].

A titolo di esempio vengono illustrate alcune logiche di controllo adattativo presenti in letteratura.

2.1.1 Controlli Model based

Per l'implementazione di questa tipologia di controllo, è richiesta la conoscenza di un modello numerico del sistema. In questa categoria di controlli rientrano il Model Based Adaptive Control (MRAC) e il Gain Scheduling. Il MRAC forza il sistema reale ad avere lo stesso comportamento del sistema ideale descritto dal modello numerico, scegliendo opportunamente la forzante di controllo. Nello schema del sistema controllato (figura 2.1), il blocco che simula il modello del sistema è in parallelo all'anello costituito dal blocco controllore e dal sistema reale. Al modello matematico del sistema vengono date in ingresso tutte le forzanti note che agiscono sul sistema reale e l'uscita del sistema reale è confrontata con quella del modello. Dal confronto vengono stabiliti i nuovi guadagni della legge di controllo [5]. Solitamente la legge di controllo fornita al sistema reale è tale da minimizzare l'errore tra uscita reale e ideale, e far sì che il sistema controllato sia stabile (utilizzando ad esempio la teoria della stabilità di Lyapunov) [6].

Nel Gain Scheduling, invece, il controllore utilizza dei guadagni prestabiliti e tabulati. A seconda della condizione in cui si trova il sistema da controllare, verrà scelto il set di guadagni più adatto tra quelli tabulati.

Uno svantaggio implicito del Gain Scheduling, è che si può ottimizzare il controllore per un numero di condizioni operative finito e determinato a priori. Una ulteriore problematica è una discontinuità nella forza di controllo, che si verifica nella transizione da una configurazione tabulata ad un'altra [7]. Il Gain scheduling inoltre implica la cono-

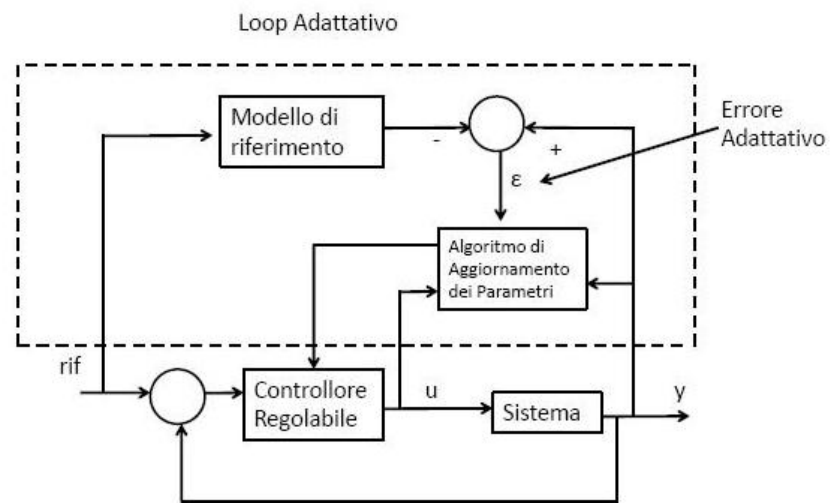


Figura 2.1: Schema a blocchi MRAC

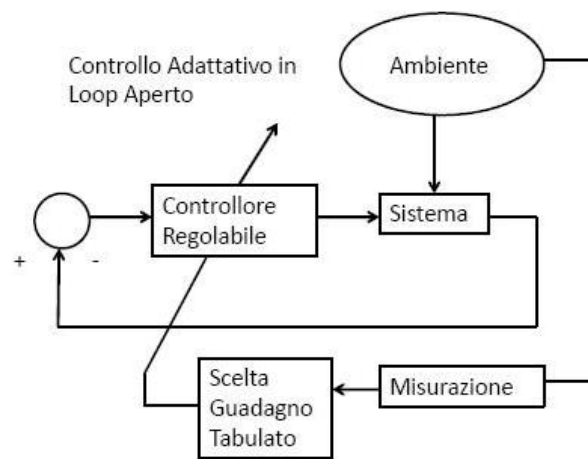


Figura 2.2: Gain Scheduling

scenza dello stato in cui il sistema sta operando, non disponendo di questa informazione, non è possibile realizzare questa logica di controllo.

2.1.2 Controlli Rule-Based

Le tecniche di controllo Rule Based (o sistemi esperti), si basano su una serie di regole prestabilite con le quali gestire le relazioni del tipo if/else/then tra lo stato attuale del sistema e la scelta dell'azione di controllo. La logica fuzzy, ad esempio, considerando le affermazioni oltre che vere o false anche parzialmente vere o parzialmente false, consente di identificare in classi di tipo qualitativo lo stato in cui si trova il sistema e di gestire le relazioni logiche per la scelta dell'azione di controllo più opportuna [4, 8].

2.1.3 Controlli non Model-Based

L'utilizzo di queste logiche di controllo, non prevede la conoscenza di un modello matematico del sistema. Per sintetizzare una legge di controllo, è possibile utilizzare le misure degli ingressi e delle uscite, senza elaborare un modello in forma esplicita. Una logica di questo tipo, per esempio, può essere implementata utilizzando un algoritmo di apprendimento che incrementa o diminuisce dei coefficienti di peso con l'obiettivo di minimizzare l'errore tra la grandezza di riferimento e quella misurata. Questi coefficienti vengono utilizzati nella definizione della legge di controllo. Inoltre vengono aggiornati passo-passo in maniera tale da considerare le modifiche sul comportamento del sistema provocate dall'applicazione della forza di controllo. I coefficienti così calcolati, non rappresentando nessuna particolare grandezza meccanica, non hanno in genere alcun significato fisico [4, 5].

2.2 Identificazione Real-Time

Nel controllo delle vibrazioni, la conoscenza continua dei parametri modali è fondamentale per la realizzazione di una legge di controllo adattativa che ha alla base un modello numerico del sistema (model based). Il sistema può essere soggetto a variazioni nel tempo delle sue risonanze, causate ad esempio dalla propagazione di una cricca, da

non linearità o dalle variazioni delle condizioni operative del sistema. Questo può comportare la variazione dei parametri modali, peggiorando le prestazioni di un controllore privo di intelligenza adattativa. Sulla base di questa considerazione, appare evidente che un passo fondamentale verso l'implementazione di un sistema di controllo adattativo, è l'identificazione delle grandezze di interesse del sistema meccanico, grazie alle quali realizzare la legge adattativa. Nel caso del controllo delle vibrazioni, è importante la conoscenza continua della matrice di stato, che contiene informazioni riguardo alle frequenze proprie del sistema e ai relativi smorzamenti adimensionali. Se la variazione di stato è repentina, non è possibile fare affidamento alle soluzioni solitamente utilizzate in un'analisi offline, come ad esempio l'analisi della funzione di risposta in frequenza del sistema (FRF) [4]. Tra le tecniche di identificazione in tempo reale, qualora ci siano solamente alcuni parametri incogniti, è possibile l'uso del filtro di Kalman esteso [9]. Al contrario, partendo dal presupposto che il sistema da identificare in real-time sia totalmente ignoto, ci si può affidare, ad esempio, a modelli stocastici detti *Box & Jenkins*, in cui il legame tra l'uscita del sistema, gli ingressi noti ed il rumore di disturbo è rappresentato attraverso due funzioni di trasferimento discrete. La prima funzione di trasferimento modella il comportamento del sistema rispetto agli ingressi deterministici, la seconda modella il comportamento del sistema rispetto al rumore di disturbo in entrata [4]. Le funzioni di trasferimento discrete, corrispondono a delle equazioni alle differenze finite nel dominio del tempo.

Fra i più noti modelli appartenenti alla famiglia *Box & Jenkins* ci sono i modelli AR, autoregressivi sull'uscita, ed i modelli ARX, autoregressivi sull'ingresso e sull'uscita. I modelli ARMA/ARMAX, aggiungono dei termini di media mobile rispetto ai rispettivi AR/ARX. Tutti questi metodi pesano i dati in ingresso con dei coefficienti, questi coefficienti si calcolano con il metodo dei minimi quadrati ricorsivi [10], ed è grazie ad essi che vengono stimati i parametri modali. Il numero di coefficienti di questi modelli identifica l'ordine del modello stesso. Nel caso di sistemi meccanici, si decide a priori il numero di modi che si vuole identificare, e la scelta dell'ordine coincide con il numero di poli e di zeri del sistema. Affinchè la stima sia corretta, è in genere necessario incrementare fittiziamente il numero di coefficienti del modello. Ripetendo la stima variando l'ordine, si identificano delle ricorrenze, cioè dei punti che sono rimasti stabili

nonostante l'ordine si sia modificato: è in prossimità di esse che si ricavano i parametri modali del sistema. Un maggiore approfondimento su queste tecniche di identificazione verrà condotto nel capitolo 3.

L'identificazione real time di un sistema meccanico, può anche essere realizzata mediante reti neurali [11]. Questa strategia è notevolmente differente rispetto ai modelli sopracitati. Le reti neurali artificiali sono nate per riprodurre attività tipiche del cervello umano come la percezione di immagini, il riconoscimento di forme, la comprensione del linguaggio, il coordinamento senso-motorio[12]. Come il cervello umano, le reti neurali artificiali hanno una notevole capacità di generalizzazione, e possono essere utilizzate per approssimare una qualsiasi funzione o sistema [13, 14]. Per riprodurre artificialmente il cervello umano occorre realizzare una rete di elementi molto semplici che sia una struttura distribuita massicciamente parallela, capace di apprendere e quindi di generalizzare. Tipicamente il neurone artificiale ha molti ingressi ed una sola uscita. L'attivazione del neurone è una funzione della somma pesata degli ingressi [12]. Le tipologie di reti neurali che sono state sviluppate sono diverse, e si classificano a seconda del tipo di connessione presenti tra i neuroni in due categorie: Feed-forward Networks e Recurrent networks. Alla prima famiglia appartengono le reti dove non esiste retroazione, cioè connessione tra ingresso e uscita di neuroni dello stesso livello o di livelli precedenti. Il segnale si propaga quindi dall'ingresso all'uscita, sommandosi e sovrapponendosi senza mai retrocedere. Nelle reti della seconda famiglia, al contrario esiste una retroazione e nascono quindi delle importanti caratteristiche dinamiche.

Ogni neurone opera nel seguente modo (figura 2.3): riceve un ingresso e fornisce un segnale di uscita che si propaga alle altre unità, secondo gli schemi precedentemente descritti. Il segnale in ingresso è dato dalla somma, pesata tramite un vettore di coefficienti \mathbf{W} , di un vettore di input \mathbf{p} con un coefficiente di offset o bias \mathbf{b} . Il risultato \mathbf{n} della sommatoria costituisce l'ingresso vero e proprio di una funzione \mathbf{f} , detta di attivazione, la cui uscita \mathbf{a} è l'uscita del neurone. L'ingresso del neurone è quindi:

$$n = \omega_{1,1}p_1 + \omega_{1,2}p_2 + \dots + \omega_{1,R}p_R + b \quad (2.1)$$

Le funzioni di trasferimento più comunemente usate sono la lineare, la sigmoide logaritmica $a = \text{logsig}(n)$, la funzione a gradino e la tangente iperbolica [11, 15, 16].

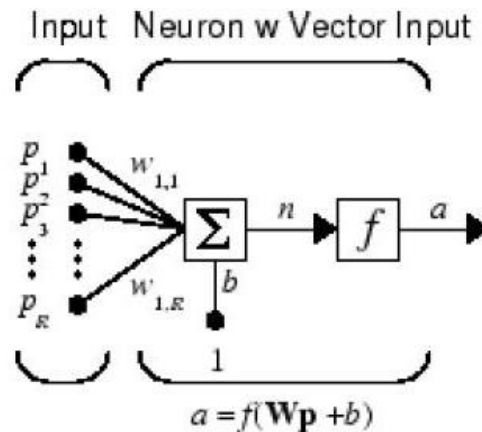


Figura 2.3: Semplice neurone a più ingressi

Quest'ultima è comunemente usata nei livelli nascosti di reti neurali multi livello. Nel controllo dei sistemi e nella loro identificazione, la tipologia di reti neurali più comunemente usata sono le reti neurali dinamiche. La rete dinamica ha memoria e la sua risposta nel tempo dipende non solamente dall'input attuale, ma dalla storia della sequenza di input ai passi precedenti [13]. Lo step successivo alla progettazione della rete neurale, consiste nel suo allenamento, che consiste nel fornire valori di ingresso e uscita noti permettendo di variare i parametri liberi della rete, pesi e offsets, in modo da adattarne le risposte ai valori voluti. Nelle reti di tipo multi-layer questa operazione viene effettuata seguendo la regola di apprendimento chiamata back-propagation. L'idea alla base di questa regola è quella di calcolare l'errore quadratico totale commesso dalla rete per poi propagarlo all'indietro attraverso i layer della rete. Questa procedura, ripetuta in maniera iterativa, permette infatti di stabilire quale parte dell'errore complessivo sia da attribuire a ciascun neurone. I pesi e l'offset vengono quindi variati in maniera proporzionale al negativo della derivata dell'errore rispetto ad ogni peso o offset secondo un algoritmo come il gradiente coniugato. Un limite di questo metodo è la possibile convergenza a minimi locali. Si può dimostrare che è sufficiente una rete neurale con n_1 ingressi, n_2 neuroni nell'unico layer nascosto (gli apici (1) e (2) nell'equazione 2.2 indicano rispettivamente il primo e il secondo livello della rete neurale) ed un'uscita per approssimare una qualsiasi funzione continua $g : R^n \rightarrow R$ col grado di precisione

desiderato. Sia infatti V l'insieme delle funzioni $R^n \rightarrow R$, continue e definite su un insieme compatto $S \subset R^n$. Si consideri il sottoinsieme A , rappresentante l'insieme delle funzioni di V ottenibili con una rete multi-layer al variare dei vettori di pesi $w^{(1)}$, $w^{(2)}$ e del vettore di offset $b^{(1)}$

$$A = \left\{ f \in V : f(x) = \sum_{k=1}^n 2w_k^{(2)} f_l \left(\sum_{j=1}^n w_{jk}^{(1)} x_j - b_k^{(1)} \right) \right\} \quad (2.2)$$

vale allora il seguente teorema:

Teorema 2.1. *Comunque si fissi un $\varepsilon > 0$ e si scelga una funzione $g \in V$, esiste una funzione $f \in A$ tale che:*

$$\max_{x \in S} |g(x) - f(x)| < \varepsilon$$

È per questo motivo che in letteratura, le reti multi-layer (figura 2.4) vengono definite approssimatori universali.

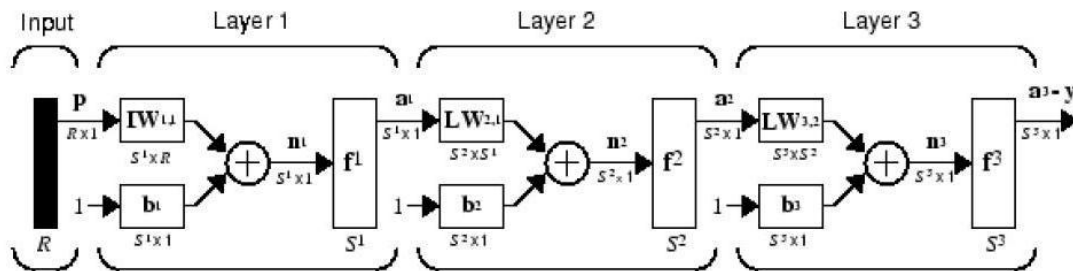


Figura 2.4: Rete Multi-layer

Alcune ricerche evidenziano che le reti neurali con due livelli nascosti forniscono generalmente le migliori prestazioni [15]. Il principale problema che si incontra durante l'allenamento di ogni tipologia di rete neurale, è l'overtraining (o sovra-allenamento). Questo fenomeno si ha quando la rete ha un sovrabbondante numero di neuroni e viene allenata eccessivamente: essa memorizza tutti i punti presentati ma perde notevolmente le capacità di generalizzazione [14]. Per questo motivo, dopo la fase di allenamento, viene effettuato un procedimento chiamato validazione incrociata (cross validation), che consiste nel testare la capacità di generalizzazione della rete neurale, fornendo ingressi e

confrontando l'uscita del sistema reale con l'uscita della rete. Potendo approssimare una qualsiasi storia temporale, le reti neurali possono essere usate per identificare un sistema meccanico, e ricavare frequenze proprie e smorzamenti. I parametri modali, si possono ricavare dai pesi che la rete neurale ha assunto per approssimare l'uscita del sistema [17]. Grazie alle reti neurali, è possibile inoltre controllare un sistema meccanico [13, 18].

La tabella 2.1 riassume i pregi e i difetti dei metodi di identificazione real-time presenti in letteratura:

METODO	PREGI	DIFETTI
AR/ARX	Semplici da implementare, basso costo computazionale	Poco robusti
ARMA/ARMAX	Aggiungono termini di media mobile rispetto agli AR/ARX	Più difficili da implementare e gestire rispetto agli AR/ARX
Reti Neurali	Possibilità di approssimare qualsiasi tipo di funzione o di modello	Difficili da gestire, molto sensibili al tipo di allenamento che è stato effettuato

Tabella 2.1: Riassunto metodi di identificazione

Capitolo 3

Algoritmo di identificazione RT

L'algoritmo di identificazione implementato nel presente lavoro di tesi è costituito da due macro algoritmi in serie, opportunamente modificati per un'ottimale identificazione dei parametri modali in tempo reale. I dati provenienti da un unico sensore di misura, vengono indirizzati all'algoritmo *Subspace Tracking* (ST), il cui compito è prendere una generica storia temporale, e dividerla in un certo numero di storie temporali monoarmonica. Ciascuna storia temporale monoarmonica in uscita dal *Subspace Tracking*, viene passata all'algoritmo ARMA, che stima frequenze proprie e smorzamenti adimensionali. Nell'algoritmo ARMA, viene calcolata una funzione di trasferimento che mette in relazione un ingresso qualsiasi del sistema, alla sua uscita. Il numero di coefficienti del modello ARMA ne identifica l'ordine, che nel caso di sistemi meccanici, coincide col numero di zeri e di poli del sistema. È proprio dal valore di questi coefficienti che vengono calcolati i parametri modali. Affinchè l'ARMA sia in grado di stimare correttamente i parametri modali del sistema meccanico, è necessario alzarne considerevolmente l'ordine, calcolando quindi un numero di poli e di zeri maggiori di quelli che si vogliono identificare. Questo procedimento fa sì che alcuni poli e zeri non modifichino la loro posizione nonostante l'ordine sia stato incrementato: i punti stabili vengono chiamati ricorrenze, ed è possibile risalire ai parametri modali tramite i poli ricorrenti. Un simile approccio è oneroso computazionalmente e non si presta a un calcolo real-time. Passando all'ARMA il segnale monoarmonica in uscita dal *Subspace Tracking*, è invece sufficiente un ARMA di ordine basso, rendendo possibile la

realizzazione di un procedimento ricorsivo per la stima dei parametri modali. Utilizzare un ARMA con ordine maggiore di 2, comporterebbe la nascita di poli fittizi e quindi da scartare. La scomposizione del segnale in storie temporali monoarmonica, ha inoltre il vantaggio aggiuntivo di filtrare il rumore nel segnale in ingresso. Infine nel caso il sistema sia forzato da una forzante monoarmonica conosciuta a priori, è possibile utilizzare un modello ARMAX, dove alla funzione di trasferimento (da determinare) che lega l'ingresso noto all'uscita del sistema, si sovrappone un'altra funzione di trasferimento (anch'essa da determinare) il cui ingresso è rumore bianco $\xi(t)$, per tenere conto di errori di misura e di modellazione. Sia l'algoritmo ARMA che il ST, sono stati adattati per un funzionamento real-time. I parametri modali in uscita dall'ARMA, nel caso di una misura reale elaborata, in determinate circostanze non sono corretti. Vengono dunque presentate delle migliorie per un'ottimale gestione dei 2 algoritmi (ST e ARMA), e viene inoltre proposta una strategia per la scelta dei parametri modali corretti fra quelli identificati in RT. Essa prevede che siano raccolti i dati in uscita dall'ARMA per un certo periodo di tempo, al termine del quale vengono elaborati per fornire frequenze proprie e smorzamenti adimensionali identificati. Il periodo di tempo deve essere tale da non pregiudicare l'inseguimento del sistema da parte dell'algoritmo, i cui risultati possono ancora essere considerati real-time. Nei paragrafi seguenti viene presentata la teoria di ST e ARMA.

3.1 Subspace Tracking (ST)

Lo strumento su cui si basa l'algoritmo del Subspace Tracking è la scomposizione di una matrice in valori singolari (SVD) [19]. Data una generica matrice reale $A \in \mathfrak{R}^{m \times n}$

$$A = USV^T \quad (3.1)$$

$U \in \mathfrak{R}^{m \times n}$ e $V \in \mathfrak{R}^{n \times n}$ matrici ortogonali, mentre $S \in \mathfrak{R}^{n \times n}$ è diagonale

$$S = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \end{pmatrix} \quad (3.2)$$

dove $\sigma_1 \dots \sigma_p$ sono i valori singolari di A , in ordine decrescente [20].

Le colonne di \mathbf{U} e di \mathbf{V} definiscono rispettivamente i vettori singolari sinistri e destri di A . I valori singolari sono le radici quadrate degli autovalori di $\mathbf{A}^T\mathbf{A}$, e la SVD può essere vista come una tecnica per ricostruire una matrice a partire dalle sue informazioni principali, permettendo quindi una semplificazione della matrice di partenza [21]. Per questa sua caratteristica, la SVD è usata frequentemente nell'informatica per l'elaborazione e la compressione delle immagini (al fine, ad esempio, di implementare un algoritmo per il riconoscimento automatico di volti [22]) o per algoritmi di Latent Semantic Indexing (LSI), grazie ai quali è possibile estrarre il contenuto concettuale da un testo [21].

La generica matrice A , avente rango r , può essere riscritta come combinazione lineare

$$A = E_1 + \dots + E_r \quad (3.3)$$

dove l' i -esima matrice E si calcola moltiplicando la matrice ottenuta dal prodotto tra l' i -esima colonna della matrice U e l' i -esima riga della matrice V^T per l' i -esimo valore singolare

$$E_i = \sigma_i U_i V_i^T \quad (3.4)$$

Qualora si abbiano a disposizione λ_i e \mathbf{U} dal calcolo degli autovalori e autovettori di $\mathbf{A}\mathbf{A}^T$, si può usare l'equazione:

$$E_i = \sqrt{\lambda_i} \mathbf{U}_i \mathbf{V}_i^T \quad (3.5)$$

dove

$$V_i = \frac{1}{\sqrt{\lambda_i}} X^T U_i \quad (3.6)$$

Tuttavia, dovendo operare in real-time, il primo passo per poter usare il ST per la scomposizione di un segnale nelle sue diverse componenti, è effettuare un'acquisizione dei dati relativi a una storia temporale, ed ordinarli in un vettore \mathbf{x}

$$\mathbf{x} = x_1, x_2, \dots, \dots, x_N$$

A questo punto i dati acquisiti possono essere ordinati nella matrice di Hankel, nella forma

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & \cdots & x_{N-m+1} \\ \vdots & \vdots & & \vdots \\ x_m & x_{m+1} & \cdots & x_N \end{pmatrix} \quad (3.7)$$

Può quindi essere calcolata la SVD della matrice \mathbf{X} .

Su un totale di p valori ordinati in ordine decrescente, i primi k sono relativi al segnale, i rimanenti sono relativi al rumore. Tale partizione é da effettuare anche nelle matrici \mathbf{U} e \mathbf{V} .

$$\mathbf{U} = [\mathbf{U}_s \quad \mathbf{U}_r] \quad (3.8)$$

$$\mathbf{V} = [\mathbf{V}_s \quad \mathbf{V}_r] \quad (3.9)$$

Ora utilizzando l'equazione 3.4 si possono calcolare le matrici \mathbf{E}_i , e grazie ad esse ricostruire le storie temporali.

