



**Politecnico
di Torino**

DIPARTIMENTO DI INGEGNERIA MECCANICA E AEROSPAZIALE
Corso di Laurea Magistrale in Ingegneria Aerospaziale

TESI DI LAUREA MAGISTRALE

Ottimizzazione aeroelastica mediante reti neurali

Relatori: Matteo Filippi
Marco Petrolo

Candidato: Davide Giofrè

Anno Accademico 2023/2024

Indice

1	Introduzione	1
1.1	Origini e motivazioni storiche dell'aeroelasticità nell'industria aeronautica	2
1.2	Intelligenza artificiale e machine learning	4
1.3	Contenuto dell'elaborato	5
2	Formulazione strutturale	7
2.1	Teorie classiche	9
2.1.1	Trave di Eulero Bernoulli - EBBT	9
2.1.2	Trave di Timoshenko - TBT	10
2.2	Carrera Unified Formulation - CUF	10
2.2.1	Polinomio di Taylor a ordine N	11
2.2.2	Polinomi di Lagrange	12
2.3	Formulazione FEM mediante CUF	15
2.4	Equazioni di governo	17
2.4.1	Analisi statica - SOL 101	18
2.4.2	Analisi dinamica (vibrazioni libere) - SOL 103	18
3	Formulazione aerodinamica	20
3.1	Metodi ai pannelli	20
3.2	Vortex Lattice Method - VLM	20
3.3	Doublet Lattice Method - DLM	24
4	Formulazione aeroelastica	27
4.1	Il metodo delle spline	27
4.2	Pendenza nei <i>Control Points</i>	29
4.3	Spostamenti nei <i>Load Points</i>	30
4.4	Equazioni di governo per il flutter	30
4.4.1	Matrici generalizzate	31
5	Cenni di reti neurali artificiali	33
5.1	Il problema della regressione	34
5.1.1	Regressione lineare	34
5.1.2	Regressione logistica	36

5.2	Assemblaggio di una rete neurale	39
5.2.1	Notazione matematica	40
5.3	Algoritmo di forward & backward propagation	41
5.4	Funzioni di attivazione	42
5.5	Underfitting & overfitting	45
5.6	Altri tipi di rete	46
5.6.1	Convolutional Neural Networks - CNNs	46
5.6.2	Physics-Informed Neural Networks - PINNs	49
6	Risultati preliminari	51
6.1	Analisi statica	51
6.2	Analisi dinamica	56
6.3	Analisi aeroelastica dinamica: flutter	58
6.4	Analisi aeroelastica statica	62
6.5	Confronto con Femap	64
6.5.1	Analisi statica	65
6.5.2	Analisi dinamica	67
6.5.3	Analisi aeroelastica dinamica: flutter	69
6.5.4	Analisi aeroelastica statica	73
6.6	Analisi di convergenza	75
6.6.1	Convergenza mesh aerodinamica	75
6.6.2	Convergenza mesh strutturale	76
7	Analisi sul velivolo finale	78
7.1	Ala di Goland	79
7.2	Analisi dell'ala	82
7.3	Analisi dinamica e di flutter del velivolo	87
7.4	Velivolo con i motori	92
7.5	Effetto della posizione dei motori sulla condizione di flutter	98
7.5.1	Effetto della posizione del motore lungo la corda	98
7.5.2	Effetto della posizione del motore lungo l'apertura alare	99
8	Analisi di machine learning	102
8.1	Creazione del dataset	102
8.2	Costruzione del modello	103
8.2.1	Numero di layers della DNN	103
8.3	Ottimizzazione delle dimensioni dei set e del numero di epoche di training	109
8.3.1	Scelta <i>manuale</i> degli input	109
8.3.2	Valutazione degli errori e scelta del dataset migliore	112
8.4	Analisi sul modello definitivo	114
9	Conclusioni	118
	Bibliografia	122

Elenco delle figure

1.1	Triangolo di Collar. A aerodinamica, E elasticità, D dinamica.	1
1.2	Primo tentativo del Great Aerodrome in decollo da una piattaforma sul fiume Potomac, il 7 ottobre 1903.	3
1.3	Soluzione di <i>mass-balance</i> per il flutter dell'alettone sull'idrovolante Van Berkel.	3
2.1	Schema di una trave con il sistema di riferimento utilizzato.	7
2.2	Sezione trasversale in coordinate reali e normalizzate per LE4.	12
2.3	Sezione trasversale in coordinate reali e normalizzate per LE9.	13
2.4	Trave discretizzata in elementi B2	15
2.5	Punto generico lungo l'asse della trave	16
3.1	Distribuzione dei pannelli e dei filamenti vorticosi su un'ala a freccia	21
3.2	Generico pannello utilizzato nel VLM. Sono mostrati la posizione del filamento vorticoso, il load point e il control point.	21
3.3	Velocità indotta in un punto C da un filamento vorticoso	22
3.4	Rappresentazione del pannello i -esimo nella sua forma deformata, reale, e nella forma non deformata presa come riferimento	23
5.1	Neurone biologico confrontato con un neurone artificiale. (a) neurone biologico; (b) neurone artificiale; (c) sinapsi biologica; (d) sinapsi artificiale	33
5.2	Schema del modello lineare con un input (una variabile indipendente).	34
5.3	Schema del modello lineare con più input	36
5.4	Confronto tra una funzione a gradino e la curva sigmoide	37
5.5	Loss function per i dati delle due classi.	38
5.6	Schema del neurone che esegue la regressione logistica a partire da d input.	39
5.7	Schema di una rete neurale profonda	39
5.8	DNN ad un singolo layer nascosto	42
5.9	ReLU, Leaky ReLU e Swish	43
5.10	Previsioni di tre diversi modelli di rete. I pallini neri sono dati di training, i quadrati rossi sono dati di test	45
5.11	Errori del modello al variare della sua complessità	45
5.12	Campi ricettivi nella corteccia visiva del cervello.	46

5.13	Architettura di una CNN impiegata per monitorare lo stato di salute della struttura di un velivolo.	47
5.14	Operazione di convoluzione tra la matrice di input e il filtro.	47
5.15	Convoluzione con la tecnica di zero-padding applicata all'input.	48
5.16	Operazione di max pooling con un filtro 2×2	48
5.17	Architettura di una PINN	49
6.1	Trave analizzata	51
6.2	Elemento L9	52
6.3	Elemento L4	53
6.4	Trave deformata. In seguito al carico applicato la trave si inflette. La freccia massima si riscontra nell'estremità libera, mentre all'incastro lo spostamento è nullo.	55
6.5	Distribuzione dello sforzo σ_y sulla trave. Dalla colorazione si può confermare l'andamento a farfalla dello stress.	55
6.6	Rappresentazione grafica delle frequenze calcolate con i vari modelli	56
6.11	Modi vibrazionali della trave.	58
6.12	Piastra analizzata	58
6.13	Sezione trasversale della piastra schematizzata con 4 elementi L9.	59
6.19	Risultati dell'analisi di flutter per i primi 5 modi vibrazionali.	62
6.20	Modello della trave su FEMAP	65
6.21	Deformata della trave su FEMAP	65
6.22	Andamento dello stress σ_y sulla superficie superiore della trave.	66
6.23	Ingrandimento della trave in cui vengono mostrati, in rosso, gli elementi 5 e 6 in cui viene valutato lo stress σ