L' i -esima componente $x_{i-ric}(t)$ è calcolata come

$$x_{i-ric}(t) = \frac{1}{t} \sum_{j=1}^t \mathbf{E}_{j,t-j+1} \quad 1 \leq t < m-1 \quad (3.10)$$

$$x_{i-ric}(t) = \frac{1}{m} \sum_{j=1}^t \mathbf{E}_{j,t-j+1} \quad m \leq t \leq N-m+1 \quad (3.11)$$

$$x_{i-ric}(t) = \frac{1}{N-t+1} \sum_{j=t-N-m+2}^{N-t+1} \mathbf{E}_{j,t-j+1} \quad N-m+1 < t \leq N \quad (3.12)$$

In questo modo, da una storia temporale qualsiasi, si sono ottenute diverse storie temporali idealmente monoarmonica. Il procedimento appena descritto può ora essere reso iterativo per una sua implementazione RT. È necessario costruire la matrice di Hankel in modo ricorsivo, dove ad ogni passo si aggiunge un vettore colonna mobile contenente gli ultimi m punti campionati. L'algoritmo iterativo [23, 24] parte con la conoscenza della matrice di Hankel \mathbf{X} in termini di scomposizione \mathbf{URV} , quindi

$$\mathbf{X} = \mathbf{URV} \quad (3.13)$$

dove $U \in \mathfrak{R}^{mxk}$, $V \in \mathfrak{R}^{kxk}$ e $R \in \mathfrak{R}^{kxk}$ [25]. Il vettore colonna mobile \mathbf{b} moltiplica la matrice \mathbf{U} rappresentante il sottospazio del segnale. In questo modo viene effettuata una proiezione del vettore \mathbf{b} sul sottospazio \mathbf{U}

$$\mathbf{r} = \mathbf{U}^T \mathbf{b} \quad (3.14)$$

Il vettore \mathbf{b}_c rappresenta il residuo di \mathbf{b}

$$\mathbf{b}_c = \mathbf{b} - \mathbf{U}\mathbf{r} \quad (3.15)$$

Normalizzando \mathbf{b}_c si ha

$$\mathbf{u}_c = \frac{\mathbf{b}_c}{\text{norm}(\mathbf{b}_c)} = \frac{\mathbf{b}_c}{\rho} \quad (3.16)$$

Ora è possibile aggiungere il vettore colonna \mathbf{u}_c a destra della matrice \mathbf{U} , che diventa dunque di $k + 1$ colonne (la colonna aggiunta rappresenta il residuo dell'ultima iterazione). Il vettore \mathbf{r} viene invece aggiunto alla matrice \mathbf{R} , che tiene conto delle componenti considerate come rumore solamente con l'informazione relativa alla norma di \mathbf{b}_c . A questo punto, la matrice \mathbf{X} risulta essere

$$\mathbf{X} = \begin{bmatrix} \mathbf{U} & \mathbf{u}_c \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{r} \\ \mathbf{0} & \rho \end{bmatrix} \begin{bmatrix} \mathbf{V} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} = \bar{\mathbf{U}}\bar{\mathbf{R}}\bar{\mathbf{V}} \quad (3.17)$$

Prima di applicare la **SVD** alla matrice \mathbf{R} , si annulla la sua ultima colonna con 2 serie di rotazioni di Givens. Le rotazioni hanno l'obiettivo di proiettare le componenti del segnale contenute nell'ultima colonna di destra di \mathbf{R} (le informazioni sulle componenti considerate come rumore, sono rappresentate solamente dal valore di ρ) sul resto della matrice \mathbf{R} . Il risultato finale delle rotazioni di Givens permette di eliminare l'ultima riga e l'ultima colonna di \mathbf{R} , e l'ultima colonna di \mathbf{U} , in quanto rappresentano informazioni sul rumore. Inoltre, in questo modo, la dimensione delle matrici resta costante ad ogni iterazione e contiene le informazioni relative alle componenti ritenute interessanti del segnale. Affinchè l'equazione 3.13 sia verificata, è necessario ruotare anche le matrici \mathbf{U} e \mathbf{V} . La matrice \mathbf{R} può essere intesa come il sottospazio dei residui, tramite le rotazioni di Givens le componenti del segnale vengono proiettate su di essa. Viceversa, la matrice \mathbf{U} è considerata il sottospazio del segnale, e tramite le rotazioni di Givens, viene proiettato il residuo (della nuova iterazione) su essa (ci vuole qualche iterazione prima

che il residuo calcolato diminuisca, e il Subspace Tracking si sintonizzi sul segnale in ingresso).

Le rotazioni di Givens sono realizzate premoltiplicando un vettore o una matrice per una matrice di Givens, che è una matrice di rotazione. Dal punto di vista geometrico, significa ruotare di θ gradi il vettore o la matrice data sul piano di coordinate (i, k) . Ad esempio se \mathbf{x} appartiene a \mathfrak{R}^n viene premoltiplicato per una matrice di Givens $\mathbf{G}(\mathbf{c}, \mathbf{s})$, dove c ed s sono rispettivamente coseno e seno dell'angolo di rotazione, allora il vettore \mathbf{y} risultante

$$\mathbf{y} = \mathbf{G}(\mathbf{c}, \mathbf{s})\mathbf{x} \quad (3.18)$$

è tale che

$$\mathbf{y}_i = c\mathbf{x}_i + s\mathbf{x}_k \quad (3.19)$$

dove c ed s sono coseno e seno. Per annullare l' i -esima componente del vettore \mathbf{y} i valori di seno e coseno da calcolare e da inserire nella matrice di Givens si ricavano risolvendo

$$\begin{cases} cx_i + sx_k = 0 \\ s^2 + c^2 = 1 \end{cases} \quad (3.20)$$

Essendo coinvolte solo 2 componenti del vettore da ruotare, non è necessario calcolare l'angolo di rotazione. Chiamando $a = x_i$ e $b = x_k$ gli elementi del vettore, si ha

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix}^T \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} r \\ 0 \end{bmatrix} \quad (3.21)$$

Si distinguono due casi

- $|b| > |a|$

$$r = -a/b; s = \frac{1}{\sqrt{1+r^2}}; c = sr \quad (3.22)$$

- $|a| \geq |b|$

$$r = -b/a; c = \frac{1}{\sqrt{1+r^2}}; s = cr \quad (3.23)$$

Prima di procedere con le rotazioni di Givens, la matrice \mathbf{R} è in forma di matrice triangolare superiore, con gli elementi diagonali non nulli.

$$\begin{array}{c}
 \begin{bmatrix} \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \boxed{\times} & \odot \\ & & & & \times \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \boxed{\times} & \times & \odot \\ & & & \times & 0 \\ & & & & \times \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & \times \\ & \boxed{\times} & \times & \times & \odot \\ & & \times & \times & 0 \\ & & & \times & 0 \\ & & & & \times \end{bmatrix} \\
 \\
 \begin{bmatrix} \boxed{\times} & \times & \times & \times & \odot \\ & \times & \times & \times & 0 \\ & & \times & \times & 0 \\ & & & \times & 0 \\ & \times & \times & \times & \times \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & 0 \\ & \times & \times & \times & 0 \\ & & \times & \times & 0 \\ & & & \times & 0 \\ \times & \times & \times & \times & \times \end{bmatrix}
 \end{array}
 \quad
 \begin{array}{c}
 \mathbf{G}_{1d} = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & c_{1d} & s_{1d} \\ & & & -s_{1d} & c_{1d} \end{bmatrix} \\
 \\
 \mathbf{G}_{2d} = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & c_{2d} & & s_{2d} \\ & & & 1 & \\ & & -s_{2d} & & c_{2d} \end{bmatrix} \\
 \\
 \mathbf{G}_{3d} = \begin{bmatrix} 1 & & & & \\ & c_{3d} & & & s_{3d} \\ & & 1 & & \\ & & & 1 & \\ & -s_{3d} & & & c_{3d} \end{bmatrix} \\
 \\
 \mathbf{G}_{kd} = \begin{bmatrix} c_{kd} & & & & s_{kd} \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ -s_{kd} & & & & c_{kd} \end{bmatrix}
 \end{array}$$

Figura 3.1: Prima sequenza di rotazioni di Givens

In figura 3.1 e 3.2, gli elementi contrassegnati dai simboli \boxtimes e \odot sono quelli rispetto ai quali vengono calcolati coseno e seno (corrispondenti agli elementi a e b nell'equazione 3.21). Gli elementi in rosso sono tutti quelli modificati dopo aver applicato la rotazione. Ad ogni passaggio la matrice \mathbf{R} viene aggiornata finchè gli elementi dell'ultima colonna (ad eccezione dell'ultimo elemento in basso) diventano zero

$$\mathbf{R} \leftarrow \mathbf{R} \cdot \mathbf{G}_{id} \quad (3.24)$$

La seconda sequenza di rotazioni di Givens è rappresentata in figura 3.2, e ha l'obiettivo di rendere nulli gli elementi della colonna e della riga $k+1$, a parte l'ultimo elemento in basso a destra che rappresenta unicamente la componente legata al rumore.

In figura 3.2, i termini contrassegnati con \odot sono prossimi allo zero (gli stessi elementi erano zero al termine della prima sequenza di rotazioni di Givens). Ad ogni rotazione di Givens si ha

$$\mathbf{R} \leftarrow \mathbf{G}_{is}^T \cdot \mathbf{R} \quad (3.25)$$

Riassumendo si ha

$$\mathbf{R}^* = \mathbf{G}_s^T \mathbf{R} \mathbf{G}_d = \begin{bmatrix} R & 0 \\ 0 & \bar{\sigma}_n \end{bmatrix} \quad (3.26)$$

dove $\bar{\sigma}_n$ è il valore singolare medio del sottospazio del rumore. Le matrici \mathbf{G}_d e \mathbf{G}_s sono invece date da

$$\mathbf{G}_d = \prod_{i=1}^k \mathbf{G}_{id} \quad (3.27)$$

$$\mathbf{G}_s = \prod_{i=1}^k \mathbf{G}_{is} \quad (3.28)$$

Per rispettare l'equazione 3.17, è necessario aggiornare le altre 2 matrici della scomposizione \mathbf{URV}

$$\mathbf{U}^* = \mathbf{U} \mathbf{G}_s \quad (3.29)$$

$$\mathbf{V}^* = \mathbf{G}_d^T \mathbf{V} \quad (3.30)$$

$$\begin{array}{c}
 \begin{bmatrix} \boxed{\times} & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ \ominus & \times & \times & \times & \times \end{bmatrix} & \mathbf{G}_{1s} = \begin{bmatrix} c_{1s} & & & s_{1s} \\ & 1 & & \\ & & 1 & \\ -s_{1s} & & & 1 \\ & & & & c_{1s} \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & \oplus \\ & \boxed{\times} & \times & \times & \times \\ & & \times & \times & \times \\ 0 & \ominus & \times & \times & \times \end{bmatrix} & \mathbf{G}_{2s} = \begin{bmatrix} 1 & & & s_{2s} \\ & c_{2s} & & \\ & & 1 & \\ -s_{2s} & & & 1 \\ & & & & c_{2s} \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & \oplus \\ & \times & \times & \times & \oplus \\ & & \boxed{\times} & \times & \times \\ 0 & 0 & \ominus & \times & \times \end{bmatrix} & \mathbf{G}_{3s} = \begin{bmatrix} 1 & & & s_{3s} \\ & 1 & & \\ & & c_{3s} & \\ & & & 1 \\ & & -s_{3s} & \\ & & & & c_{3s} \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & \oplus \\ & \times & \times & \times & \oplus \\ & & \times & \times & \oplus \\ 0 & 0 & 0 & \boxed{\times} & \times \\ & & & \ominus & \times \end{bmatrix} & \mathbf{G}_{ks} = \begin{bmatrix} 1 & & & s_{ks} \\ & 1 & & \\ & & 1 & \\ & & & c_{ks} \\ & & & -s_{ks} \\ & & & & c_{ks} \end{bmatrix} \\
 \\
 \begin{bmatrix} \times & \times & \times & \times & \oplus \\ & \times & \times & \times & \oplus \\ & & \times & \times & \oplus \\ 0 & 0 & 0 & 0 & \oplus \\ & & & \times & \times \end{bmatrix}
 \end{array}$$

Figura 3.2: Seconda sequenza di rotazioni di Givens

A questo punto si possono eliminare l'ultima riga e colonna della matrice \mathbf{R} perdendo così le informazioni sul rumore, cioè sulle componenti con valore singolare minore del k -esimo valore singolare. Le matrici \mathbf{U}^* , \mathbf{R}^* e \mathbf{V}^* sono pronte per ripetere il procedimento coi nuovi dati che verranno aggiunti nel vettore mobile \mathbf{b} . Sono quindi corrispondenti alle matrici \mathbf{U} , \mathbf{R} e \mathbf{V} del successivo istante di campionamento.

Ora non resta che effettuare la *SVD* della matrice \mathbf{R}^* , che grazie alla ridotta dimensione k non risulta particolarmente onerosa dal punto di vista computazionale (viveversa, la versione off-line del ST prevede l'applicazione della *SVD* su una matrice di grosse dimensioni)

$$\bar{\mathbf{R}}^* = \mathbf{U}_R \mathbf{S}_R \mathbf{V}_R^T \quad (3.31)$$

quindi

$$\mathbf{X} = \mathbf{U}^* \mathbf{U}_R \mathbf{S}_R \mathbf{V}_R^T \mathbf{V}^* = \bar{\mathbf{U}} \bar{\mathbf{S}} \bar{\mathbf{V}}^T \quad (3.32)$$

La *SVD* della matrice \mathbf{R}^* fornisce le informazioni relative alle k componenti più importanti del segnale. È possibile ricostruire le matrici \mathbf{E}_i a partire dalla 3.32 usando la 3.3. Al fine di ricostruire ciascuna componente del segnale, fino a un totale di k componenti, è sufficiente conoscere l'elemento in basso a destra delle matrici \mathbf{E}_i , che essendo l'ultimo elemento, rappresenta direttamente il valore dell' i -esima componente per l'istante considerato.

$$x_{i-ric} = \sigma_i \bar{\mathbf{U}}(m, i) \bar{\mathbf{V}}^T(i, k+1) \quad (3.33)$$

l' x_{i-ric} così calcolato, è una delle k componenti del segnale in ingresso, che si è scelto di non considerare come rumore. In realtà, ciascuna delle x_{i-ric} non è sufficiente a ricostruire l' i -esima storia temporale, esse infatti vanno sommate a 2 a 2, possono essere immaginate come 2 parti distinte della stessa armonica (dal punto di vista matematico, i valori singolari di \mathbf{R}^* si presentano a coppie, ad eccezione che il segnale elaborato non abbia componente a media non nulla). Se ne deduce che da k sottospazi, si ricavano $k/2$ storie temporali. Quindi la prima storia temporale, quella a modulo maggiore, sarà data da

$$x_{11} = x_{1-ric} + x_{2-ric} \quad (3.34)$$

Facendo riferimento alle 3.5 e 3.6 si ricava

$$\bar{\bar{\mathbf{V}}}(i, k+1) = \frac{1}{\sigma_i} \mathbf{b}^T \bar{\bar{\mathbf{U}}}(\mathbf{1} : \mathbf{m}, i) \quad (3.35)$$

In questo modo non è necessario aggiornare $\bar{\bar{\mathbf{V}}}$, risparmiando tempo di calcolo.

Ad esempio, la prima storia temporale monoarmonica al generico istante t viene calcolata come

$$x_1(t) = \mathbf{b}^T \bar{\bar{\mathbf{U}}}(:, 1) \bar{\bar{\mathbf{U}}}(m, 1) + \mathbf{b}^T \bar{\bar{\mathbf{U}}}(:, 2) \bar{\bar{\mathbf{U}}}(m, 2) \quad (3.36)$$

Si noti come la consocenza della matrice di Hankel è implicita avendo a disposizione le matrici \mathbf{U} , \mathbf{R} e \mathbf{V} . Rimane da definire l'inizializzazione di queste 3 matrici

$$\begin{aligned} \mathbf{U} &= \mathbf{I} \in \mathfrak{R}^{mxk} \\ \mathbf{R} &= \mathbf{0} \in \mathfrak{R}^{kxk} \\ \mathbf{V} &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathfrak{R}^{k+1 \times k+1} \end{aligned} \quad (3.37)$$

Dopo che la finestra mobile si è riempita, l'algoritmo impiega qualche istante prima di andare a regime e fornire i risultati attesi.

Un'ultima considerazione è sul comportamento del ST qualora il segnale di ingresso abbia un valore medio non nullo: in questo caso alla componente con media non nulla corrisponde un solo valore singolare (e non una coppia di valori singolari), e questo compromette la ricostruzione delle singole storie temporali utilizzando il procedimento sopra esposto (equazione 3.36).

Sono necessarie delle considerazioni aggiuntive importanti soprattutto qualora il segnale non sia stazionario. Partendo dalla teoria, è infatti fondamentale impostare al meglio 2 parametri chiave del subspace tracking. Uno di questi è già stato introdotto, ed è la lunghezza della finestra mobile \mathbf{m} , l'altro parametro fondamentale per un funzionamento corretto del subspace tracking, è il coefficiente di oblio λ , che modifica il peso dei dati passati rispetto ai dati recenti. Il coefficiente di oblio si inserisce nella matrice \mathbf{R}

$$\mathbf{X} = \begin{bmatrix} \mathbf{U} & \mathbf{u}_c \end{bmatrix} \begin{bmatrix} \lambda \mathbf{R} & \mathbf{r} \\ \mathbf{0} & \rho \end{bmatrix} \begin{bmatrix} \mathbf{V} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} = \bar{\bar{\mathbf{U}}} \hat{\mathbf{R}} \bar{\bar{\mathbf{V}}} \quad (3.38)$$

Per descrivere meglio il funzionamento del ST, si ipotizza di dover scomporre un segnale costituito dalla somma di 3 armoniche a 3 differenti frequenze, al quale è stato

aggiunto del rumore bianco. Nell'ipotesi che improvvisamente il segnale subisca una variazione nella frequenza, la finestra mobile di lunghezza m , avrà al suo interno una parte di dati relativi alla situazione di partenza, e una parte di dati posteriori alla variazione del segnale in ingresso. Una simile situazione non farà lavorare bene l'algoritmo del subspace tracking. Quindi valori di m troppo elevati, rischiano di compromettere l'efficienza della scomposizione del segnale. Viceversa valori troppo bassi rischiano di non essere sufficienti affinché ci siano abbastanza informazioni per ricostruire l'armonica.

Si consideri il segnale

$$y_{s1} = 20 * \sin(10 * 2 * \pi * t + \pi/4) + 10 * \sin(20 * 2 * \pi * t + \pi/2) + 3 * \sin(50 * 2 * \pi * t) \quad (3.39)$$

a cui è stato aggiunto del rumore bianco. Il segnale campionato alla frequenza di 250 Hz è visibile in figura 3.3.

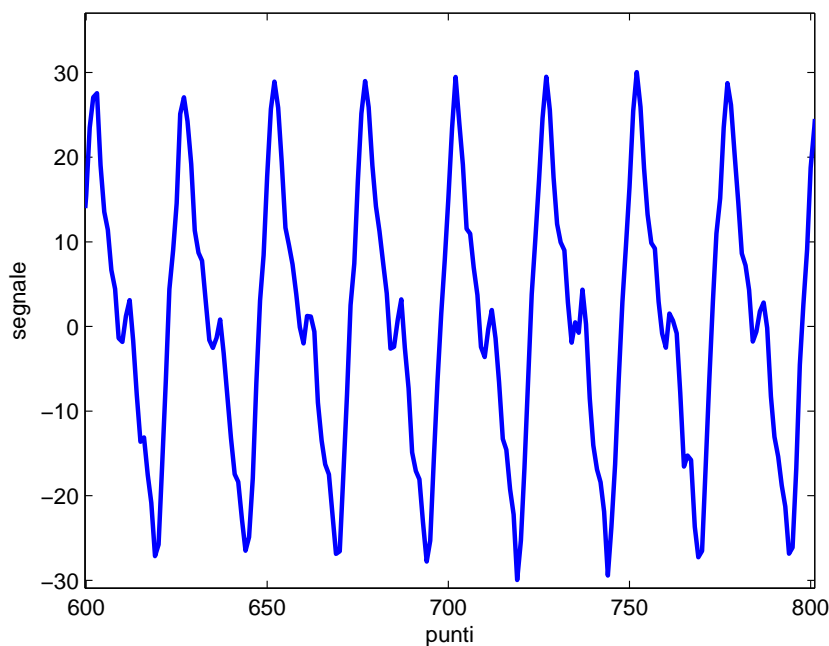


Figura 3.3: Segnale a 3 componenti con rumore bianco

Viene applicato l'algoritmo ST con $m = 200$, $\lambda = 0.99999995$ e $k = 10$. Si hanno 5

storie temporali in uscita dal ST, di cui solo 3 sono quelle delle sinusoidi in ingresso, le altre 2 componenti, che hanno una componente prossima a 0, sono rumore.

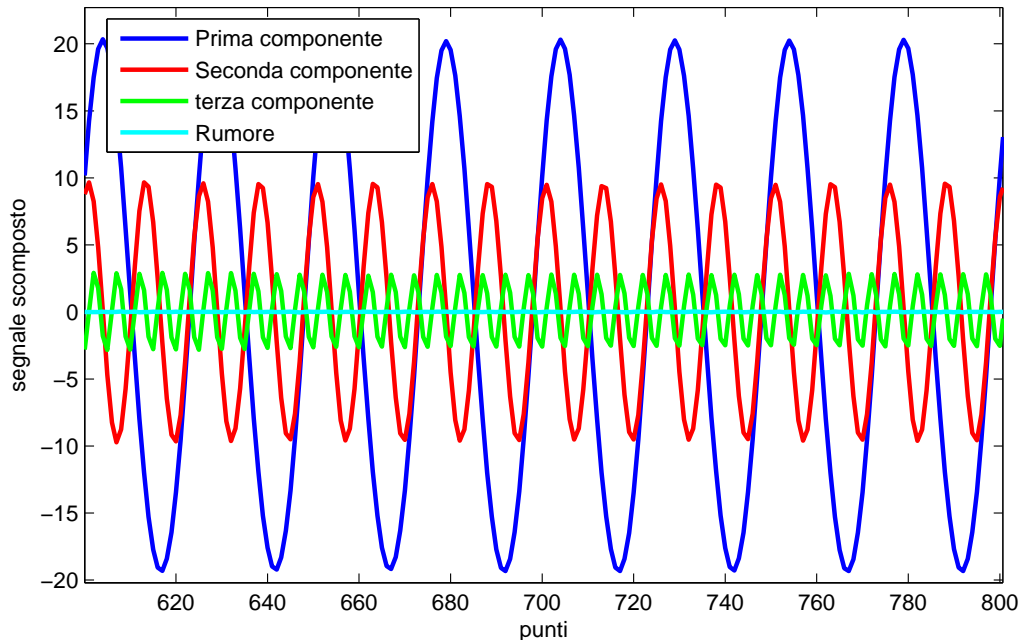


Figura 3.4: Uscita del Subspace Tracking

Dall'analisi della figura 3.4, si nota come l'algoritmo ha scomposto correttamente il segnale, e si sia comportato da filtro per il rumore. Delle tre storie temporali in uscita dal ST, la prima (quella ricavata a partire dalla prima coppia di valori singolari), è quella a modulo maggiore. Seguono in ordine decrescente di modulo, le altre storie temporali monoarmonica.

3.2 Il modello ARMA

Il modello ARMA/ARMAX è una tecnica di identificazione non strutturata, siccome per identificare i parametri del sistema meccanico, si basa esclusivamente sulle misure effettuate.

Nel modello ARMAX (figura 3.5), la funzione di trasferimento $G(z)$ (da determinare)

lega ingresso noto a uscita del sistema, al quale si sovrappone del rumore bianco $\xi(t)$ per tenere conto di errori di misura e di modellazione. Nel modello ARMA (figura 3.6), non è presente l'informazione sull'ingresso noto: la funzione di trasferimento $W(z)$ (da determinare) lega un ingresso di qualsiasi tipo (ad esempio rumore bianco $\xi(t)$) all'uscita misurata del sistema. Il modello deve specificare il modo con cui i valori recenti della variabile sono determinati da quelli passati. Qualora il sistema sia forzato da una forzante monoarmonica conosciuta, l'unico metodo per il calcolo dei parametri modali è l'utilizzo del modello ARMAX. Tuttavia, se ci si pone nelle condizioni di non conoscere la natura della forzante, il modello ARMAX non può essere usato. Questo è il motivo per il quale verrà data enfasi al modello ARMA. Le relazioni che legano ingresso ed

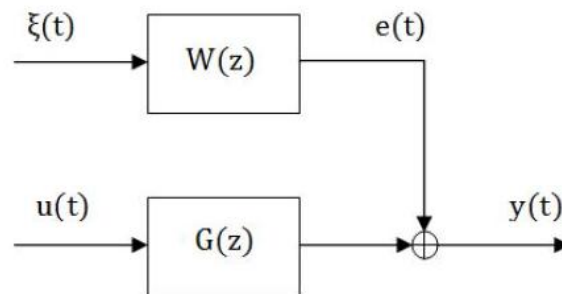


Figura 3.5: Modello di identificazione con ingresso $u(t)$, uscita $y(t)$ e rumore bianco (ARMAX)

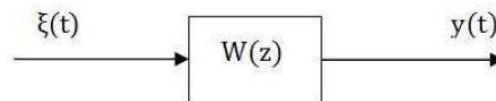


Figura 3.6: Modello ARMA: in ingresso c'è solamente rumore bianco

uscita nel modello ARMAX sono

$$y(t) = G(z)u(t) + W(z)\xi(t) \quad (3.40)$$

modello ARMA, invece, non si conosce l'ingresso noto $u(t)$ che non compare dunque nella 3.40. L'equazione alle differenze finite caratterizzante il modello ARMA è

$$y(t) + a_1y(t-1) + \dots + a_ny(t-n_a) = \xi(t) + c_1\xi(t-1) + \dots + c_n\xi(t-n_c) \quad (3.41)$$

n_a ed n_c identificano l'ordine dell'ARMA. Facendo uso della trasformata z , si riscrive la 3.41 come

$$A(z)y(t) = C(z)\xi(t) \quad (3.42)$$

dove

$$A(z) = 1 + a_1z^{-1} + \dots + a_{n_a}z^{-n_a} \quad (3.43)$$

$$B(z) = b_1z^{-1} + \dots + b_{n_b}z^{-n_b} \quad (3.44)$$

$$C(z) = 1 + c_1z^{-1} + \dots + c_{n_c}z^{-n_c} \quad (3.45)$$

Si può scrivere il modello ARMA in forma predittiva, definendo l'errore di predizione del modello ARMA come:

$$e(t, \theta) = y(t) - \hat{y}(t, \theta) \quad (3.46)$$

Il vettore θ ha la forma

$$\theta = [a_1, \dots, a_{n_a}, c_1, \dots, c_{n_c}]^T \quad (3.47)$$

Conoscendo l'errore tra $y(t)$ e il valore predetto $\hat{y}(t)$, si scrive l'equazione del modello ARMA in forma predittiva

$$\hat{y}(t, \theta) = \varphi^T(t, \theta)\theta \quad (3.48)$$

dove

$$\varphi = [-y(t-1) \dots y(t-n_a) e(t-1, \theta) \dots e(t-n_c, \theta)]^T \quad (3.49)$$

In questo modo l'ingresso del modello ARMA non è più rumore bianco, ma il valore dell'uscita del modello e l'errore del modello. Si può dimostrare che il predittore lineare così definito in (3.48) è ottimo qualora l'ingresso è rumore bianco [10]. I valori di n_a e n_c definiscono l'ordine del modello ARMA, come verrà discusso con maggior dettaglio nel seguito.

Si vuole adesso fare luce su come vengono calcolati i coefficienti θ del modello ARMA. La versione offline, dalla quale si parte per poter realizzare una versione online,

consiste nel metodo dei minimi quadrati ricorsivi che è applicabile con un predittore lineare. Si vuole minimizzare la funzione di errore quadratica

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N \mu^{N-i} [y(i) - \varphi^{\mathbf{T}}(i)\theta]^2 \quad (3.50)$$

Il coefficiente μ è il coefficiente di oblio, che pesa in modo diverso i dati passati rispetto ai dati recenti, in modo analogo a quanto descritto per il Subspace Tracking. Il minimo della funzione di errore quadratica si trova ponendo la sua derivata pari a 0, quindi

$$\frac{\partial J(\theta)}{\partial \theta} = -\frac{2}{N} \sum_{i=1}^N \mu^{N-i} \varphi(i)y(i) + \frac{2}{N} \sum_{i=1}^N \mu^{N-i} \varphi(i)\varphi^{\mathbf{T}}(i)\theta = 0 \quad (3.51)$$

da cui

$$\theta = \left[\sum_{i=1}^N \mu^{N-i} \varphi(i)\varphi^{\mathbf{T}}(i) \right]^{-1} \sum_{i=1}^N \mu^{N-i} \varphi(i)y(i) \quad (3.52)$$

Si definiscono

$$\mathbf{Q}(N) = \sum_{i=1}^N \mu^{N-i} \varphi(i)\varphi^{\mathbf{T}}(i) \quad (3.53)$$

e

$$\mathbf{f}(N) = \sum_{i=1}^N \mu^{N-i} \varphi(i)y(i) \quad (3.54)$$

e si può quindi riscrivere la 3.52 in forma compatta

$$\theta = \mathbf{Q}(N)^{-1} \mathbf{f}(N) \quad (3.55)$$

Siccome l'algoritmo deve stimare i parametri modali ogni volta che riceve un nuovo dato, le precedenti equazioni devono essere riscritte in modo ricorsivo e la sommatoria deve arrivare fino al generico istante di tempo t . Quindi

$$\mathbf{Q}(t) = \sum_{i=1}^t \mu^{t-i} \varphi(i)\varphi^{\mathbf{T}}(i) \quad (3.56)$$

$$\mathbf{f}(t) = \sum_{i=1}^t \mu^{t-i} \varphi(i)y(i) \quad (3.57)$$

$$\theta(t) = \mathbf{Q}(t)^{-1} \mathbf{f}(t) \quad (3.58)$$

Le equazioni 3.56 e 3.57 possono essere riscritte come

$$\mathbf{Q}(t) = \mu\mathbf{Q}(t-1) + \boldsymbol{\varphi}(t)\boldsymbol{\varphi}^T(t) \quad (3.59)$$

$$\mathbf{f}(t) = \mu\mathbf{f}(t-1) + \boldsymbol{\varphi}(t)y(t) \quad (3.60)$$

Effettuando alcuni passaggi algebrici partendo dalle equazioni (3.58), (3.59) e (3.60) (si rimanda a [4]) si perviene a

$$\mathbf{Q}(t) = \mathbf{Q}(t-1) + \mathbf{Q}(t-1)^{-1}\boldsymbol{\varphi}(t)[y(t) - \boldsymbol{\varphi}^T(t)\boldsymbol{\theta}(t-1)] \quad (3.61)$$

In questo modo, ad ogni passo, è necessario invertire la matrice \mathbf{Q} . È possibile aggirare questo ostacolo ricorrendo al lemma dell'inversione della matrice [26]. Date quattro matrici $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$, ipotizzando che la quantità $\mathbf{A} + \mathbf{BCD}$ abbia senso e sia invertibile, e che siano invertibili anche le matrici \mathbf{A}, \mathbf{C} , allora

$$[\mathbf{A} + \mathbf{BCD}]^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}[\mathbf{DA}^{-1}\mathbf{B} + \mathbf{C}^{-1}\mathbf{DA} - \mathbf{1}] \quad (3.62)$$

Quindi, ponendo

$$\begin{aligned} \mathbf{A} &= \mu\mathbf{Q}(t-1) \\ \mathbf{B} &= \mathbf{D}^T = \boldsymbol{\varphi}(t) \\ \mathbf{C} &= \mathbf{1} \\ \mathbf{P}(t) &= \mathbf{Q}^{-1}(t) \end{aligned} \quad (3.63)$$

ed utilizzando il lemma dell'inversione della matrice, si perviene a

$$\mathbf{P}(t) = \frac{1}{\mu} \left[\mathbf{P}(t-1) - \frac{\mathbf{P}(t-1)\boldsymbol{\varphi}(t)\boldsymbol{\varphi}^T(t)\mathbf{P}(t-1)}{\mu + \boldsymbol{\varphi}^T(t)\mathbf{P}(t-1)\boldsymbol{\varphi}(t)} \right] \quad (3.64)$$

Quindi, in definitiva, il calcolo di $\boldsymbol{\theta}$ diventa

$$\boldsymbol{\theta}(t) = \boldsymbol{\theta}(t-1) + \mathbb{I}(t)\boldsymbol{\varphi}(t)e(t) \quad (3.65)$$

Quando l'algoritmo inizia, si fornisce una $\boldsymbol{\theta}$ iniziale e una matrice \mathbf{P} iniziale. Viene calcolato un errore di predizione definito come

$$e(t) = y(t) - \boldsymbol{\varphi}^T(t)\boldsymbol{\theta}(t-1) \quad (3.66)$$

usando l'errore di predizione e la matrice \mathbf{Q} , si può procedere al calcolo del nuovo θ , che contiene le informazioni per ricavare i parametri del sistema meccanico. La scelta di θ iniziale \mathbf{P} iniziale è, in genere

$$\begin{aligned}\mathbf{P} &= \mathbf{I} \\ \theta(0) &= \mathbf{0}\end{aligned}\tag{3.67}$$

Il vettore θ ha un numero di elementi n_θ . Gli zeri di un'equazione algebrica di grado $n_\theta + 1$, con il coefficiente del grado massimo posto unitario, rappresentano i poli discreti del sistema. La relazione che li lega ai poli del corrispondente sistema continuo è

$$s_i = \frac{1}{dt_c} \ln z_i\tag{3.68}$$

Frequenze proprie f e smorzamenti adimensionali ξ sono ora calcolabili mediante

$$\begin{aligned}f &= \text{imag}(p_i)/(2\pi) \\ \xi &= \text{real}(p_i)/\text{imag}(p_i)\end{aligned}\tag{3.69}$$

dove p_i è l' i -esimo polo del sistema continuo, con i che va da 1 a n_a . Come conclusione alla trattazione teorica dell'algorithmo di identificazione RT, si vuole sottolineare che esso coinvolge alcuni parametri che devono essere opportunamente settati. Questi parametri sono:

- coefficiente di oblio di ST e ARMA;
- lunghezza finestra mobile del ST (m);
- numero di storie temporali in uscita dal ST ($k/2$);

Nel capitolo 4 verrà ottimizzato l'intero algorithmo per una sua applicazione sperimentale su una smart structure, parte di questa ottimizzazione prevede il corretto settaggio dei parametri sopra elencati.

Capitolo 4

Identificazione RT - Prove sperimentali

Si vogliono ora testare robustezza e prestazioni del sistema di identificazione modale sviluppato nel capitolo precedente, su una piastra in fibra di carbonio (smart structure). L'identificazione real-time può essere utilizzata per conoscere la forzante di disturbo (supposta ignota), o per identificare i parametri modali di un sistema. Qualora la forzante sia monoarmonica, ci si aspetta che l'algoritmo di identificazione fornisca in uscita la frequenza della forzante. Questo utilizzo verrà impiegato nel seguito per realizzare una logica di controllo adattativo con gain scheduling e controllo di un sottoinsieme del totale dei modi di vibrare modellati numericamente. In questo capitolo ci si concentrerà sull'identificazione real-time dei parametri modali del sistema meccanico. Si procederà all'identificazione dei parametri modali del sistema con differenti modalità di forzamento e modificando la posizione del sensore grazie al quale l'algoritmo di identificazione funziona.

4.1 Il Banco prova

Il banco prova è costituito da:

- Piastra in fibra di carbonio le cui caratteristiche sono mostrate nella tabella 4.1;

PARAMETRO	VALORE
Dimensioni	1.1x0.95 m

Spessore	$1.41 \cdot 10^{-3} m$
Strati	7 strati, disposti alternativamente a 10° e 90°
E_{11} [Pa]	$3e10$
E_{22} [Pa]	$3e10$
G_{12} [Pa]	$1.25e10$
$\nu_{1,2}$	0.4

Tabella 4.1: Caratteristiche smart structure

- telaio di sostegno che permette di modulare la tensione della piastra agendo su una vite;
- 4 patch piezoelettriche: una atta allo sviluppo di una forzante di disturbo, e le altre 3 predisposte alla realizzazione di un'eventuale azione di controllo (si veda la tabella 4.2);

PARAMETRO	VALORE
Input [V]	$\pm 200V$
deformazione massima	$560 \mu \epsilon$
Forza massima	300 N

Tabella 4.2: Caratteristiche patch piezoelettriche

- 3 estensimetri;

PARAMETRO	VALORE
Gauge factor	$2.04 \pm 1.0\%$
Resistenza	$120 \pm 0.35\% \Omega$

Tabella 4.3: Caratteristiche estensimetri

- centrale di condizionamento per estensimetri KWS;
- 3 accelerometri;

PARAMETRO	VALORE
Range operativo	78 g
Sensibility	100 mV/g

Tabella 4.4: Caratteristiche accelerometri

- centrale di condizionamento per accelerometri Nexus;
- dSpace 1006;
- filtro elettronico passa-basso a 500 Hz 4.1. L'input del filtro è l'uscita della dSpace, e l'output del filtro è indirizzato all'amplificatore;
- amplificatore T-400 che amplifica un segnale in ingresso a $\pm 10V$ in un segnale di uscita a $\pm 200V$ indirizzato agli attuatori piezoelettrici;

Facendo riferimento alle figure 4.2 4.3, si individuano:

- punto A: regolatore di tensione della piastra;
- punto B: attuatore piezoelettrico n4, tramite il quale viene applicata una forza di disturbo;
- punto C: estensimetro n1;
- punto D: estensimetro n2;
- punto E: estensimetro n3;
- punto F: estensimetro n4;
- punto G: estensimetro n5;
- punto H: estensimetro n6;

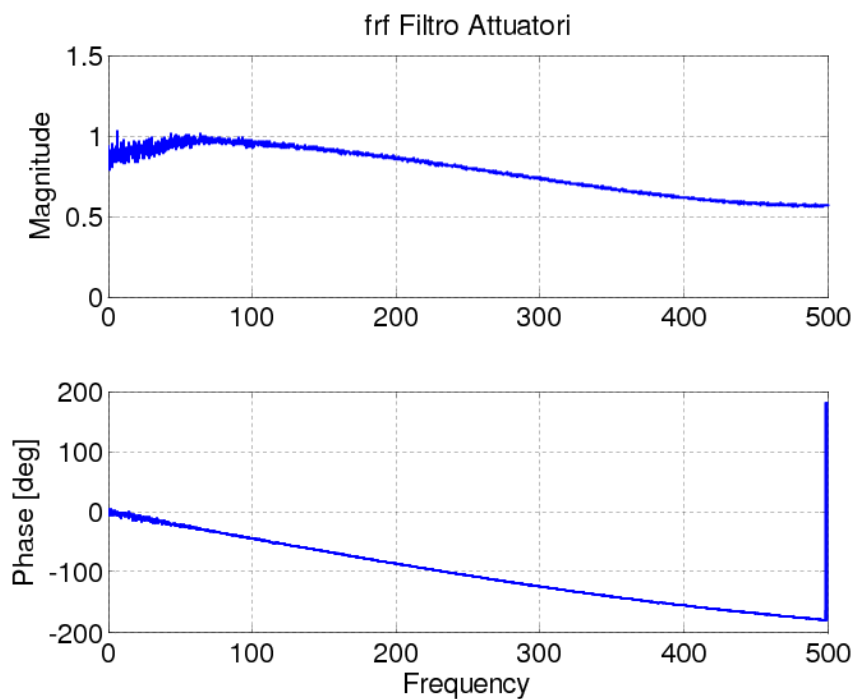


Figura 4.1: Funzione di trasferimento filtro, 0-500 Hz

- punto I: accelerometro n1 (contrassegnato dal numero 8 scritto sulla piastra);
- punto J: accelerometro n2 (contrassegnato dal numero 9 scritto sulla piastra);
- punto K: accelerometro n3 (contrassegnato dal numero 10 scritto sulla piastra);
- punto L: attuatore piezoelettrico n3 (per fornire la forza di controllo);
- punto M: attuatore piezoelettrico n2 (per fornire la forza di controllo);
- punto N: attuatore piezoelettrico n1 (per fornire la forza di controllo);

Per le prove di identificazione, verrà applicato un disturbo nel punto B.

4.2 Applicazione sperimentale di ST e ARMA

Definite le basi teoriche dell' algoritmo di identificazione, si vogliono chiarire meglio alcuni concetti relativi ad esso, ed apportare alcune migliorie che verranno utilizzate per

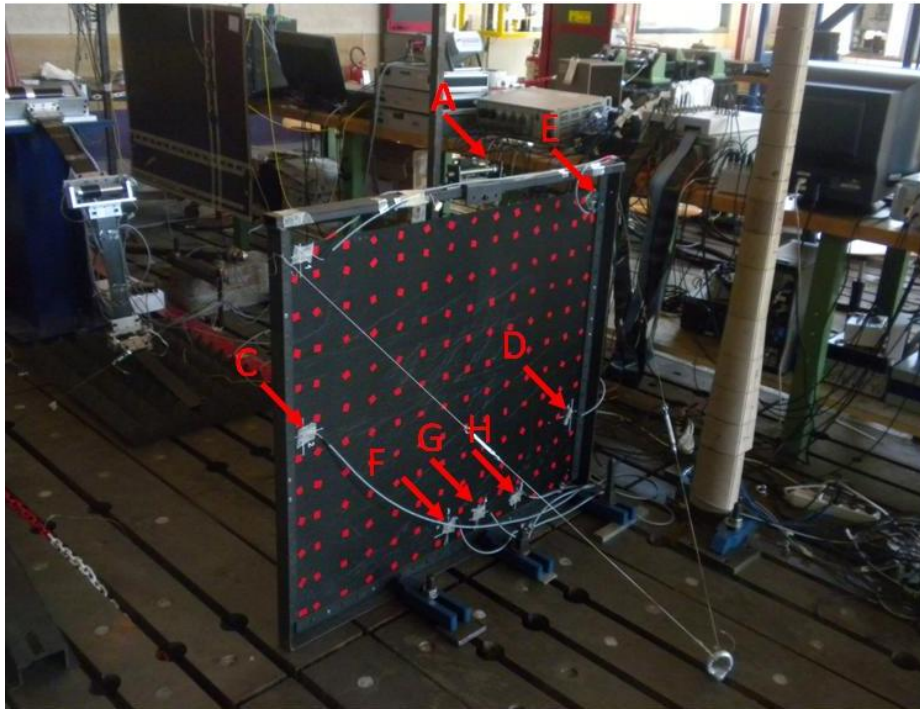


Figura 4.2: Piastra, lato A

ottenere risultati sperimentali soddisfacenti e che si renderanno indispensabili per poter realizzare una logica di controllo adattativo.

A tal fine, è stata acquisita, alla frequenza di campionamento di 250 Hz, una storia temporale proveniente da un accelerometro montato sulla piastra sopra descritta. La storia temporale è stata elaborata off-line con l'algoritmo di identificazione in precedenza sviluppato, nel seguito verranno presentati i risultati ottenuti e descritte nel dettaglio le migliori apportate. In figura 4.2 si può vedere il segnale campionato a 250 Hz, mentre la figura 4.5 è lo spettro dello stesso segnale, le cui frequenze più importanti (identificate con una semplice analisi di Fourier off-line) sono visibili in tabella 4.5.

<i>Frequenza</i>	<i>Hz</i>
1	25
2	26
3	66
4	83

5	77
6	35

Tabella 4.5: Frequenze presenti nel segnale random

La prima parte dell'algoritmo di identificazione è la scomposizione del segnale utilizzando il Subspace Tracking con lunghezza della finestra mobile $m = 400$, e con coefficiente di oblio $\lambda_{ST} = 0.99999995$. Il segnale di figura 4.6 è piatto per i primi 400 punti, siccome la prima finestra mobile di dati del ST non si è ancora riempita, e si sceglie di mandare in uscita il valore 0. La prima storia temporale proveniente dal Subspace Tracking viene elaborata dall'algoritmo ARMA adattato al funzionamento real-time. Aumentando l'ordine dell'ARMA sull'uscita (quindi passando da un ARMA(2,2) ad un ARMA(3,2)), si ottengono dei benefici nonostante la storia che esso elabora ha idealmente componente monoarmonica, e sarebbe quindi sufficiente un ARMA(2,2). Tuttavia, un segnale reale in uscita dal Subspace Tracking, non sempre è monoarmonico per tutta la lunghezza della finestra m . Questo comporta l'introduzione di brusche irregolarità. Incrementando l'ordine dell'ARMA, vengono introdotti poli aggiuntivi, che assorbono le irregolarità del segnale. Usando un ARMA(3,2), si ottiene solo un polo complesso coniugato che è quello da considerare. Incrementare ulteriormente l'ordine dell'ARMA renderebbe complicato computazionalmente l'automatizzazione della scelta del polo corretto.

Elaborando la prima storia temporale del Subspace Tracking con un ARMA(3,2) (dove 3 e 2 sono rispettivamente n_a ed n_c) si ottiene una sola coppia di poli complessi e coniugati. Scegliendo il polo con parte immaginaria >0 , si ottiene quanto riportato in figura 4.7, che in effetti trova conferma nel confronto con lo spettro delle frequenze del segnale analizzato.

Elaborando la seconda uscita del Subspace Tracking (visibile in figura 4.8) con un ARMA(3,2), si ottiene il risultato visibile in figura 4.9: si può notare un tratto obliquo dove dall'identificazione della frequenza a 80 Hz si passa ad identificare quella a 25 Hz. Il fenomeno accade quando cambia il modulo delle armoniche dei modi identificati, e di conseguenza cambia l'ordine con cui queste compaiono nell'uscita del ST. Nel ST, la

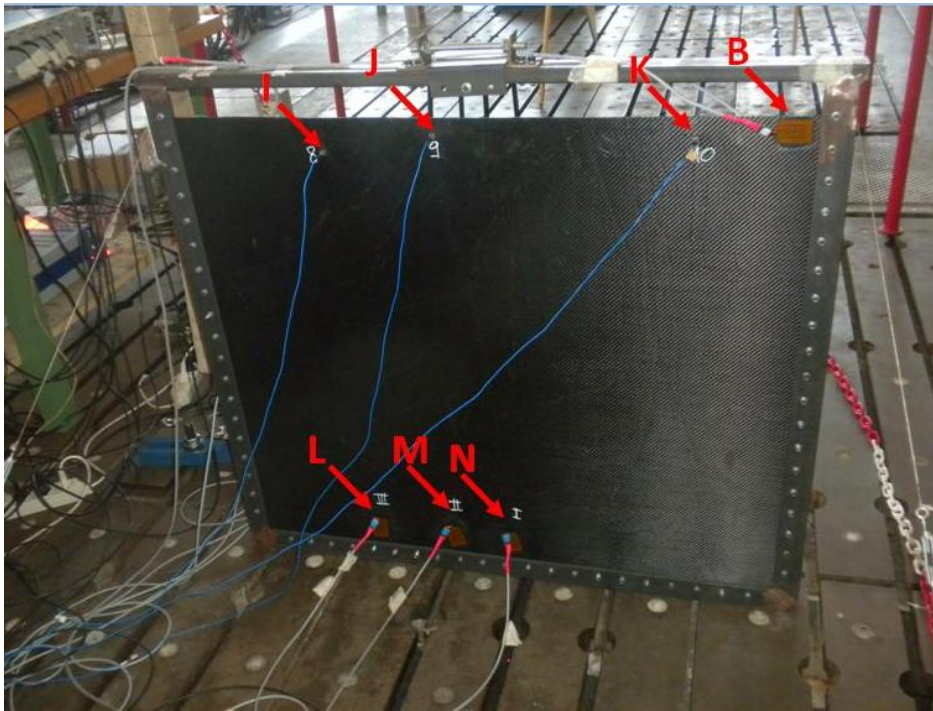


Figura 4.3: Piastra, lato B

prima uscita (che è una storia temporale), è l'armonica a valore assoluto maggiore, le rimanenti uscite contengono armoniche a modulo sempre più basso, fino all'ultima uscita del ST che contiene l'armonica con il minor modulo (fra le armoniche scomposte dal ST). Siccome le armoniche possono cambiare il valore del proprio modulo, una singola storia temporale in uscita dal ST può contenere al suo interno monoarmoniche differenti (alcuni tratti un'armonica, e altri tratti un'armonica differente). L'uscita del Subspace Tracking che può così contenere più di un'armonica, viene elaborata dall'ARMA, che si sintonizza gradualmente da una frequenza all'altra. Durante questo transitorio l'ARMA fornisce delle frequenze intermedie tra le 2 frequenze presenti nella singola uscita del Subspace tracking, quest'ultima però non le contiene, si può dunque affermare che durante il transitorio, l'ARMA identifica frequenze inesistenti. Elaborando con l'ARMA la terza storia temporale in uscita dal Subspace Tracking (figura 4.10) si ottiene quanto riportato in figura 4.11. Si nota che c'è un continuo transitorio tra la frequenza a 80 Hz e quella a 25 Hz (figura 4.12). Questo rende inutilizzabile il processo di identifica-

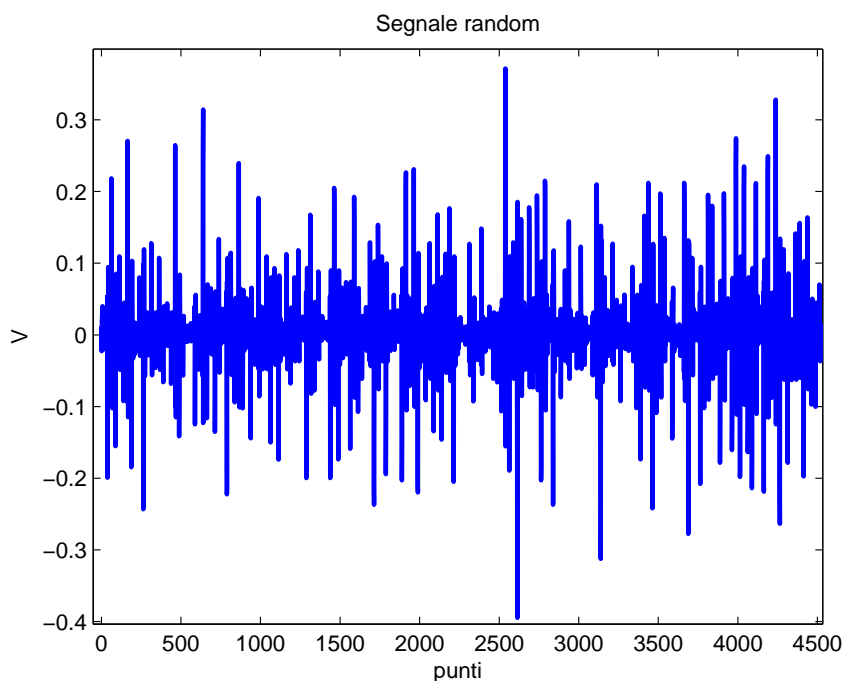


Figura 4.4: Segnale random

zione. Questo problema potrebbe essere ridotto modificando il coefficiente di oblio del Subspace Tracking, e settandolo ad un valore più basso. In questo modo i dati passati avrebbero un minor contributo, rendendo l'algoritmo più pronto durante i transitori, ma anche più sensibile ai disturbi (rumore). Viceversa, un valore elevato del coefficiente di oblio, rende il ST meno sensibile al rumore ma anche meno pronto durante i transitori da una frequenza ad un'altra. Per ovviare al problema della scelta del coefficiente di oblio, si è scelto di renderlo adattativo, passando da un coefficiente all'altro a seconda delle condizioni in cui si trova il ST (e quindi il sistema). Si è dunque implementato un algoritmo che rende possibile il passaggio da un valore all'altro del coefficiente di oblio (*oblio adattativo*). I criteri su cui si basa tale algoritmo sono:

- calcolo della varianza su una storia temporale di m punti;
- se la varianza calcolata sull' i -esima finestra m è maggiore di oltre il 40 % della varianza calcolata sulla finestra $i-1$, allora si modifica il coefficiente di oblio impostandolo al valore di transitorio;

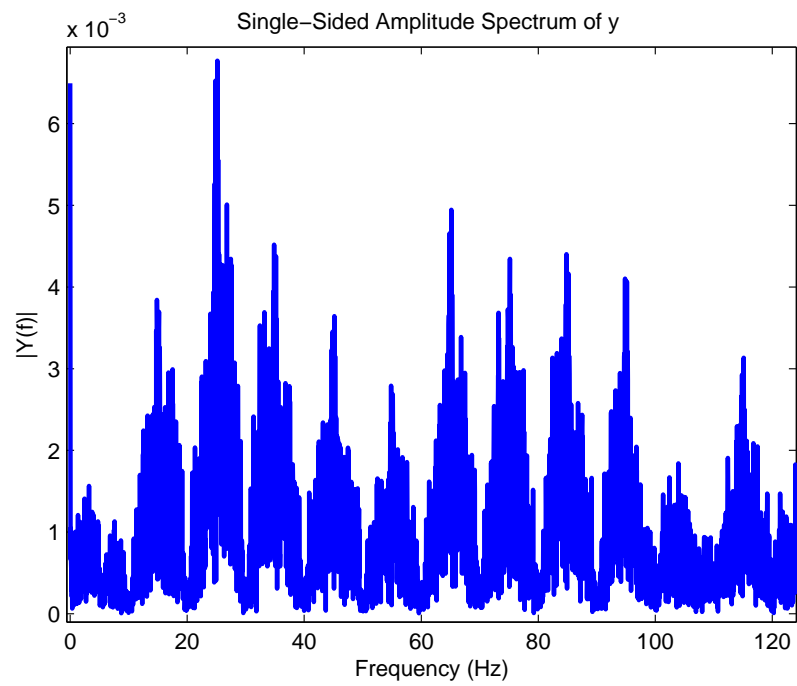


Figura 4.5: Segnale random, spettro di frequenze

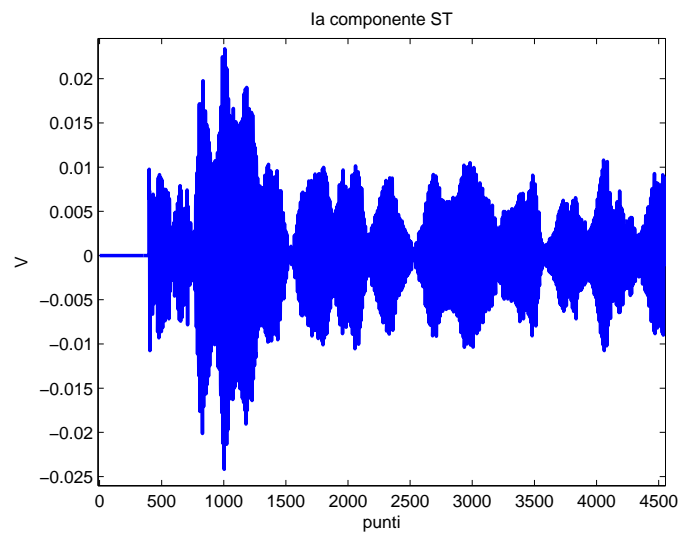


Figura 4.6: Prima storia temporale in uscita dal ST

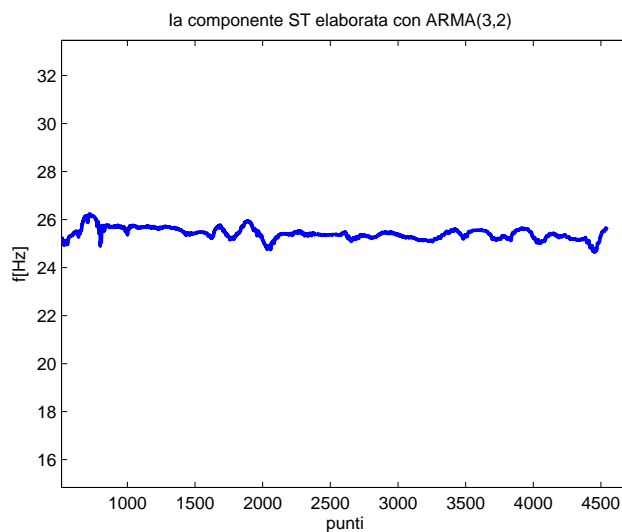


Figura 4.7: Prima storia temporale del ST elaborata dall'ARMA

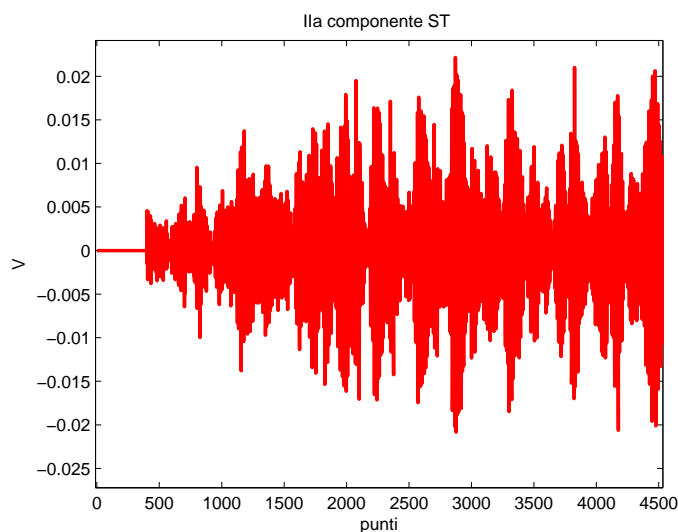


Figura 4.8: Seconda storia temporale in uscita dal ST

- il valore di regime del coefficiente di oblio verrà ripristinato solamente quando il rapporto tra le varianze tornerà ad un valore minore del 40 %;

Rielaborando il segnale acquisito con l'opzione dell'oblio adattativo, si hanno miglioramenti sull'uscita dell'ARMA(3,2) come visibile in figura 4.13.

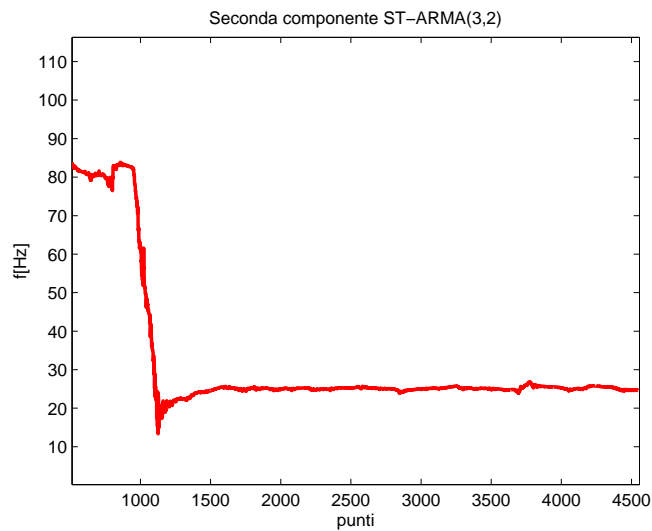


Figura 4.9: Seconda storia temporale del ST elaborata dall'ARMA

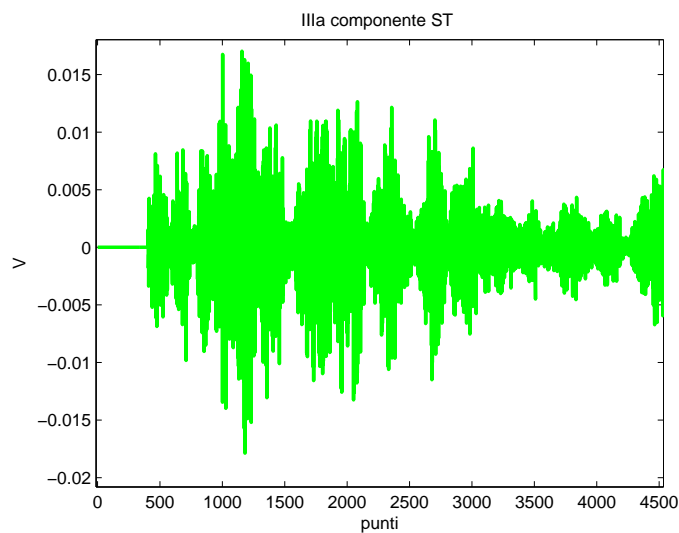


Figura 4.10: Terza storia temporale del ST

Anche alle frequenze con meno contributo in vibrazione, quindi più difficilmente identificabili perchè con un rapporto segnale/rumore inferiore, l'*oblio adattativo* apporta dei vantaggi, come visibile nel confronto tra le figure 4.14 e 4.15.

Un'ultima considerazione a proposito del coefficiente m del Subspace Tracking. Si

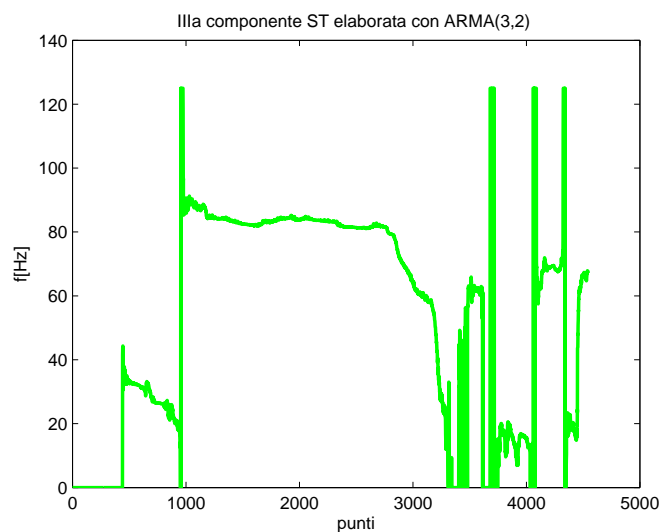


Figura 4.11: Terza storia temporale del ST elaborata dall'ARMA

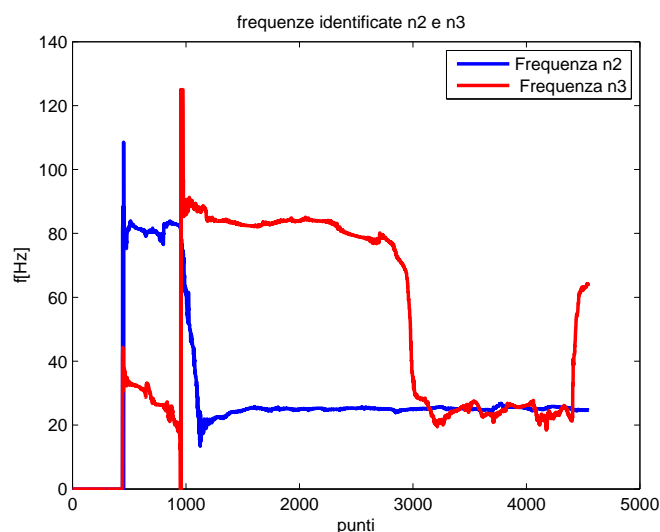


Figura 4.12: Scambio di frequenze identificate

è osservato in maniera empirica che l'algoritmo di identificazione funziona in modo accettabile a patto che sia verificata la condizione

$$m \frac{f}{f_s} > 3.2 \quad (4.1)$$

dove f_s è la frequenza di campionamento, e f è la frequenza in Hz della più bassa

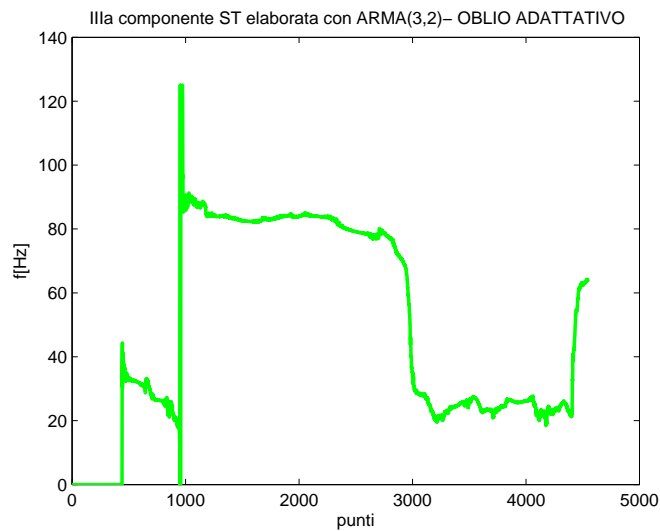


Figura 4.13: Terza storia temporale del ST elaborata dall'ARMA - oblio adattativo

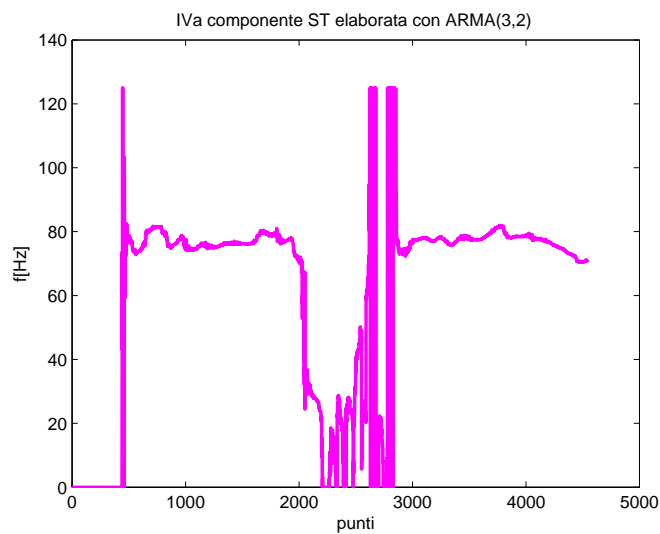


Figura 4.14: Quarta storia temporale del ST elaborata dall'ARMA - senza oblio adattativo

frequenza da identificare. Per stabilire il valore di m è necessario prima assegnare f_s . Nel caso in esame, volendo identificare frequenze maggiori di 100 Hz, esso deve essere maggiore di 200 Hz, per evitare aliasing. Si può ad esempio fissare $f_s = 250\text{Hz}$. Si

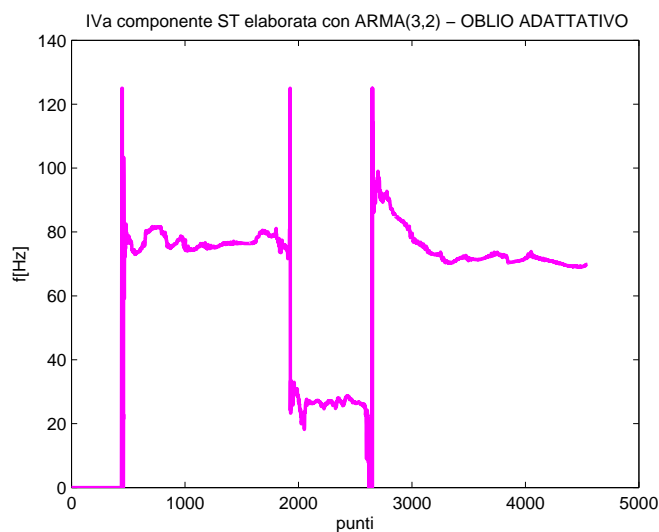


Figura 4.15: Quarta storia temporale del ST elaborata dall'ARMA - oblio adattativo

sceglie dunque la frequenza più bassa che si vuole identificare accuratamente, $f = 5Hz$. La relazione 4.1 porta a scegliere un valore $m \geq 160$. Viene scelto il valore minimo di 160, per limitare il costo computazionale dell'algoritmo.

f_{min}	f_s	m
5	300	192 punti
7	220	100 punti
7	250	114 punti
1	1000	3200 punti
100	1000	32 punti

Tabella 4.6: Scelta di m in funzione della frequenza minima e di f_s

In conclusione, si è scelto per il seguito di utilizzare un algoritmo di identificazione coi seguenti parametri:

PARAMETRO	SIGNIFICATO	VALORE
m	Lunghezza della finestra mobile di dati del Subspace tracking	160 punti
λ_{ST}	Coefficiente di oblio del Subspace Tracking	0.99999995
$\lambda_{ST_{trans}}$	Coefficiente di oblio del Subspace Tracking durante il cambio frequenza	0.9995
λ	Coefficiente di oblio dell'ARMA	0.99
Ordine ARMA		3,2

Tabella 4.7: Riassunto dei parametri caratteristici dell'algoritmo di identificazione

4.3 La gestione dei parametri identificati

Alla luce dei risultati esposti nel paragrafo 4.2, è possibile osservare delle condizioni che rendono i parametri modali identificati sperimentalmente (nel paragrafo 4.2 sono state valutate solamente le frequenze proprie) non sempre attendibili. Uno dei motivi principali è la non stazionarietà del modulo delle frequenze proprie del sistema, fenomeno amplificato dal fatto che in una piastra è molto elevata la densità di frequenze proprie. In questi casi i parametri identificati con ST + ARMA non possono essere utilizzati così come sono per sintetizzare un controllo adattativo. Si rende necessaria un'elaborazione dei parametri modali in uscita dall'ARMA affinché si possano stimare i parametri corretti del sistema. Si usa quindi un approccio statistico grazie al quale si scelgono le coppie smorzamento-frequenza identificate più volte all'interno di un certo periodo di tempo. Il metodo prevede la creazione di un array bidimensionale (ogni colonna è una frequenza, ogni riga uno smorzamento adimensionale), le cui celle contengono il numero di volte col quale una certa frequenza e un certo smorzamento si sono presentati. Dopo un certo periodo di tempo, vengono sommate le colonne e ordinate per ordine decrescente: in questo modo si scelgono le frequenze più sollecitate. Dopo que-

sta operazione, si effettua la scelta dello smorzamento col valore di comparsa più alto per ciascuna delle frequenze scelte. In questo modo non viene aggiornata la matrice di stato coi valori puntuali, ma l'aggiornamento viene effettuato con un certo ritardo. La scelta di questo ritardo deve essere tale da consentire ugualmente un aggiornamento dei parametri modali compatibile con la dinamica del sistema. Osservando la figura 4.16,

FREQUENZE →	10 Hz	10.5 Hz	11.0 Hz	11.5 Hz	12.0 Hz	12.5 Hz	13 Hz
SMORZAMENTI ↓	0	0	0	0	0	1	0
0,0075	2	8	1	0	0	0	1
0,0079	4	16	6	0	0	0	0
0,0083	15	15	22	4	1	0	0
0,0087	3	6	2	7	2	0	0
0,0091	0	1	4	12	4	1	2
0,0095	0	0	2	4	2	0	0
...	0	0	0	0	0	1	0

Figura 4.16: Esempio di contatore bidimensionale per le coppie frequenza-smorzamento identificati dall'ARMA(3,2)

vengono selezionati i valori il cui contatore ha lo sfondo bianco. Non sempre però ci sono dei massimi assoluti come in questo caso. Osservando la figura 4.17, i valori con

FREQUENZE →	10 Hz	10.5 Hz	11.0 Hz	11.5 Hz	12.0 Hz	12.5 Hz	13 Hz
SMORZAMENTI ↓	0	0	0	0	0	1	0
0,0075	2	8	1	0	0	0	1
0,0079	4	4	6	0	0	0	0
0,0083	15	22	13	4	1	0	0
0,0087	3	6	2	7	2	0	0
0,0091	0	1	4	12	4	1	2
0,0095	0	0	2	4	2	0	0
...	0	0	0	0	0	1	0

Figura 4.17: Contatore bidimensionale: frequenze vicine

sfondo bianco adiacenti l'un l'altro (aventi valori 15,22,13 e 12), identificano in realtà la stessa frequenza. Tuttavia l'algoritmo le vede come se fossero 4 frequenze distinte,

effettuando una stima errata dei parametri modali. Una soluzione potrebbe essere quella di allargare la griglia dei valori di frequenza, in modo tale da occupare con 4 frequenze un unico spazio della discretizzazione (griglia). In questo modo si commetterebbe un errore di arrotondamento molto marcato. Inoltre è altrettanto difficoltoso stabilire i criteri secondo i quali n frequenze adiacenti nella griglia corrispondono a una sola frequenza. Un'altra soluzione può essere quella di utilizzare una griglia larga, e mediare i valori che cadono all'interno di ciascuno dei suoi campi. In questo modo vengono parzialmente persi i benefici dell'approccio statistico, grazie al quale vengono considerate i parametri modali identificati alla stregua di una qualsiasi misura effettuata su un sistema (quindi con distribuzione gaussiana con dispersione attorno al valore medio). Inoltre resta sempre il problema di stabilire qual'è il valore oltre il quale 2 frequenze adiacenti nella griglia possano essere considerate 2 frequenze distinte.

La soluzione scelta prevede dunque la presenza di un contatore bidimensionale (buffer bidimensionale) a griglia larga, che permetta di considerare come indipendenti 2 frequenze fra loro adiacenti nella griglia. Ciascuna cella della griglia ha all'interno un'altra partizione: una griglia piccola. Si procede dunque nel seguente modo:

- Vengono selezionate le n griglie larghe col numero maggiore, (si ricorda che la griglia contiene un numero, è una cella del buffer bidimensionale precedentemente discusso) che corrispondono a n frequenze distinte, per l'ipotesi fatta in precedenza;
- viene scelta la cella della griglia piccola col numero maggiore, a cui corrisponde un valore di frequenza e smorzamento adimensionale ipotizzati corretti, ad esempio in figura 4.18 verrebbero scelte le frequenze a 11.0 e a 12.0 Hz, con i rispettivi smorzamenti adimensionali (in questo caso entrambi a 0.0083). Si nota come in questo modo, si pone maggiore enfasi alla stima della frequenza a discapito dello smorzamento, siccome la divisione in griglie grosse è effettuata solo sul campo delle frequenze e non su quello degli smorzamenti. D'altronde, come verrà successivamente discusso, l'errore sulla stima dello smorzamento dipende soprattutto dall'ARMA.
- Qualora si verificano condizioni particolari (figura 4.19), cioè andrebbero scelte

2 frequenze adiacenti ma in 2 macrocelle differenti, viene scelto il loro valore medio. Essendo fissata a priori la dimensione del vettore dei parametri identificati, in questo caso l'uscita dell'algoritmo avrà una coppia di cifre in meno, siccome 2 frequenze sono collassate in un'unica frequenza media. In tal caso l'algoritmo restituisce un vettore di frequenze e smorzamenti il cui elemento mancante viene rimpiazzato con un simbolico -1, in modo tale che non si terrà in considerazione di quel valore nei successivi calcoli. Viene inoltre modificato il valore della frequenza ai bordi della griglia grossa. Il valore di destra viene diventa il valore medio delle 2 frequenze adiacenti identificate, e quello di sinistra è uguale al nuovo valore di destra, a cui viene sottratto l'intervallo di discretizzazione (che nel caso di figura 4.19, dove è illustrato questo procedimento, è pari a 0.5). In questo modo si cerca di convergere a un valore di frequenza stabile, mantenendo possibile la lettura di variazioni nel sistema. Si è visto empiricamente che si hanno buoni risultati dividendo il range di frequenze in 100 colonne piccole e in 20 colonne grosse (una colonna grossa contiene 5 colonne piccole). Per quanto riguarda la discretizzazione del campo degli smorzamenti, si usa un totale di 100 righe.

N. FREQUENZA	FREQUENZA 1			FREQUENZA 2			...
FREQUENZE →	10 Hz	10.5 Hz	11.0 Hz	11.5 Hz	12.0 Hz	12.5 Hz	13 Hz
SMORZAMENTI ↓	0	0	0	0	0	1	0
0,0075	2	8	1	0	0	0	1
0,0079	4	16	6	0	0	0	0
0,0083	15	15	22	4	19	0	0
0,0087	3	6	2	7	7	0	0
0,0091	0	1	4	4	4	1	2
0,0095	0	0	2	4	2	0	0
...	0	0	0	0	0	1	0
Somma per MACROCELLA	105			53			...

Figura 4.18: Contatore bidimensionale con griglia grossa e piccola

N.frequenza →	frequenza 1			frequenza 2			...
FREQUENZE →	10 Hz	10.5 Hz	11 Hz	11.5 Hz	12 Hz	12.5 Hz	...
SMORZAMENTI ↓	0	0	0	0	0	1	0
0,0075	2	8	1	0	0	0	1
0,0079	4	16	6	0	0	0	0
0,0083	15	22	23	24	1	0	0
0,0087	3	6	2	7	2	0	0
0,0091	0	1	4	12	4	1	2
0,0095	0	0	2	4	2	0	0
...	0	0	0	0	0	1	0
somma per frequenza	115			59			
N.frequenza →	frequenza 1			frequenza 2			...
FREQUENZE →	10 Hz	10.5 Hz	10.75 Hz	11.25 Hz	12.0 Hz	12.5 Hz	13 Hz
SMORZAMENTI ↓	0	0	0	0	0	1	0
0,0075	2	8	1	0	0	0	1
0,0079	4	16	6	0	0	0	0
0,0083	15	22	23	24	1	0	0
0,0087	3	6	2	7	2	0	0
0,0091	0	1	4	12	4	1	2
0,0095	0	0	2	4	2	0	0
...	0	0	0	0	0	1	0
somma per frequenza	115			59			

Figura 4.19: Contatore bidimensionale: caso particolare

4.4 Parametri modali identificati RT

L'algoritmo di identificazione ottimizzato nel paragrafo 4.2, viene ora testato per l'identificazione dei parametri modali in diverse condizioni di forzamento e con diverse posizioni del sensore di acquisizione. Partendo inizialmente dalla struttura sollecitata con una forza di disturbo random con spettro tra 0 e 100 Hz, verrà successivamente ridotto il campo spettrale, testando l'algoritmo con un disturbo random avente banda spettrale compresa tra 0 e 50 Hz e con del rumore rosa.

4.4.1 Disturbo random 0-100 Hz

Per poter valutare la bontà dei parametri stimati real-time, è necessario disporre dei valori di frequenze proprie e smorzamenti ottenuti con un'accurata stima off-line (metodo di Prony). Tali parametri sono riassunti in tabella 4.8.

NUMERO MODO	FREQUENZA	SMORZAMENTO ADIMENSIONALE
1	8.1509 Hz	0.0247
2	14.8380 Hz	0.0081
3	19.8387 Hz	0.0154
4	25.1974 Hz	0.0175
5	28.3753 Hz	0.0070
6	35.0306 Hz	0.0062
7	39.2706 Hz	0.0074
8	44.4320 Hz	0.0085
9	52.8567 Hz	0.0040
10	58.0124 Hz	0.0022
11	62.1973 Hz	0.0045
12	66.4040 Hz	0.0036
13	74.7389 Hz	0.0041
14	80.0650 Hz	0.0043
15	80.3535 Hz	0.0037

16	84.7947 Hz	0.0041
17	88.1897 Hz	0.0060

Tabella 4.8: Parametri modali della piastra identificati offline

NUMERO MODO	FREQUENZA	SMORZAMENTO ADIMENSIONALE
1	19.78 Hz	0.1240
2	25.44 Hz	0.0089
3	62.18 Hz	0.0473
4	65.69 Hz	0.0116
5	67.36 Hz	0.0303
6	74.07 Hz	0.0239
7	81.96 Hz	0.0265
8	84.76 Hz	0.0179
9	84.96 Hz	0.0078
10	85.05 Hz	0.0028

Tabella 4.9: Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: random 0-100 Hz. Estensimetro n1.

Facendo riferimento alla tabella 4.9, si osserva precisione dell'identificazione delle frequenze proprie. La precisione sull'identificazione degli smorzamenti, al contrario, è nettamente inferiore, essendo un parametro di più difficile identificazione. L'imprecisione nella stima degli smorzamenti è a monte dell'approccio statistico con il quale questo viene scelto, ed è legato quindi all'algoritmo ARMA.

NUMERO MODO	FREQUENZA	SMORZAMENTO ADIMENSIONALE
1	58.14 Hz	0.0064

2	64.21 Hz	0.0615
3	67.33 Hz	0.0063
4	70.20 Hz	0.0288
5	72.01 Hz	0.0136
6	80.06 Hz	0.0018
7	80.52 Hz	0.0007
8	81.90 Hz	0.0230
9	84.72 Hz	0.0047
10	85.77 Hz	0.0041

Tabella 4.10: Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: random 0-100 Hz. Accelerometro n1.

Al variare del sensore utilizzato per l'identificazione ci sono delle differenze nelle frequenze identificate, come era lecito aspettarsi dalla logica che utilizza l'algoritmo di identificazione, che si basa sui dati provenienti da un solo sensore di misura: spostando questo sensore, esso può trovarsi in una posizione dove la misura è influenzata maggiormente da alcuni modi anziché da altri. Per questo motivo sarebbe indispensabile un'analisi offline della struttura, per scegliere i modi che si vogliono identificare ed eventualmente controllare, e di conseguenza selezionare il sensore che meglio si presta all'identificazione di questi modi, il tutto nell'ipotesi che la forma dei modi di vibrare non subisca variazioni.

4.4.2 Disturbo random 0-50 Hz e rumore rosa

Si forza ora la piastra con un disturbo random tra 0 e 50 Hz, e successivamente con un rumore rosa la cui funzione di trasferimento è visibile in figura 4.20. In termini di potenza, il rumore rosa decade di 3 dB per ottava.

NUMERO MODO	FREQUENZA	SMORZAMENTO ADIMENSIONALE
----------------	-----------	------------------------------

1	15.12 Hz	0.0011
2	19.27 Hz	0.0210
3	25.80 Hz	0.0454
4	38.80 Hz	0.0107
5	41.60 Hz	0.0199
6	46.13 Hz	0.0006
7	48.41 Hz	0.0008
8	52.48 Hz	0.0890

Tabella 4.11: Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: random 0-50 Hz. Estensimetro n1.

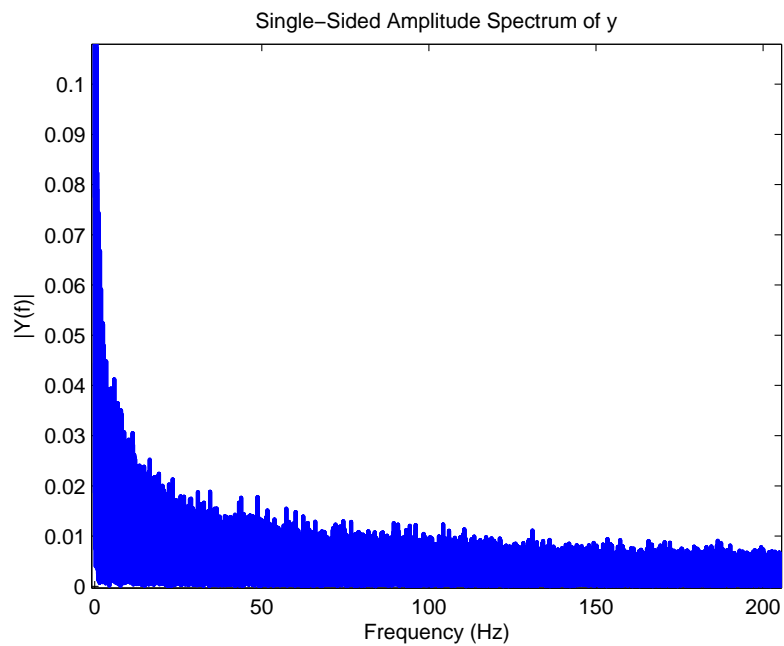


Figura 4.20: Rumore Rosa, fft a valle del filtro elettronico

NUMERO MODO	FREQUENZA	SMORZAMENTO ADIMENSIONALE
1	8.05 Hz	-0.0001
2	10.00 Hz	0.0001
3	42.64 Hz	-0.0009
4	48.84 Hz	0.0211
5	51.17 Hz	0.0297
6	52.76 Hz	0.0108
7	71.26 Hz	-0.0005
8	75.23 Hz	0.0675

Tabella 4.12: Parametri modali identificati dopo 40 secondi con identificazione real-time. Disturbo: rumore rosa. Estensimetro n1.

Forzando il sistema con un disturbo a banda spettrale più ristretta, si ottengono i risultati riportati in tabella 4.11 e 4.12. Nel caso di disturbo rosa vengono commessi alcuni errori di identificazione: nonostante alcune frequenze vengano ugualmente identificate in modo corretto, su altri modi l'algoritmo commette degli errori, identificando frequenze non presenti nell'elaborazione off-line della stessa storia temporale. Talvolta vengono identificati valori dello smorzamento adimensionale molto prossimi allo zero, che non corrispondono a nessuno smorzamento adimensionale del sistema, ma che sono legati all'identificazione di una forzante.

Alla luce di questi risultati, l'algoritmo di identificazione ideato può essere utilizzato per stimare le frequenze proprie nel caso di disturbo random ad ampia banda spettrale. La stima dello smorzamento adimensionale non è precisa in nessun caso. È però possibile usare il valore di smorzamento adimensionale identificato per stabilire lo stato in cui il sistema si trova. Valori prossimi allo zero indicano che la relativa frequenza identificata è relativa ad una forzante, si può quindi adattare il controllore affinché sia ottimizzato per l'abbattimento delle vibrazioni in quella particolare condizione. Inoltre valori negativi indicano che il sistema sta aumentando l'ampiezza della vibrazione,

e che quindi - ad esempio - il controllo sta peggiorando la precedente situazione. Un problema analogo si instaura con l'introduzione di uno smorzamento derivante dal controllo: l'algoritmo identifica infatti valori elevati di smorzamento adimensionale. Nel caso di un sistema controllato, diviene ancora più difficile la stima degli smorzamenti adimensionali. Sulla base di queste considerazioni è necessario scegliere le condizioni nelle quali effettuare un'aggiornamento della matrice di stato, valutando lo smorzamento adimensionale identificato: l'aggiornamento si effettuerà qualora esso dovesse cadere all'interno di una banda di confidenza.

Per delimitare il limite inferiore e superiore della banda di confidenza, si usano i parametri modali identificati offline per ottenere mediante interpolazione la curva di Rayleigh $\xi = \frac{\alpha}{\omega} + \beta \omega$, che rappresenta l'andamento teorico dello smorzamento in funzione della frequenza propria, nell'ipotesi che la matrice degli smorzamenti sia diagonale in seguito al passaggio in coordinate modali (sistema disaccoppiato). Si delimitano le curve limite della banda di confidenza in modo tale che al centro vi siano i punti della curva di Rayleigh interpolata (figura 4.21). A titolo di esempio, si riporta in figura 4.22 i valori della frequenza che l'algoritmo di identificazione fornisce: si può notare il graduale avvicinamento al valore identificato off-line (a parità di storia temporale elaborata).

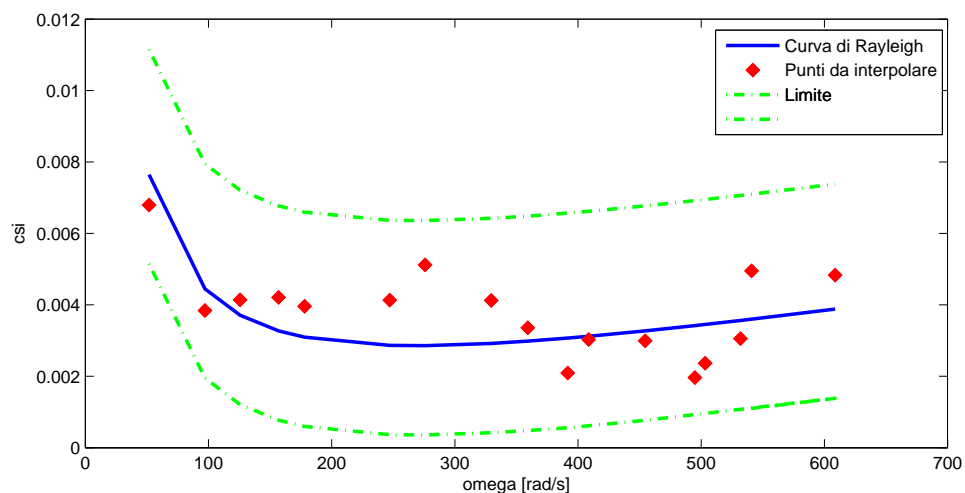


Figura 4.21: Curva di Rayleigh interpolata

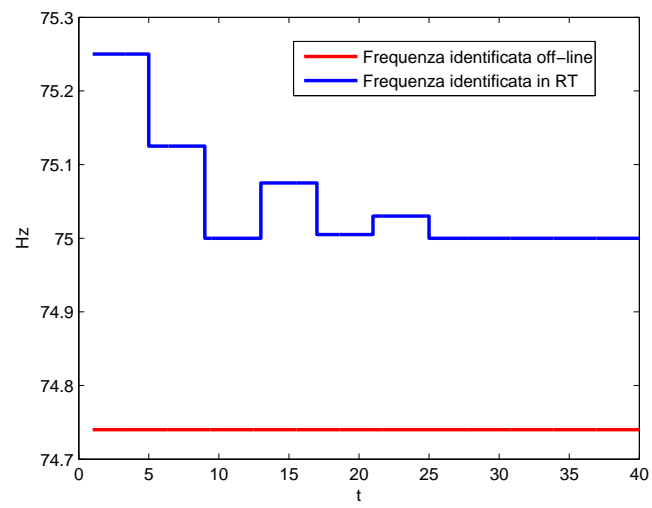


Figura 4.22: Valori identificati real-time per una singola frequenza

Capitolo 5

Logiche di controllo adattativo

In questo capitolo vengono proposte delle logiche di controllo modale adattativo, che verranno sperimentate numericamente e sperimentalmente sulla piastra in fibra di carbonio descritta nel capitolo 3.

Del modello numerico si conoscono le matrici del sistema in forma di stato:

$$\begin{cases} \dot{\mathbf{q}} = \mathbf{A}\mathbf{q} + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{q} \end{cases} \quad (5.1)$$

dove:

- \mathbf{q} è il vettore di stato in coordinate modali;
- \mathbf{u} è il vettore degli ingressi;
- \mathbf{y} è il vettore delle misure;
- $\mathbf{A} \in \mathfrak{R}^{34 \times 34}$ matrice di stato;
- $\mathbf{B} \in \mathfrak{R}^{34 \times 4}$: le prime 3 colonne sono relative agli attuatori del controllo e l'ultima colonna è relativa all'attuatore che genera il disturbo;
- $\mathbf{C} \in \mathfrak{R}^{6 \times 34}$: le prime 3 righe sono relative a misure estensimetriche, e le restanti righe sono relative alle misure degli accelerometri;

Frequenze proprie e smorzamenti del modello numerico sono visibili in tabella 5.1.

NUMERO MODO	FREQUENZA	SMORZAMENTO ADIMENSIONALE
1	10.4792 Hz	0.0025
2	13.9576 Hz	0.0035
3	20.0752 Hz	0.0016
4	21.7524 Hz	0.0058
5	24.2867 Hz	0.0019
6	34.1821 Hz	0.0025
7	37.3004 Hz	0.0017
8	42.5379 Hz	0.0016
9	48.0400 Hz	0.0013
10	55.9315 Hz	0.0012
11	61.3946 Hz	0.0037
12	68.1291 Hz	0.0050
13	69.5917 Hz	0.0022
14	71.9450 Hz	0.0022
15	82.0575 Hz	0.0012
16	86.1588 Hz	0.0043
17	88.3500 Hz	0.0049

Tabella 5.1: Parametri modali della piastra identificati offline

Le logiche di controllo fanno uso del controllore LQR , e si basano sull'algoritmo di identificazione descritto al capitolo 3. Effettuando un controllo sulla velocità modale \dot{q} , sarà necessario introdurre l'osservatore di stato, che fornisce la stima delle q a partire dalle misure y provenienti dagli estensimetri e dagli accelerometri (il cui ingresso viene integrato per avere una misura di velocità). Se non specificato diversamente nella descrizione delle prove numeriche e sperimentali all'interno della presente tesi, la variabile \dot{q} rappresenta sempre una velocità modale.

Verranno proposti 2 differenti algoritmi: il primo algoritmo mira ad abbassare il tempo di calcolo e la forza di controllo agli attuatori, controllando 3 modi alla volta (i più

sollecitati dalla forzante di disturbo), su un totale di 17 modi modellati numericamente. L'algoritmo di identificazione in questo caso identifica i modi più sollecitati, e il controllore focalizza la sua attenzione solamente su di essi. Il secondo algoritmo punta invece a migliorare le prestazioni nell'abbattimento delle vibrazioni, controllando tutti i 17 modi, i cui parametri modali vengono aggiornati sulla base delle informazioni provenienti dall'algoritmo di identificazione. In questo caso l'identificazione viene utilizzata per aggiornare i parametri modali del sistema, nell'ipotesi che la forzante sia tale da permetterne la loro identificazione (senza ad esempio che venga identificata la frequenza della forza di disturbo, o che venga identificato uno smorzamento maggiore rispetto a quello reale per via dell'azione di controllo che introduce uno smorzamento sui modi controllati).

5.1 Richiami sul controllo modale

Il controllo di vibrazioni proposto in questa tesi fa uso di un approccio modale. Per questo motivo, nel seguito, si riporta una breve descrizione delle logiche di controllo basate su questo approccio. Data la generica equazione di un sistema meccanico:

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{R}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \Lambda_{\mathbf{F}_c}^T \mathbf{F}_c + \Lambda_{\mathbf{F}_d}^T \mathbf{F}_d \quad (5.2)$$

è possibile applicare la trasformazione in coordinate modali q :

$$\mathbf{x} = \Phi \mathbf{q} \quad (5.3)$$

ottenendo il sistema di equazioni disaccoppiate:

$$\mathbf{m}\ddot{\mathbf{q}} + \mathbf{r}\dot{\mathbf{q}} + \mathbf{k}\mathbf{q} = \Phi^T \Lambda_{\mathbf{F}_c}^T \mathbf{F}_c + \Phi^T \Lambda_{\mathbf{F}_d}^T \mathbf{F}_d \quad (5.4)$$

dove F_c può essere un'azione di controllo del tipo:

$$\mathbf{F}_c = -(\Phi^T \Lambda_{\mathbf{F}_c}^T)^{-1} (\mathbf{G}_d \dot{\mathbf{q}} + \mathbf{G}_p \mathbf{q}) \quad (5.5)$$

F_c mantiene disaccoppiati i modi ed è basata sul fatto che $(\Phi^T \Lambda_{\mathbf{F}_c}^T)$ sia invertibile, avendo quindi tanti attuatori indipendenti quanti modi. L'equazione in forma di stato del

sistema in anello chiuso è

$$\begin{bmatrix} \ddot{\mathbf{q}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} -\mathbf{m}^{-1}\mathbf{r} & -\mathbf{m}^{-1}\mathbf{k} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} \Phi^T \Lambda_{F_c}^T \\ \mathbf{0} \end{bmatrix} \mathbf{F}_c + \begin{bmatrix} \Phi^T \Lambda_{F_d}^T \\ \mathbf{0} \end{bmatrix} \mathbf{F}_d \quad (5.6)$$

Ora si può definire:

$$\mathbf{q}_x = \begin{bmatrix} \dot{\mathbf{q}} \\ \mathbf{q} \end{bmatrix} \quad (5.7)$$

Un sistema meccanico ha infiniti gradi di libertà, è quindi necessario fare una distinzione tra i modi che si vogliono e non si vogliono controllare. Eseguendo quindi la partizione tra i modi modellati e controllati (pedice R) e i modi non controllati (pedice Nc), si ottiene:

$$\begin{bmatrix} \dot{\mathbf{q}}_{xR} \\ \dot{\mathbf{q}}_{xNc} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{Nc} \end{bmatrix} \begin{bmatrix} \mathbf{q}_{xR} \\ \mathbf{q}_{xNc} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{cR} \\ \mathbf{C}_{cNc} \end{bmatrix} \mathbf{F}_c + \begin{bmatrix} \mathbf{B}_{dR} \\ \mathbf{C}_{dNc} \end{bmatrix} \mathbf{F}_d \quad (5.8)$$

Si aggiunge inoltre l'equazione dell'osservazione, considerando i soli modi modellati:

$$\hat{\dot{\mathbf{q}}}_{xR} = \mathbf{A}_R \hat{\mathbf{q}}_{xR} + \mathbf{B}_c \mathbf{F}_c + \mathbf{K}(\mathbf{y} - \mathbf{C}_R \hat{\mathbf{q}}_{xR}) \quad (5.9)$$

Assunto che \mathbf{y} , misura di grandezze fisiche del sistema, sia influenzata unicamente dai modi modellati e controllati, e definendo $\boldsymbol{\varepsilon} = \hat{\mathbf{q}}_{xR} - \mathbf{q}_{xR}$ la differenza fra stato stimato e stato effettivo del sistema, le equazioni 5.8 e 5.9 diventano:

$$\begin{bmatrix} \dot{\mathbf{q}}_{xR} \\ \dot{\mathbf{q}}_{xNc} \\ \dot{\boldsymbol{\varepsilon}} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_R - \mathbf{G} & \mathbf{0} & -\mathbf{G} \\ -\mathbf{B}_{cNc} \mathbf{B}_{cR}^{-1} \mathbf{G} & \mathbf{A}_{Nc} & -\mathbf{B}_{cNc} \mathbf{B}_{cR}^{-1} \mathbf{G} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_R - \mathbf{K} \mathbf{C}_R \end{bmatrix} \begin{bmatrix} \mathbf{q}_{xR} \\ \mathbf{q}_{xNc} \\ \boldsymbol{\varepsilon} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{dR} \\ \mathbf{B}_{dNc} \\ -\mathbf{B}_{dR} \end{bmatrix} \mathbf{F}_d \quad (5.10)$$

L'equazione 5.10 è importante per comprendere il fenomeno dello spillover di controllo, fenomeno per il quale la forza di controllo influenza anche i modi non modellati. Matematicamente questo fenomeno è visibile nella matrice $\mathbf{B}_{cNc} \mathbf{B}_{cR}^{-1} \mathbf{G}$. Gli autovalori della matrice del sistema, nel caso dell'equazione 5.10, sono influenzati solamente dagli elementi sulla diagonale, essendo una matrice triangolare a blocchi. Questo fa sì che lo spillover di controllo si limita a peggiorare le prestazioni del sistema ma non lo rende instabile. Normalmente le misure fisiche \mathbf{y} non dipendono solamente dai modi modellati, ma è presente anche un contributo dei restanti modi: questo fenomeno genera

il cosiddetto spillover di osservazione. Si ha dunque $\mathbf{y} = \mathbf{C}_R \mathbf{q}_{xR} + \mathbf{C}_{NC} \mathbf{q}_{xNC}$, va quindi modificata l'equazione 5.10:

$$\begin{bmatrix} \dot{\mathbf{q}}_{xR} \\ \dot{\mathbf{q}}_{xNC} \\ \dot{\boldsymbol{\varepsilon}} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_R - \mathbf{G} & \mathbf{0} & -\mathbf{G} \\ -\mathbf{B}_{cNC} \mathbf{B}_{cR}^{-1} \mathbf{G} & \mathbf{A}_{NC} & -\mathbf{B}_{cNC} \mathbf{B}_{cR}^{-1} \mathbf{G} \\ \mathbf{0} & \mathbf{K} \mathbf{C}_{NC} & \mathbf{A}_R - \mathbf{K} \mathbf{C}_R \end{bmatrix} \begin{bmatrix} \mathbf{q}_{xR} \\ \mathbf{q}_{xNC} \\ \boldsymbol{\varepsilon} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{dR} \\ \mathbf{B}_{dNC} \\ -\mathbf{B}_{dR} \end{bmatrix} \mathbf{F}_d \quad (5.11)$$

L'introduzione di $\mathbf{K} \mathbf{C}_{NC}$ fa sì che gli autovalori del sistema non dipendano solamente dai termini sulla diagonale. L'accoppiamento fra equazione del sistema e dell'osservatore modifica gli autovalori del sistema, facendo sì che possa essere instabilizzato dall'azione di controllo.

5.2 Controllo ottimo LQR

La teoria del controllo ottimo si è sviluppata in concomitanza con l'esplorazione spaziale, con l'avvento dei calcolatori e della robotica e l'obiettivo è quello di sviluppare sistemi di controllo robusti e stabili adatti a complessi sistemi MIMO [27]. Il controllo ottimo consiste nello scegliere una legge di controllo che minimizzi una funzione obiettivo o funzione di costo, che può includere:

- una misura dell'errore;
- una misura della forza di controllo;
- ogni altra caratteristica di interesse ingegneristico.

Dato un generico sistema dinamico in forma di stato:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u} \\ \mathbf{y} = \mathbf{C} \mathbf{x} + \mathbf{D} \mathbf{u} \end{cases} \quad (5.12)$$

si vuole mantenere lo stato $x(t)$ del sistema il più possibile prossimo ad uno stato di riferimento $x_{rif}(t)$ in un istante di tempo compreso tra t_0 e t_f . I guadagni del controllore vengono scelti in modo tale che, retroazionando lo stato, venga minimizzata una funzione di costo J basata sulla prestazione [27]:

$$J = \int_{t_0}^{t_f} ([x(t) - x_{rif}(t)]^T \mathbf{Q} [x(t) - x_{rif}(t)] + c_c^T(t) \mathbf{R} u_c(t)) dt \quad (5.13)$$

La matrice \mathbf{Q} pesa diversamente le singole componenti del vettore di stato rispetto allo scostamento dal riferimento. Se in un sistema è più importante il controllo delle basse frequenze, il valore della matrice \mathbf{Q} sarà più alto per i modi in corrispondenza di tali frequenze. La matrice \mathbf{R} pesa invece l'entità dell'azione di controllo. È intuibile come sia importante il valore relativo tra gli elementi delle 2 matrici: nelle logiche di controllo analizzate in questo capitolo, verrà definita la matrice \mathbf{R} come matrice identità, lasciando che sia la matrice \mathbf{Q} ad avere in sé le informazioni desiderate affinché venga dato un diverso peso alle diverse componenti.

Nel caso in cui il processo debba essere controllato per un tempo infinito, con il sistema retto da un'equazione lineare tempo invariante, il limite superiore dell'integrale della 5.13 diventa infinito, ed è possibile dimostrare che il guadagno ottimo della legge di controllo è:

$$\mathbf{u}_c = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{K}\mathbf{x} = -\mathbf{G}\mathbf{x} \quad (5.14)$$

dove la matrice \mathbf{K} è ottenuta risolvendo l'equazione algebrica di Riccati:

$$\mathbf{K}\mathbf{A} - \mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{K} + \mathbf{Q} + \mathbf{A}^T\mathbf{K} = \mathbf{0} \quad (5.15)$$

Per realizzare un controllo di vibrazioni che controlli n_c modi, occorre stimare dalle misure y del sistema, le coordinate modali q degli n_o modi osservati. La relazione che lega le coordinate fisiche y alle coordinate modali q è:

$$\hat{\mathbf{q}} = \mathbf{A}\mathbf{q} + \mathbf{B}\mathbf{F}_c + \mathbf{K}(y - \mathbf{C}\hat{\mathbf{q}}) \quad (5.16)$$

È dimostrabile che la matrice \mathbf{K} , detto guadagno di Kalman, è ottenibile minimizzando la matrice di covarianza dell'errore:

$$\mathbf{P} = \lim_{t \rightarrow \infty} E([\mathbf{x} - \hat{\mathbf{x}}][\mathbf{x} - \hat{\mathbf{x}}]^T) \quad (5.17)$$

dove è altresì dimostrabile che \mathbf{P} è l'unica soluzione definita positiva dell'equazione algebrica di Riccati.

5.3 Controllo LQR adattativo con 3 modi controllati

In questa sezione si testerà numericamente un sistema di controllo basato sulla logica LQR precedentemente introdotta. Per questo genere di controllo, è indispensabile una conoscenza preliminare del sistema che si vuole controllare, senza la quale non si hanno dati sufficienti per la risoluzione dell'equazione algebrica di Riccati, grazie alla quale si ottengono le matrici dei guadagni dell'osservatore di stato e del controllore. Sia data la struttura in fibra di carbonio introdotta ad inizio capitolo: una piastra di cui si conoscono le matrici del sistema in forma di stato. Le matrici contengono le informazioni relative a 17 modi di vibrare, 4 attuatori e 6 punti di misura. Si vuole realizzare una logica di controllo atta a controllare i 3 modi maggiormente sollecitati, nell'ipotesi che la forzante di disturbo abbia una banda spettrale molto ristretta. In questa condizione, l'algoritmo di identificazione è utile per stabilire le forzanti di disturbo a cui è soggetto il sistema, non essendo nelle condizioni in cui è utilizzabile per la stima dei parametri modali. Si può scegliere di controllare un numero minore di modi, qualora:

- si voglia limitare la forza di controllo, agendo solamente sui modi aventi frequenza propria vicina a quella della forzante di disturbo;
- si voglia ottenere un'elevata velocità di calcolo, identificando, osservando (tramite osservatore di stato) e controllando un sottoinsieme dei modi totali (nel caso in esame si identificano i parametri modali alla frequenza di 250 Hz, e il controllore e l'osservatore di stato lavorano alla frequenza di 1000 Hz);
- ci sia un basso effetto di spillover, quindi l'osservazione ed il controllo di 3 modi è possibile ed offre dei vantaggi;

Il controllo è stato testato numericamente in diverse condizioni, che vanno oltre ai motivi per i quali tale strategia è stata realizzata: in questo modo è stato possibile delimitare i limiti applicativi di una simile metodologia di controllo. L'algoritmo implementato è così strutturato:

- continua identificazione delle più importanti componenti modali presenti nel sistema, tramite l'algoritmo di identificazione costituito da Subspace Tracking più ARMA(3,2)(trattato nel capitolo 3);

- valutazione della frequenza contenuta nella matrice di stato totale (\mathbf{A}_{tot} , contenente le informazioni relative a 17 modi di vibrare, è stata identificata a priori offline) che più si avvicina alla frequenza identificata dall'algoritmo di identificazione;
- conteggio del numero di volte che la frequenza precedentemente cercata compare in un certo intervallo di tempo, da scegliere con cura sulla base della velocità con cui si modifica la forzante di disturbo agente sul sistema. Vengono utilizzati 17 contatori, uno per ogni modo di vibrare modellato del sistema. Se viene identificata l' i -esima frequenza della matrice \mathbf{A}_{tot} , verrà incrementato l' i -esimo contatore;
- scelta dei 3 modi del sistema che sono stati maggiormente identificati e quindi sollecitati, mediante la lettura dei 17 contatori;
- a partire dai 3 modi così selezionati, creazione della sottomatrice di stato (\mathbf{A}_{sub} , di dimensioni 6×6), contenente le informazioni relative ai soli 3 modi più sollecitati. Questo aggiornamento viene effettuato solamente quando l'algoritmo di identificazione è a regime, evitando di cadere nel problema già trattato dell'identificazione di frequenze non presenti. L'indice che viene utilizzato per valutare la presenza di un transitorio nel sistema di identificazione, è lo stesso di cui si serve il sistema che rende adattativo il coefficiente di oblio dell'algoritmo di identificazione (*oblio adattativo*) per la cui trattazione si rimanda al capitolo 3;
- per la combinazione di modi così selezionata, si sceglie da un dataset una coppia di matrici \mathbf{G} , una è il guadagno dell'osservatore \mathbf{G}_{oss} , l'altra il guadagno del controllo \mathbf{G}_{con} . Questa coppia di matrici è stata calcolata risolvendo l'equazione algebrica di Riccati per ogni combinazione possibile di 3 modi su un totale di 17 modi (680 combinazioni possibili). I guadagni ottimi sono stati calcolati con un algoritmo genetico (Appendice A);
- vengono mandate le forze di controllo al sistema numerico, le cui uscite (misure) verranno utilizzate dall'osservatore di stato e dall'algoritmo di identificazione (per quest'ultimo basta una sola uscita);

- i 3 modi identificati vengono controllati fino al nuovo aggiornamento della matrice A_{sub} . L'aggiornamento è effettuato solamente se il contributo dei nuovi modi identificati supera una certa soglia, questo per evitare inutili discontinuità nella forza di controllo.

Il diagramma relativo a questa logica di controllo è visibile in figura 5.1. I test numerici riportati di seguito mirano a testare l'algoritmo descritto in precedenza. Si inizierà fornendo alla piastra una forzante di disturbo a ristretta banda spettrale, costituita dalla somma di 3 sinusoidi in risonanza: ci si aspetta che l'algoritmo focalizzi l'attenzione solo sui 3 modi in risonanza. Successivamente verrà messo alla prova il sistema di identificazione, eseguendo un cambio forzante, che passerà da una frequenza ad un'altra. In questo caso l'algoritmo dovrebbe provvedere all'aggiornamento della matrice A (6X6), e il controllo dovrebbe continuare a fornire risultati soddisfacenti. Infine verrà analizzato il controllo qualora la piastra venga sollecitata con una forzante fuori dalla risonanza.

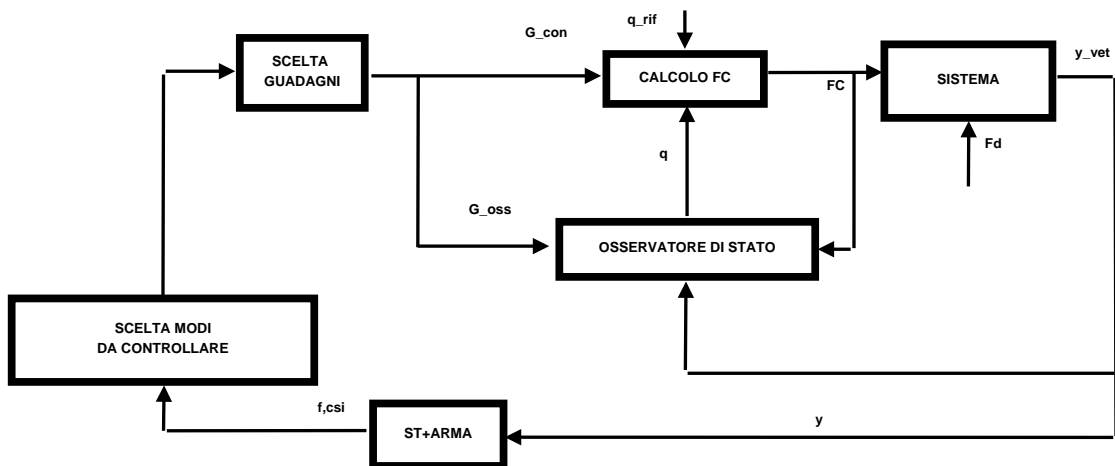


Figura 5.1: Diagramma schema di controllo a 3 modi controllati

5.3.1 Forzante data dalla somma di 3 sinusoidi in risonanza

Si testa il sistema di controllo con un disturbo dato dalla somma di 3 sinusoidi in risonanza. Le sinusoidi corrispondono alle frequenze proprie del sistema numero 1 (10.48

Hz), 5 (24.29 Hz) e 9 (48.04 Hz). La presenza di 3 sinusoidi è un presupposto affinché l'ARMA commetta degli errori di identificazione: l'azione del controllo può infatti essere più preponderante sulla sinusoidi col maggior contributo vibratorio, e l'abbattimento può essere tale da non renderla la componente più importante del segnale. Questo può comportare la presenza di più armoniche in una singola storia temporale del Subspace Tracking, con la conseguenza che l'ARMA impiega del tempo per identificarle, e durante la transizione tra una e l'altra vengono identificate frequenze non presenti nel sistema. La più grave conseguenza di questo fenomeno è la creazione di una matrice di stato \mathbf{A}_{sub} che non contiene i 3 modi più sollecitati. Chiamando nm il numero di punti necessari per l'aggiornamento di \mathbf{A}_{sub} , un valore di nm troppo piccolo è notevolmente influenzato da un eventuale transitorio dell'ARMA, un valore di nm troppo elevato rischia di compromettere il celere inseguimento della forzante di disturbo. Si è scelto un valore di nm pari a $nm = 2m$ (m è la lunghezza della finestra mobile del ST, in questo caso si è posta pari a 160). Lo spettro del sistema con e senza controllo e l'andamento nel tempo, nel caso del sistema soggetto a questo disturbo, sono riportati in figura 5.2. Si può notare l'abbattimento delle vibrazioni.

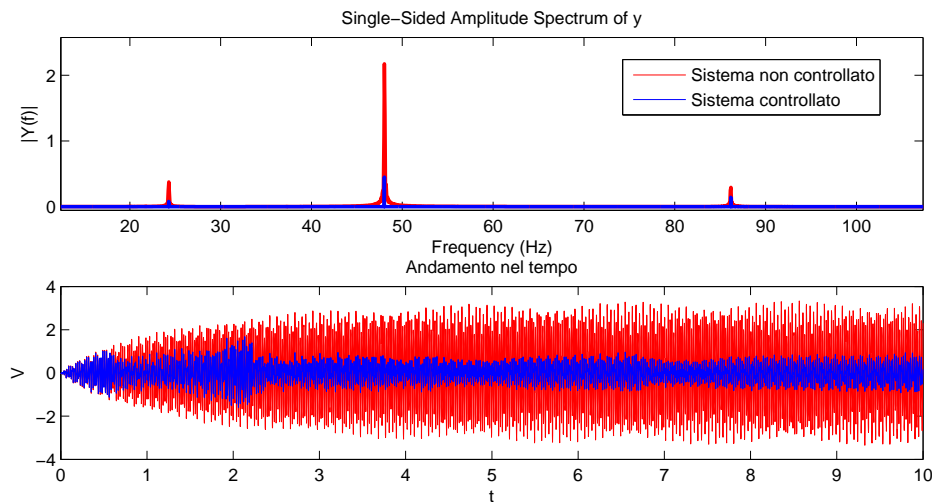


Figura 5.2: Somma di 3 sinusoidi in risonanza

5.3.2 Cambio di forzante

Si vuole testare il sistema di controllo nel caso vi sia una variazione nella forzante di disturbo. Nella prova numerica effettuata, all'istante $t_0 = 4s$ la forzante di disturbo iniziale (forzante in risonanza alla frequenza propria numero 3, a 20.07 Hz) diminuisce linearmente arrivando a 0 dopo $t_{trans} = 2s$. Nello stesso periodo di tempo, nasce una forzante alla frequenza propria del sistema numero 7, a 37.30 Hz. L'ampiezza della forzante cresce linearmente raggiungendo il suo valore di regime a $t_0 + t_{trans} = 6s$. Lo spettro del sistema con e senza controllo e l'andamento nel tempo del segnale misurato sono riportati in figura 5.3: il controllo abbatte le vibrazioni e non vengono introdotte drastiche discontinuità durante l'aggiornamento della matrice \mathbf{G}_{con} del controllore.

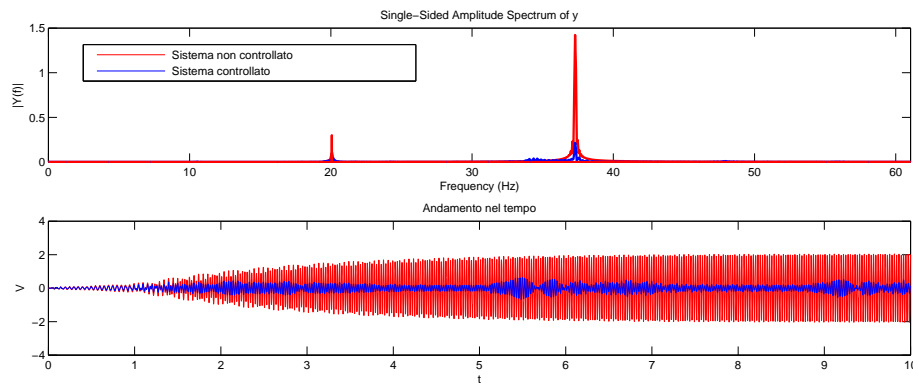


Figura 5.3: Cambio di forzante

5.3.3 Forzante fuori dalla risonanza

Si vuole analizzare il comportamento del controllo qualora venga identificata una forzante di disturbo fuori dalle risonanze del sistema. Ci si aspetta che non ci sia molta differenza tra lo spettro del sistema controllato e non controllato, per la natura stessa del controllo che è ottimizzato per abbattere frequenze prossime alla risonanza (che hanno quindi un maggiore contributo nella vibrazione). È fondamentale che la logica di controllo, trovandosi in una situazione per la quale non è stata ideata, non peggiori le prestazioni. Lo spettro del sistema con e senza controllo e l'andamento nel tempo sono

visibili in figura 5.4: si può notare che, come atteso, il controllo non introduce evidenti miglioramenti, ma non amplifica nemmeno le vibrazioni.

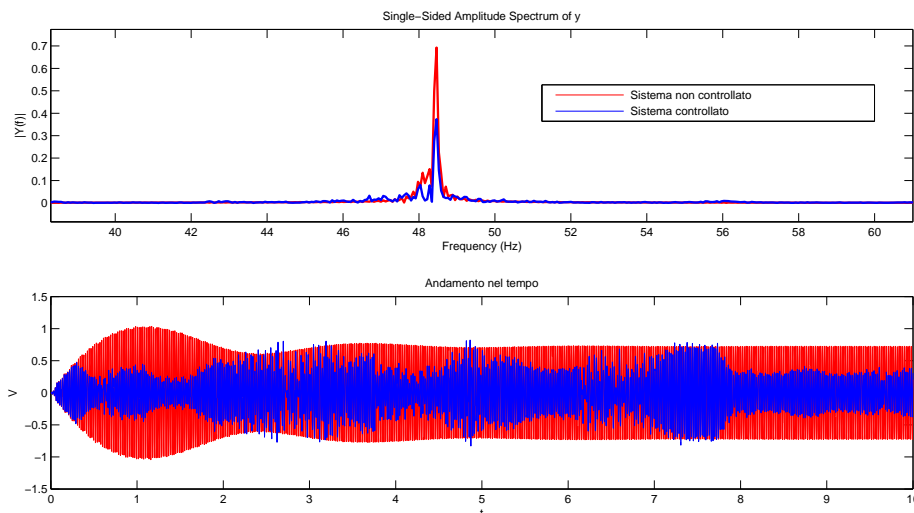


Figura 5.4: Forzante fuori risonanza

5.4 Controllo LQR adattativo con 17 modi controllati

Nell'ottica di inseguire le variazioni del sistema, e non più della forzante, si implementa una logica di controllo adattativo che sfrutta il controllore ottimo LQR (il relativo diagramma è visibile in figura 5.5). Il limite applicativo di questa tipologia di controllo è dettato dal limite di impiego dell'algoritmo di identificazione nello stimare i parametri modali, e dal tempo di calcolo molto oneroso. Per la stima delle grandezze modali, si è dotato l'algoritmo di identificazione di valori variabili del coefficiente di oblio, a seconda che l'ARMA sia o meno in stato di transitorio (a tal proposito, si rimanda al paragrafo 4.2). La scelta delle frequenze e degli smorzamenti coi quali aggiornare la matrice di stato viene eseguita attraverso il metodo statistico già discusso in precedenza nel capitolo 3. L'aggiornamento della matrice di stato viene effettuato solo sulle frequenze dei modi di vibrare identificati, e sotto l'ipotesi che il sistema di identificazione non stia identificando la frequenza della forzante. A tal proposito viene eseguito un

controllo sullo smorzamento adimensionale identificato: valori prossimi allo 0 indicano che molto probabilmente non si sta identificando la frequenza associata al modo di vibrare della piastra, ma si sta identificando la frequenza di una forzante. Le operazioni di aggiornamento della matrice di stato (di dimensioni 34X34) non possono essere eseguite alla stessa frequenza dell' algoritmo di identificazione, siccome troppo onerose computazionalmente. Inoltre è inutile eseguirle in continuazione, in quanto esse sono effettuate solamente quando il buffer bidimensionale (il contatore che conta il numero di volte in cui una certa coppia $\omega - \xi$ è stata identificata) contiene un sufficiente numero di punti. Da sottolineare come la velocità con la quale viene eseguito l'aggiornamento della matrice di stato, è sufficiente nella ragionevole ipotesi che la struttura cambi lentamente nel tempo. Si sceglie quindi di realizzare 3 cicli con frequenze di calcolo differenti:

- CICLO VELOCE: è il ciclo dell'osservatore di stato e del calcolo della forza di controllo (800 Hz);
- CICLO MEDIO: è il ciclo dell' algoritmo di identificazione (200 Hz);
- CICLO LENTO: è il ciclo del calcolo dell'aggiornamento della matrice di stato (0.25 Hz);

Si vuole fare presente che la velocità del ciclo lento dipende dalla discretizzazione effettuata sul campo delle frequenze e degli smorzamenti: più la griglia di discretizzazione è fitta, più aumenta il tempo di calcolo. Inoltre il numero di punti in ingresso dopo il quale il buffer viene azzerato, deve essere tale da permetterne il suo azzeramento alla stessa frequenza del ciclo lento. Si è visto che alla frequenza del ciclo medio (ST + ARMA) di 200 Hz, si identificano bene i parametri modali azzerando il buffer ogni 800 punti acquisiti, cioè ogni 4 secondi (alla frequenza di 200 Hz). Di conseguenza la velocità del ciclo lento viene imposta a 0.25 Hz, e rappresenta l'effettiva frequenza con cui i dati provenienti dall'identificazione dei parametri modali forniscono valori attendibili. L'ultima velocità che resta da definire è la velocità del ciclo veloce: essa è vincolata principalmente alla potenza dell'elaboratore e al fatto che le frequenze devono essere multiple fra di loro, si sceglie quindi la frequenza di 800 Hz.

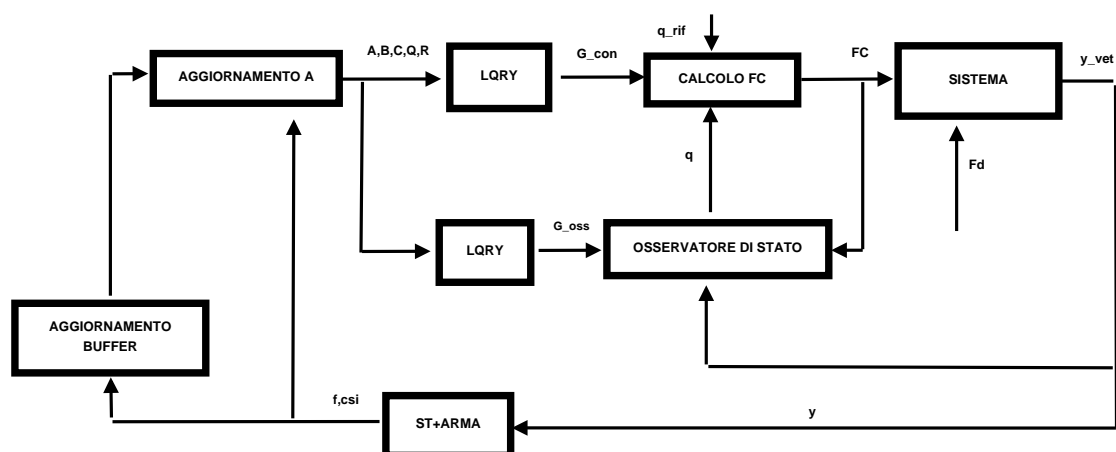


Figura 5.5: Diagramma del controllo LQR adattativo a 17 modi

5.4.1 Forzante random sul sistema modificato

Per simulare numericamente una possibile applicazione sperimentale di questo controllo, viene variata leggermente la matrice di stato usata nelle precedenti simulazioni. In tabella 5.2 si possono vedere le nuove frequenze proprie del sistema.

NUMERO MODO	FREQUENZA MODIFICATA	SMORZAMENTO ADIMENSIONALE
1	9.3792 Hz	0.0025
2	18.3476 Hz	0.0035
3	19.0752 Hz	0.0016
4	22.2524 Hz	0.0058
5	29.5767 Hz	0.0019
6	33.8921 Hz	0.0025
7	38.3004 Hz	0.0017
8	47.8379 Hz	0.0016
9	49.1400 Hz	0.0013
10	51.6815 Hz	0.0012
11	66.4946 Hz	0.0037
12	68.2791 Hz	0.0050

13	69.8417 Hz	0.0022
14	77.1950 Hz	0.0022
15	83.2475 Hz	0.0012
16	86.3988 Hz	0.0043
17	89.6900 Hz	0.0049

Tabella 5.2: Parametri modali della piastra identificati offline

Il controllo adattativo implementato (figura 5.6) si è comportato meglio rispetto ad un controllo non adattativo (figura 5.7). In figura 5.8 c'è invece il controllo adattativo che controlla il sistema non modificato (vedere tabella 5.1). Il controllo adattativo ha tuttavia delle imprecisioni, siccome non vengono identificate tutte le frequenze proprie, ma vengono identificate i parametri modali di 6 modi. Questa scelta è vincolata dall'onere computazionale dell'algoritmo. Inoltre le frequenze identificate in RT, risentono della discretizzazione effettuata (vedere paragrafo 4.3).

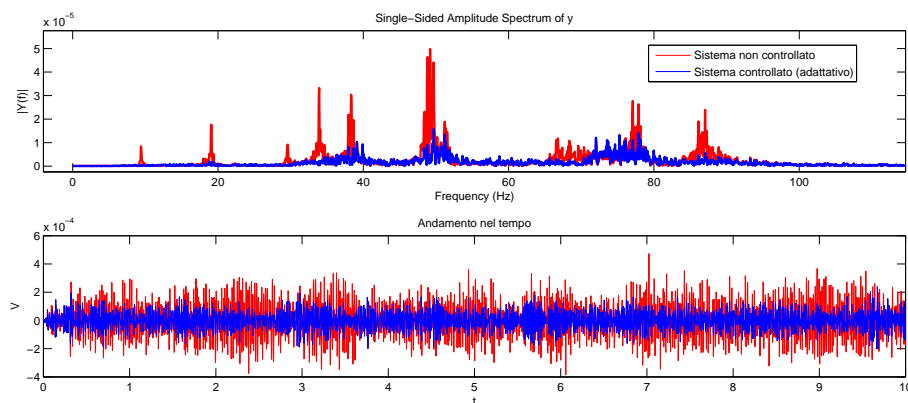


Figura 5.6: Forzante random 0-100 Hz, matrice di stato modificata. Controllo LQR adattativo.

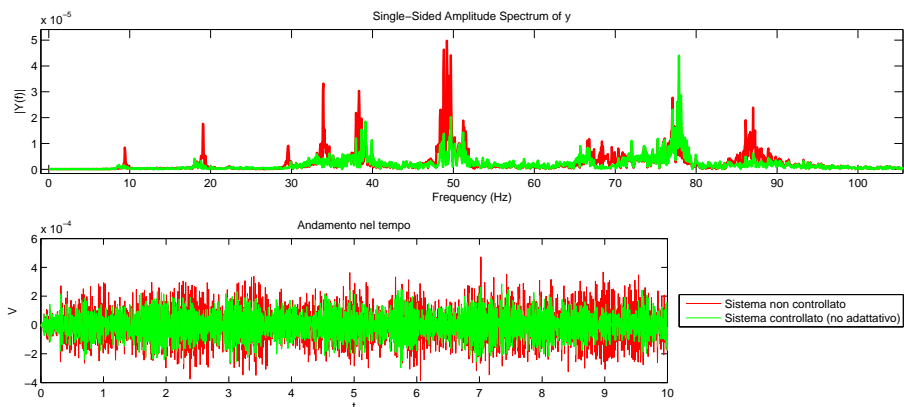


Figura 5.7: Forzante random 0-100 Hz, matrice di stato modificata. Controllo LQR non adattativo.

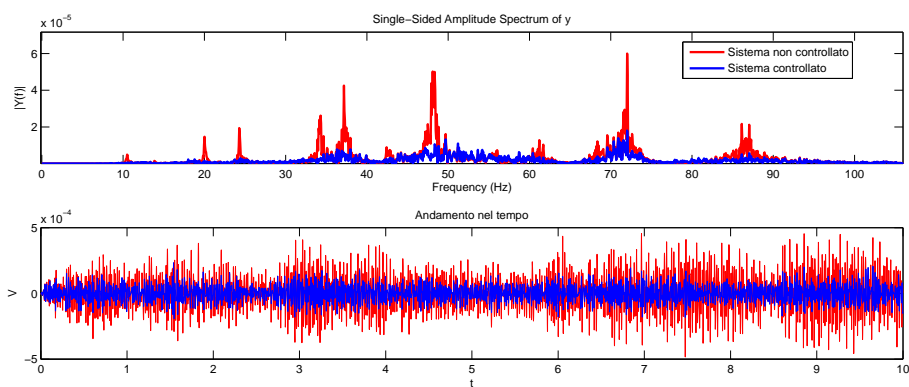


Figura 5.8: Forzante random 0-100 Hz, matrice di stato non modificata. Controllo LQR adattativo.

5.4.2 Controllo Gain Scheduling su 17 modi

Si è visto che la tipologia di controllo adattativo introdotta al paragrafo 5.4 è di difficile implementazione sperimentale, non potendo essere supportata dalla piattaforma DSpace. Si sceglie dunque di ricorrere ad una sua semplificazione, dove l'algoritmo di identificazione è utilizzato per l'aggiornamento della matrice di stato e per la scelta delle matrici dei guadagni a seconda dello stato in cui il sistema si trova.

In particolare, l'algoritmo di identificazione RT è in grado di riconoscere le frequenze proprie del sistema solo in certe condizioni (si rimanda al capitolo 4). Utilizzando lo smorzamento adimensionale identificato, è possibile decidere se aggiornare la matrice di stato con la frequenza identificata dall'algoritmo: si è visto che a valori dello smorzamento adimensionale molto bassi, la frequenza identificata dall'algoritmo può essere attribuita alla frequenza della forzante di disturbo (a tal proposito si rimanda al capitolo 4). Vengono inoltre tabulate 3 matrici dei guadagni, a seconda che il sistema sia forzato con una delle seguenti tipologie di disturbo:

- forzante random 0-100 Hz;
- forzante random 0-50 Hz;
- forzante random 50-100 Hz;

La condizione di forzamento in cui si trova il sistema viene valutata sulla base delle frequenze identificate: ad esempio se vengono identificate in prevalenza frequenze sotto i 50 Hz, si suppone che la forza di disturbo abbia uno spettro spostato verso le basse frequenze. Se la differenza tra il numero di frequenze identificate sopra i 50 Hz e quelle identificate sotto i 50 Hz non è molta, si ipotizza che il sistema sia forzato con un disturbo ad ampia banda spettrale. Per ciascuna condizione di forzamento, sono state calcolate delle matrici dei guadagni con la logica del controllo LQR. Le matrici \mathbf{G} dei guadagni, sono state calcolate utilizzando matrici \mathbf{Q} che pesano in modo diverso l'importanza dei singoli modi di vibrare all'interno dell'equazione 5.13 del funzionale. Per il calcolo delle matrici \mathbf{G} dei guadagni, viene risolta l'equazione algebrica di Riccati. Oltre alla matrice di stato, vengono coinvolte nei calcoli le altre matrici del sistema in

forma di stato (matrici \mathbf{B} e \mathbf{C}), che devono essere identificate off-line, prima di procedere al controllo. Per poter confrontare questa logica di controllo con quella esposta al paragrafo 5.4, viene forzato il sistema modificato come da tabella 5.2, con un disturbo random tra 0 e 100 Hz. I risultati della simulazione numerica sono visibili in figura 5.10, dalla quale si può vedere come le vibrazioni vengono abbattute e si hanno miglioramenti rispetto ad un controllo non adattativo (figura 5.7), ma le prestazioni sono inferiori rispetto al controllo LQR adattativo.

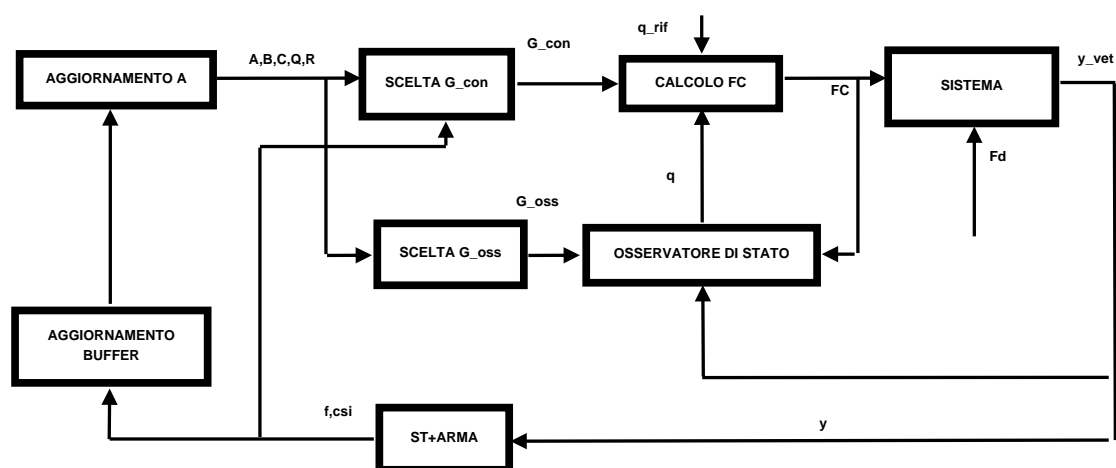


Figura 5.9: Diagramma controllo Gain Scheduling. 17 modi controllati.

5.5 Controllo adattativo - Prove sperimentali

In questa sezione verranno sperimentate differenti logiche di controllo. Inizialmente si valuterà l'influenza dell'aggiornamento della matrice di stato all'interno della logica di controllo LQR. Successivamente verrà testata la logica di controllo adattativo presentata nel paragrafo 5.3. Il sistema oggetto di sperimentazione è la piastra in fibra di carbonio descritta nel capitolo 4.

5.5.1 LQR con aggiornamento della matrice di stato

Viene ora testata l'influenza dell'aggiornamento della matrice di stato all'interno di una logica di controllo LQR, dove il guadagno è fisso ed è stato calcolato per il controllo

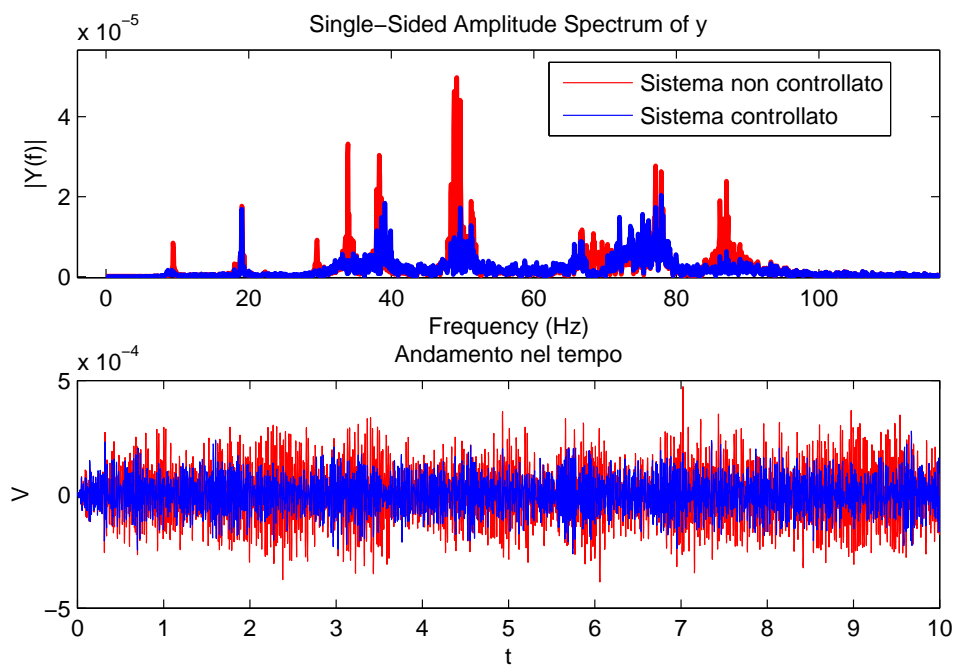


Figura 5.10: Forzamento random 0-100 Hz con matrice di stato modificata. Controllo adattativo Gain Scheduling. 17 modi controllati

degli 8 modi elencati in tabella 5.3.

Il numero di modi identificati dall'algoritmo di identificazione è stato fissato a 6, e il buffer bidimensionale che conta il numero di ricorrenze di una certa coppia frequenza-smorzamento (per un approfondimento si rimanda al paragrafo 4.3) è di dimensioni 100 X 100. Grazie a questi accorgimenti si riesce a portare la velocità del ciclo veloce (controllore e osservatore di stato) a 2 kHz, mentre la velocità dell'algoritmo di identificazione RT è di 200 Hz e i parametri modali vengono aggiornati ogni 0.25 Hz.

NUMERO MODO	FREQUENZA
1	16.09 Hz
2	39.67 Hz
3	44.23 Hz
4	53.00 Hz
5	59.23 Hz
6	62.13 Hz
7	81.09 Hz
8	87.99 Hz

Tabella 5.3: Modi controllati

Il sistema di controllo viene testato con una forza di disturbo random tra 0 e 100 Hz. In figura 5.11 è visibile lo spettro relativo al sistema controllato con controllo adattativo, mentre in figura 5.12 il sistema è controllato in modo non adattativo. L'adattatività di questo controllo influisce sull'aggiornamento della matrice di stato (utilizzata dall'osservatore di stato).

Le differenze tra controllo adattativo e controllo non adattativo non sono molte, questo è dovuto al fatto che la matrice di stato utilizzata dal controllo non adattativo descrive bene il sistema.

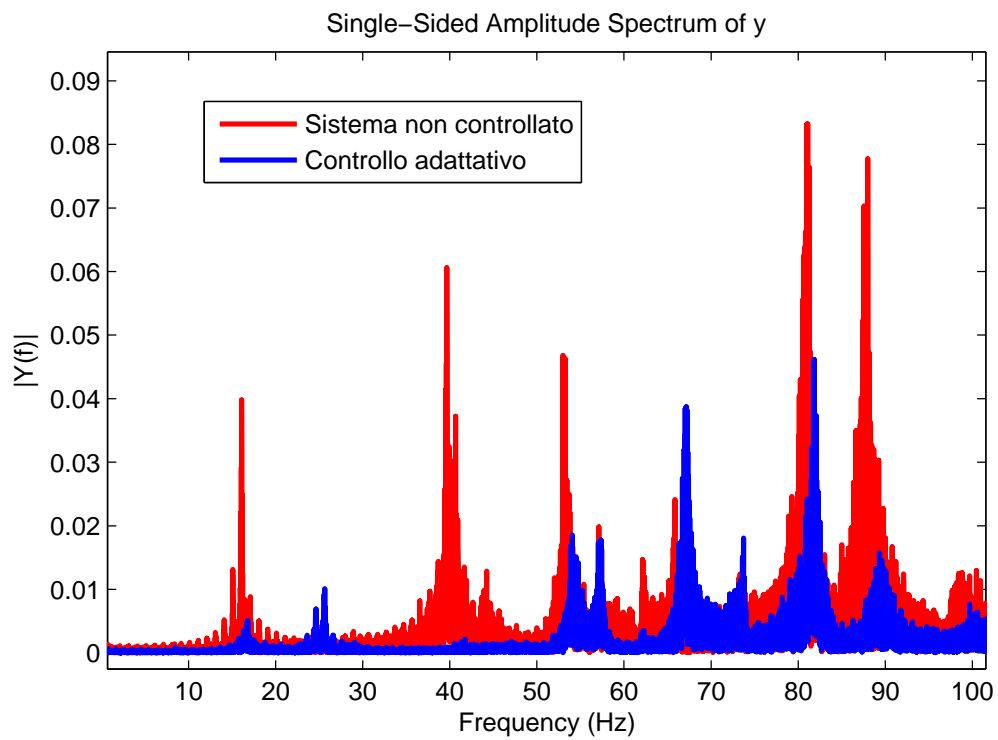


Figura 5.11: Prove sperimentali. Forzante random 0-100 Hz. Controllo adattativo

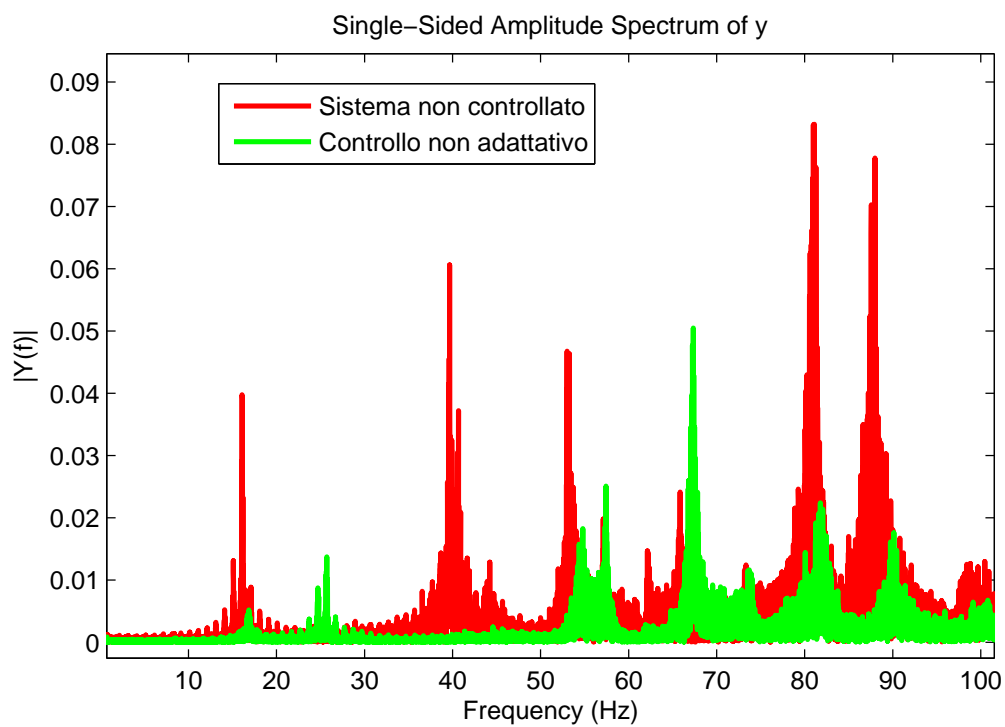


Figura 5.12: Prove sperimentali. Forzante random 0-100 Hz. Controllo non adattativo

5.5.2 Gain Scheduling, 3 modi controllati

L'obiettivo di questo controllo è poter controllare ogni combinazione possibile di 3 modi, su un totale di 17 modi modellati numericamente del sistema. La logica di controllo è descritta nel paragrafo 5.3, ma a differenza delle prove numeriche, nelle prove sperimentali verrà inizialmente utilizzata una forzante random tra 0 e 100 Hz, e successivamente una forzante random tra 0 e 50 Hz. L'utilizzo di forzanti random ad ampia banda spettrale permette l'identificazione dei parametri modali, e di conseguenza l'aggiornamento della matrice di stato.

In questo caso l'adattatività consiste nell'aggiornamento della matrice di stato (utilizzata dall'osservatore di stato) e dalla scelta dei guadagni per i 3 modi più sollecitati del sistema (che vengono selezionati dall' algoritmo di identificazione RT). Il controllo adattativo viene confrontato con un controllo LQR dove la matrice dei guadagni è costante ed è tale da controllare i 3 modi più sollecitati del sistema (selezionati in seguito ad un'analisi di Fourier off-line con sistema forzato su tutta la sua banda spettrale con un disturbo random tra 0 e 100 Hz). Questi modi sono quelli corrispondenti alle frequenze di 57.36 Hz, 66.69 Hz e 81.09 Hz.

Forzando il sistema con una forzante random tra 0 e 100 Hz, le differenze tra logica di controllo adattativa (figura 5.13) e non adattativa (figura 5.14) sono minime, in quanto l'adattatività consiste solamente nell'aggiornamento della matrice di stato utilizzata nei calcoli dall'osservatore (siccome la matrice dei guadagni selezionata dall' algoritmo di identificazione RT è la stessa che utilizza il controllo LQR non adattativo).

Forzando il sistema con una forzante random tra 0 e 50 hz, si notano delle differenze. Mentre il controllo non adattativo (figura 5.16) continua a controllare gli stessi modi, il controllo adattativo (figura 5.15) focalizza la sua attenzione sui 3 modi più sollecitati del sistema. Le prestazioni del controllo adattativo sono migliori se confrontate con il controllo non adattativo.

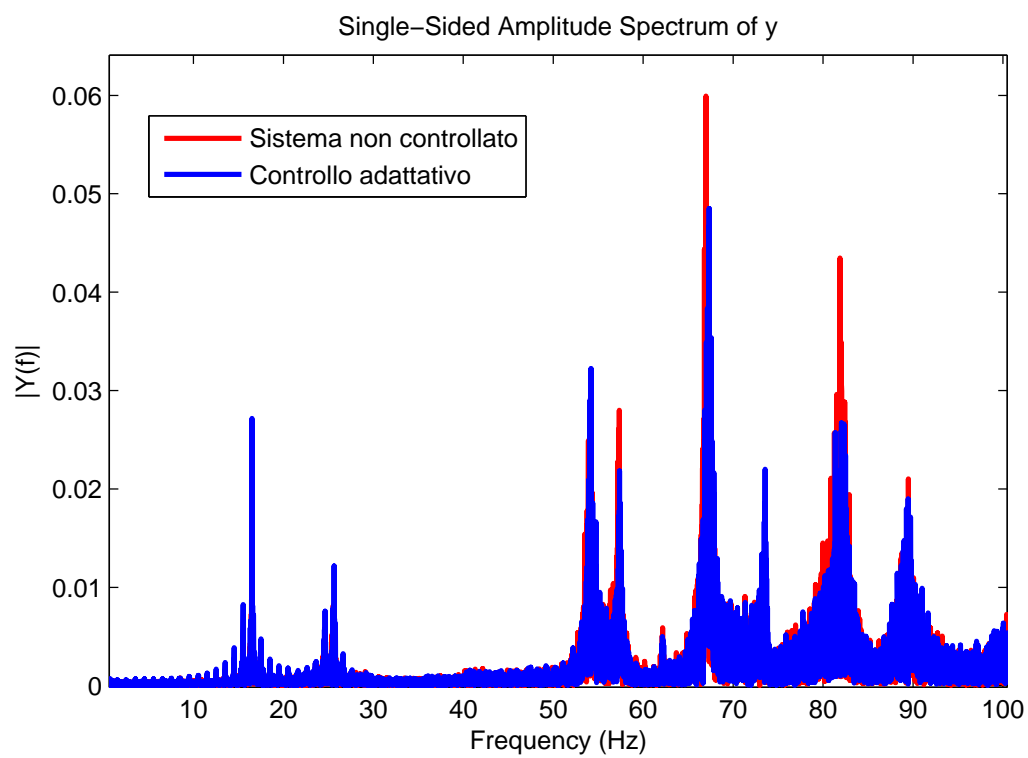


Figura 5.13: Prove sperimentali. Forzante random 0-100 Hz. Controllo adattativo

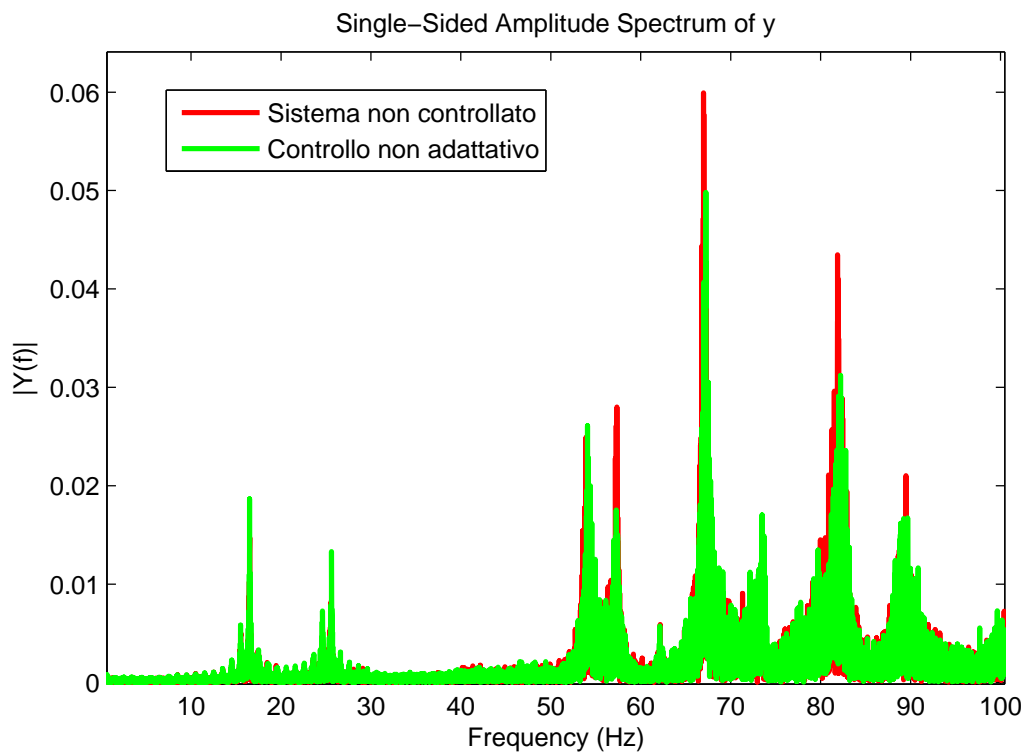


Figura 5.14: Prove sperimentali. Forzante random 0-100 Hz. Controllo non adattativo

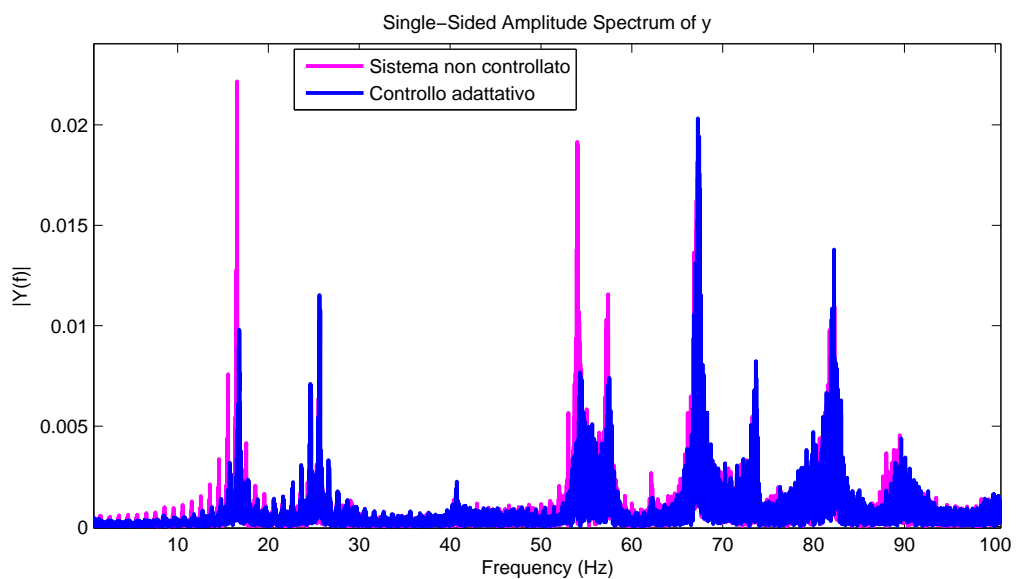


Figura 5.15: Prove sperimentali. Forzante random 0-50 Hz. Controllo adattativo

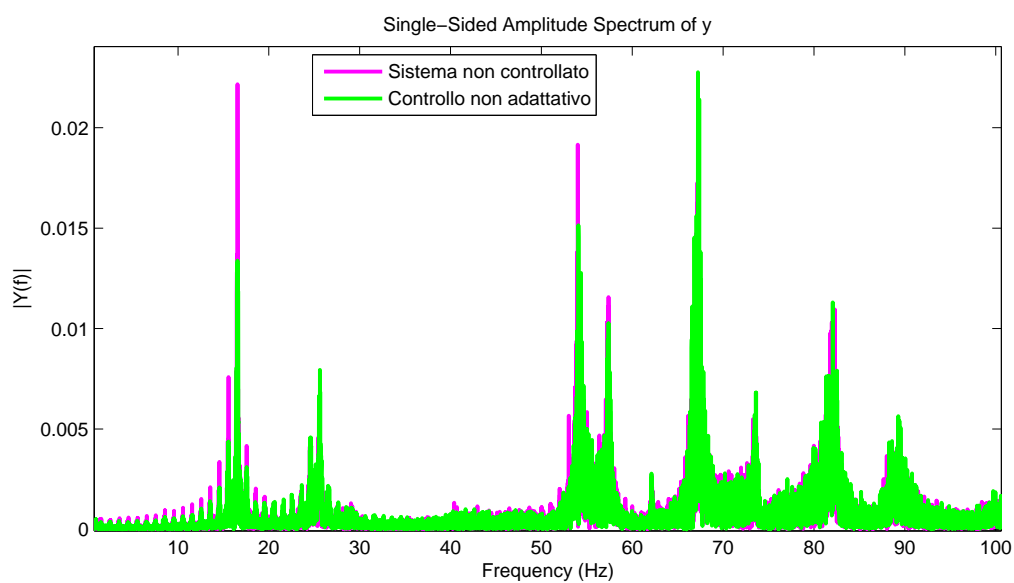


Figura 5.16: Prove sperimentali. Forzante random 0-50 Hz. Controllo non adattativo

Conclusioni

Con l'obiettivo di realizzare un controllo adattativo per la riduzione delle vibrazioni in una smart structure, è stato implementato un algoritmo di identificazione Real-Time, ottimizzato per poter essere applicato sperimentalmente. Questo algoritmo è stato testato sperimentalmente su una piastra in fibra di carbonio (smart structure). È emerso che:

- è possibile identificare correttamente le frequenze proprie qualora la forzante di disturbo sia un rumore bianco a larga banda spettrale (0-100 Hz, 0-50 Hz);
- l'algoritmo ha difficoltà nell'identificazione dello smorzamento adimensionale corrispondente alle frequenze identificate;
- i valori di smorzamento adimensionale identificati sono indicativi dello stato in cui si trova il sistema: è possibile ad esempio valutare se le vibrazioni stanno incrementando (smorzamento adimensionale <0) o se si sta identificando la frequenza della forzante di disturbo (smorzamento adimensionale vicino a 0);

In base a queste considerazioni è possibile realizzare un meccanismo adattativo che:

- aggiorni la matrice dei guadagni del controllore a seconda della forzante di disturbo identificata;
- aggiorni la matrice di stato con i valori delle frequenze proprie identificate, qualora ci si trovi nelle condizioni nelle quali questa identificazione è corretta. Dunque viene aggiornata la matrice di stato se i valori dello smorzamento adimensionale identificati rientrano in una certa banda di confidenza calcolata sulla base dell'interpolazione della curva di Rayleigh: ad esempio un valore di smorzamento

adimensionale al di sotto del limite inferiore della banda di confidenza indica che con buone probabilità si sta identificando la frequenza della forzante. Per un meccanismo adattativo di questo tipo, è necessaria un'analisi preliminare off-line per la conoscenza della matrice di stato e delle frequenze proprie. Questa matrice di stato viene successivamente aggiornata coi dati provenienti dall'algoritmo di identificazione real-time. L'aggiornamento viene effettuato qualora le frequenze identificate in real-time cadano in un intorno delle frequenze proprie identificate precedentemente off-line.

Sono state testate numericamente diverse logiche di controllo:

- Gain Scheduling sui 3 modi del sistema maggiormente sollecitati (su un totale di 17 modi modellati numericamente): l'algoritmo di identificazione implementato è utile per la scelta dei 3 modi da controllare. Questa logica riduce la forza di controllo e il tempo di calcolo, e fornisce risultati soddisfacenti qualora la forzante di disturbo abbia un ristretto campo spettrale e forzi in risonanza il sistema.
- LQR adattativo su 17 modi: grazie all'algoritmo di identificazione RT, è reso possibile un'aggiornamento della matrice di stato. Viene risolta l'equazione algebrica di Riccati in modo periodico, grazie alla quale si calcolano i guadagni del controllore e dell'osservatore di stato. Questa logica di controllo è stata testata su un sistema al quale è stata modificata la matrice di stato.
- Gain Scheduling su 17 modi: viene testato in 3 differenti tipologie di disturbo. In particolare viene testato su un disturbo random tra 0-100 Hz, su un random tra 0 e 50 Hz e su un random tra 50 e 100 Hz. L'algoritmo di identificazione riconosce il tipo di forzamento agente sul sistema, e aggiorna in automatico la matrice dei guadagni, scegliendo da un dataset la più adatta. Viene inoltre aggiornata periodicamente la matrice di stato.

Infine vengono effettuate delle prove sperimentali di controllo. Si valuta inizialmente l'influenza dell'aggiornamento della matrice di stato all'interno di una logica di controllo LQR. La matrice di stato con cui lavora l'osservatore è infatti aggiornata con i dati provenienti dall'algoritmo di identificazione RT. Questo aggiornamento non compromette la stabilità del sistema, ma le prestazioni del controllo non subiscono sostanziali

miglioramenti, se confrontate con un controllo LQR dove l'osservatore di stato lavora sempre con le matrici costanti (che evidentemente, descrivono bene il sistema). Viene successivamente sperimentata la logica di controllo Gain Scheduling, dove per ogni combinazione possibile di 3 modi (su un totale di 17 modi modellati numericamente), viene scelta un'opportuna matrice, calcolata in precedenza off-line con la logica LQR. Il controllo così implementato viene testato con una forzante random tra 0 e 50 Hz: il sistema identifica correttamente i modi più sollecitati dal disturbo, sceglie di conseguenza delle opportune matrici dei guadagni per il controllore e per l'osservatore, ed effettua l'aggiornamento della matrice di stato.

Questa tipologia di controllo porta notevoli miglioramenti rispetto al corrispondente controllo LQR non adattativo (dove vengono controllati sempre i soliti 3 modi).

5.6 Sviluppi per il futuro

La presenza di un solo sensore destinato all'identificazione real-time, fa sì che esso debba essere scelto a priori affinché vengano identificate al meglio le componenti modali relative ai modi di vibrare ritenuti fondamentali da controllare. È quindi necessaria una valutazione preliminare offline per la scelta del sensore su cui questi modi danno il maggiore contributo. A tal proposito, un possibile sviluppo è rendere possibile l'utilizzo simultaneo di più sensori per l'identificazione. Un forte limite dell'algoritmo di identificazione sviluppato, è rappresentato dall'errore nella stima degli smorzamenti adimensionali: un'idea sarebbe provare ad utilizzare una rete neurale correttamente addestrata al posto dell'ARMA. Un altro traguardo da raggiungere, è un sistema che identifichi in real-time non solamente i parametri modali, ma anche le altre matrici del sistema in forma di stato. Per quanto riguarda la parte di controllo, i risultati numerici e sperimentali sono soddisfacenti. Tuttavia è necessario fare un'importante considerazione a proposito del controllo adattativo LQR. Su piattaforma DSpace, non è stato possibile risolvere l'equazione algebrica di Riccati, ne consegue che non è stato possibile provare sperimentalmente le logiche di controllo adattativo che ne prevedevano la risoluzione in modo periodico (cioè ogni volta che veniva aggiornata la matrice di stato). Il limite è di natura informatica, e si farebbe un passo avanti se questo ostacolo

venisse aggirato. Focalizzando l'attenzione sulla parte analitica, sarebbe utile studiare la stabilità delle diverse logiche di controllo di vibrazioni implementabili su una smart structure. Inoltre si propone uno studio per scartare automaticamente i parametri modali identificati RT e che possono essere considerati come errati. A tal proposito si ricorda che il modello ARMA utilizzato (ARMA di ordine 3 sull'uscita del sistema) calcola una coppia di poli e un polo reale: la coppia di poli (poli complessi coniugati) viene utilizzata per la stima dei parametri modali. Si è visto empiricamente che il valore del polo reale può essere un indice che valuta eventuali errori nella stima RT dei parametri modali identificati.

Appendice A

Algoritmo genetico

Le matrici Q presenti nell'equazione 5.13 sono frutto di un'ottimizzazione mediante algoritmi genetici (AG). Gli algoritmi genetici sono procedure adattative finalizzate alla risoluzione di problemi di ricerca ed ottimizzazione basate concettualmente sui principi che regolano l'evoluzione naturale della specie. L'idea che sta alla base degli AG è quella di scegliere le soluzioni migliori secondo il principio della selezione naturale di Darwin (cioè della sopravvivenza del più adatto) e di ricombinarle fra loro in maniera tale che esse evolvano verso un punto di ottimo. Le caratteristiche principali degli algoritmi genetici sono:

- gli AG utilizzano una codifica delle variabili di progetto e non le variabili stesse;
- essi operano su popolazioni di individui (insieme di soluzioni) invece che su singole soluzioni;
- essi utilizzano unicamente i valori assunti dalle funzioni obiettivo non richiedendo alcuna valutazione di gradienti o derivate;
- gli AG usano regole probabilistiche e non deterministiche;

Un AG parte da un certo numero di possibili soluzioni (individui) chiamate popolazione e provvede a farle evolvere nel corso dell'esecuzione: a ciascuna iterazione, esso opera una selezione di individui della popolazione corrente, impiegandoli per generare nuovi elementi che andranno a costituire la nuova popolazione per la generazione seguente,

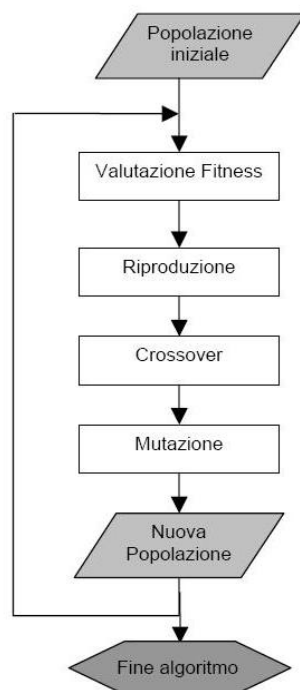


Figura A.1: Sequenza di operazioni di un algoritmo genetico

corrispondente alla nuova iterazione [14]. È necessario definire una funzione obiettivo da massimizzare, che regola l'evoluzione delle generazioni. Le operazioni fondamentali di un algoritmo genetico sono:

- creazione di una popolazione di individui, dove a ciascun individuo è associato un certo valore delle variabili di progetto;
- valutazione della funzione obiettivo di ciascun individuo;
- codifica delle variabili in numeri binari;
- scelta degli individui il cui accoppiamento determinerà la nascita di nuovi individui, la probabilità di ciascun individuo di essere scelto per la riproduzione è legata al valore della sua funzione obiettivo: individui con una fitness elevata (cioè con un valore elevato della funzione obiettivo) avranno maggiori probabilità di essere selezionati;

- fusione dei geni (cioè dei valori di 0 e 1 presenti nella stringa binaria) con il metodo del crossover: viene tagliato il DNA (sequenza binaria) dei genitori in frammenti, e mischiato per la creazione di nuovi individui;
- mutazione casuale di alcune stringhe di DNA;
- decodifica delle stringhe binarie;
- valutazione della funzione obiettivo della popolazione (genitori e figli);
- eliminazione degli individui con la fitness più bassa;
- l'algoritmo termina dopo un numero prestabilito di generazioni.

Gli AG sono particolarmente indicati in funzioni con presenza di minimi locali e variabili che possono assumere solo valori discreti. Per la determinazione delle matrici \mathbf{Q} grazie alle quali è stato possibile il calcolo dei guadagni dell'osservatore di stato e del controllore, si è imposto che le matrici \mathbf{Q} fossero diagonali, è stato quindi supposto che il sistema fosse disaccoppiato (approccio modale). Ciascun elemento sulla diagonale delle matrici, è considerato una variabile indipendente. Il motivo per il quale non si è scelto di ricavare direttamente le matrici dei guadagni \mathbf{G} , è dovuto al fatto che esse devono essere calcolate periodicamente risolvendo l'equazione algebrica di Riccati. L'algoritmo genetico utilizzato ottimizza una funzione obiettivo che:

- minimizza la somma della parte reale degli autovalori del sistema controllato, e attribuisce una forte penalità qualora un solo autovalore abbia parte reale maggiore di una certa soglia b o abbia parte immaginaria uguale a 0;
- attribuisce forti penalità qualora un singolo autovalore della matrice del sistema osservato e controllato abbia parte reale maggiore di 0 o abbia parte immaginaria uguale a 0;

Il valore della soglia b non è noto a priori. Usando un valore di b troppo elevato, si rischia di non raggiungere buone prestazioni di controllo. Usando una soglia di b troppo bassa e proibitiva, si rischia che essa sia irraggiungibile, con il rischio che l'algoritmo non trovi la strada giusta verso il punto di ottimo. Si è visto che i migliori risultati si

sono ottenuti usando un valore della soglia b variabile: il valore della soglia viene abbassato qualora il 10% degli individui riesca a superarlo. C'è però il rischio che nessun individuo riesca a raggiungere la nuova soglia con tutti i suoi autovalori: questo può comportare l'ottimizzazione di certi autovalori a discapito di altri. Per ovviare a questo inconveniente, si attribuisce la massima penalità (funzione obiettivo =0) agli individui che non hanno la parte reale di tutti gli autovalori inferiore della precedente soglia, che è quindi fondamentale salvare ogni volta che una nuova soglia b viene raggiunta.

Appendice B

Approfondimenti sulla SVD

Data una generica matrice (anche complessa) \mathbf{A} , con m righe ed n colonne, se σ è uno scalare positivo, e \mathbf{U} e \mathbf{V} sono vettori non nulli di m ed n elementi rispettivamente, allora

$$\mathbf{A}\mathbf{V} = \sigma\mathbf{U} \quad (\text{B.1})$$

e

$$\mathbf{A}^*\mathbf{U} = \sigma\mathbf{V} \quad (\text{B.2})$$

La decomposizione ai valori singolari (SVD), può essere vista come un metodo per trasformare le variabili correlate in un insieme di variabili non correlate che evidenzia meglio le varie relazioni presenti nella matrice di partenza. Questo permette di poter considerare solamente le informazioni più importanti contenute in essa [28]. I valori singolari della generica matrice \mathbf{A} sono dati dalle radici quadrate degli autovalori non nulli di $\mathbf{A}\mathbf{A}^T$. Tuttavia, calcolare in questo modo i valori singolari, non è solitamente stabile [29]. Un algoritmo comune ([30]) per la SVD è esposto nel seguito.

STEP 1 : riduzione di Householder alla forma bidiagonale

Input: m, n, \mathbf{A} . Dove \mathbf{A} è $m \times n$.

Output: $\mathbf{B}, \mathbf{U}, \mathbf{V}$.

\mathbf{B} è una matrice bidiagonale superiore, \mathbf{U} e \mathbf{V} sono i prodotti delle matrici di Householder.

$$\mathbf{A} = \mathbf{UBV}^T$$

1. $\mathbf{B} \leftarrow \mathbf{A}$
2. $\mathbf{U} = \mathbf{I}_{m \times n}$
3. $\mathbf{U} = \mathbf{I}_{n \times n}$
4. For $k=1, \dots, n$

(a) Determinare la matrice di Householder \mathbf{Q}_k con le seguenti proprietà:

- La moltiplicazione sinistra per \mathbf{Q}_k lascia inalterati le componenti $1, \dots, k-1$
-

$$\mathbf{Q}_k \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_{k-1,k} \\ b_{k,k} \\ b_{k+1,k} \\ \vdots \\ b_{m,k} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b_{k-1,k} \\ s \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (\text{B.3})$$

con

$$s = \pm \sqrt{\sum_{i=k}^m b_{i,k}^2} \quad (\text{B.4})$$

- (b) $\mathbf{B} \leftarrow \mathbf{Q}_k \mathbf{B}$
- (c) $\mathbf{U} \leftarrow \mathbf{U} \mathbf{Q}_k$
- (d) se $k \leq n-2$, determinare la matrice di Householder \mathbf{P}_{k+1} con la proprietà che:
 - la moltiplicazione destra per \mathbf{P}_{k+1} lascia inalterate le componenti $1, \dots, k$

•

$$\begin{bmatrix} 0 & \cdots & \mathbf{b}_{k,k} & \mathbf{b}_{k,k+1} & \mathbf{b}_{k,k+2} & \cdots & \mathbf{b}_{k,n} \end{bmatrix} \mathbf{P}_{k+1} = \begin{bmatrix} 0 & \cdots & 0 & \mathbf{b}_{k,k} & s & 0 & \cdots & 0 \end{bmatrix} \quad (\text{B.5})$$

con

$$s = \pm \sqrt{\sum_{j=k+1}^n b_{k,j}^2} \quad (\text{B.6})$$

$$(e) \mathbf{B} \leftarrow \mathbf{B}\mathbf{P}_{k+1}$$

$$(f) \mathbf{V} \leftarrow \mathbf{P}_{k+1}\mathbf{V}$$

STEP 2: algoritmo di Golub-Reinsch per la SVD.

Input: m, n, \mathbf{A} , dove \mathbf{A} è $m \times n$

Output: $\sigma, \mathbf{U}, \mathbf{V}$ tale che $\mathbf{A} = \mathbf{U}\sigma\mathbf{V}^T$ (decomposizione ai valori singolari)

1. Applicare lo step 1 per ottenere \mathbf{B}, \mathbf{U} e \mathbf{V}

2. Ripetere:

(a) Se per ogni $i = 1, \dots, n-1$, $|b_{i,i+1}| \leq \epsilon(|b_{i,i}| + |b_{i+1,i+1}|)$, imponi $b_{i,i+1} = 0$

(b) Determinare il più piccolo p e il più grande q tale che \mathbf{B} può essere a blocchi:

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_{1,1} & 0 & 0 \\ 0 & \mathbf{B}_{2,2} & 0 \\ 0 & 0 & \mathbf{B}_{3,3} \end{bmatrix} \quad (\text{B.7})$$

dove la matrice $\mathbf{B}_{1,1}$ ha p righe, la matrice $\mathbf{B}_{2,2}$ ha $n - p - q$ righe e la matrice $\mathbf{B}_{3,3}$ ha q righe. $\mathbf{B}_{3,3}$ è diagonale e $\mathbf{B}_{2,2}$ ha i valori non nulli sopra la diagonale.

(c) Se $q = n$, poni σ uguale alla porzione diagonale di \mathbf{B} e termina l'algoritmo.

(d) Se per $i = p+1, \dots, n-q-1$, $b_{i,j} = 0$, allora

- Applica le rotazioni di Givens finchè $b_{i,i+1} = 0$ e $\mathbf{B}_{2,2}$ è ancora una bidiagonale superiore.

Altrimenti

- Applicare l' algoritmo di Golub-Kahan a $n, \mathbf{B}, \mathbf{U}, \mathbf{V}, p, q$

ALGORITMO DI GOLUB-KAHAN

Input: $n, \mathbf{B}, \mathbf{Q}, \mathbf{P}, p, q$ dove \mathbf{B} è di dimensioni $n \times n$ bidiagonale superiore, \mathbf{Q} e \mathbf{P} hanno colonne ortogonali, e $\mathbf{A} = \mathbf{Q}\mathbf{B}\mathbf{P}^T$

Output: $\mathbf{B}, \mathbf{Q}, \mathbf{P}$

1. Definire \mathbf{C} come la sottomatrice inferiore destra di dimensioni 2×2 di $\mathbf{B}_{2,2}^T \mathbf{B}_{2,2}$
2. ricavare gli autovalori λ_1, λ_2 di \mathbf{C} . Definire μ come l'autovalore appena calcolato più vicino a $c_{2,2}$
3. $k = p + 1, \alpha = b_{k,k}^2 - \mu, \beta = b_{k,k} b_{k,k+1}$
4. Per $k = p + 1, \dots, n - q - 1$

- (a) Determinare $c = \cos(\theta)$ e $s = \sin(\theta)$ con la proprietà che:

$$\begin{bmatrix} \alpha & \beta \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} = \begin{bmatrix} \sqrt{\alpha^2 + \beta^2} & 0 \end{bmatrix} \quad (\text{B.8})$$

- (b) $\mathbf{B} \leftarrow \mathbf{B}\mathbf{R}_{k,k+1}(c, s)$ dove $\mathbf{R}_{k,k+1}(c, s)$ è la matrice di rotazione di Givens che agisce sulla colonna k e $k + 1$ durante la moltiplicazione a destra;
- (c) $\alpha = b_{k,k}, \beta = b_{k+1,k}$
- (d) Determinare $c = \cos(\theta)$ e $s = \sin(\theta)$ con la proprietà che

$$\begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \sqrt{\alpha^2 + \beta^2} \\ 0 \end{bmatrix} \quad (\text{B.9})$$

- (e) $\mathbf{B} \leftarrow \mathbf{R}_{k,k+1}(c, s)\mathbf{B}$, dove $\mathbf{R}_{k,k+1}(c, s)$ è la matrice di rotazione di Givens che agisce sulla riga k e $k + 1$ nella moltiplicazione a sinistra;
- (f) $\mathbf{B} \leftarrow \mathbf{Q}\mathbf{R}_{k,k+1}(c, s)$
- (g) se $k \leq n - q - \alpha = b_{k,k+1}$, allora $\beta = b_{k,k+2}$

Bibliografia

- [1] Brian McLaughlan. *Vibration measurement and control*, 2006.
- [2] Jwusheng Hu Jyh-Feng Lin. Feedforward active noise controller design in ducts without independent noise source measurements. *Control Systems Technology*, 2000.
- [3] Wittenmark Astrom. *Adaptive Control*. Addison-Wesley, 2 edition, 1995.
- [4] Morlacchi. *Sviluppo di un algoritmo per l'identificazione dei parametri e il controllo adattativo di un sistema meccanico*. 2012.
- [5] Jing SUn A. Ioannou. *Robust Adaptive Control*. Prentice-Hall, 1996.
- [6] E.D. Sontag. *Mathematical Control Theory*. Springer, 2 edition, 1998.
- [7] M'Saad Karimi Landau, Lozano. *Adaptive Control*. Springer, 2010.
- [8] Lee C.C. Fuzzy logic in control systems: Fuzzy logic controller - part 1. *IEEE Transactions on Systems, Man, and cybernetics*, Vol. 20, March/April 1990.
- [9] A. Zaknich. *Principles of Adaptive Filters and Self-learning Systems*. Springer, 2005.
- [10] Bittanti. *Identificazione dei modelli e sistemi adattativi*.
- [11] K. J. HuntD. SbarbaroR. ŻbikowskiP. J. Gawthrop. Neural networks for control systems: a survey autore. *Automatica (Journal of IFAC) Volume 28 Issue 6, Nov. 1992 Pages 1083 - 1112*, 1992.

- [12] Lazzerini. *Introduzione alle reti neurali*.
- [13] Howard B. Demuth Mark Hudson Beale, Martin T. Hagan. *Neural Network Toolbox*.
- [14] Gobbi M. *Appunti di Progettazione Ottima*.
- [15] J.C. Surkan, A.J. ; Singleton. Neural networks for bond rating improved by multiple hidden layers. *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, 1990.
- [16] Simon Haykin. *Neural Networks: A Comprehensive Foundation 2nd*. Prentice Hall, 1998.
- [17] Wang X. Xu, J. *A Structural Identification Method Based on Recurrent Neural Network and Auto-Regressive and Moving Average Model*. 2012.
- [18] Steve Dini and Mark Serrano. *Combining Q-Learning with Artificial Neural Networks in an Adaptive Light Seeking Robot*. 2012.
- [19] A.A.Tsonis J.B.Elsner. *Singular Spectrum Analysis: a new tool in time series analysis*. Plenum Publishing Corporation, 1996.
- [20] B. De Moor M.Moonen. *SVD and Signal Processing III: Algorithms, Architectures and Applications*. Elsevier Science, 1995.
- [21] Carla Guerrini. *Information retrieval e algebra lineare*.
- [22] Mario Rosario Guarracino Enzo Imposimato. Riconoscimento e identificazione automatica dei volti. Technical report, Consiglio Nazionale delle Ricerche Istituto di Calcolo e Reti ad Alte Prestazioni, 2009.
- [23] R. Vandebriel N. Mastronardi, M. Van Barel. A note on the recursive calculation of dominant singular subspaces. *Numerical Algorithms* 38, 2005.
- [24] R. Vandebriel N. Mastronardi, M. Van Barel. A fast algorithm for the recursive calculation of dominant singular subspaces. *Journal of Computational and Applied Mathematics, Volume 218*, 2008.

- [25] G.W.Stewart. An updating algorithm for subspace tracking. *IEEE Transactions on Signal Processing*, vol. 40, no.6, 1992.
- [26] D.Simon. *Optimal State Estimation: Kalman, Hinf, and Nonlinear Approaches*. Wiley, 2006.
- [27] Braghin. *Dispense sistemi meccatronici*, 2013.
- [28] Kirk Baker. *Singular Value Decomposition Tutorial*, March 2005.
- [29] Greg Fasshauer. *Computational Mathematics*.
- [30] Inderjit S. Dhillon Alan Kaylor Cline. *computation of the Singular Value Decomposition*.

Ringraziamenti

Questo lavoro è stato realizzato sotto la supervisione dell'ingegnere Gabriele Cazulani e dell'ingegnere Francesco Ripamonti, del dipartimento di meccanica del Politecnico di Milano.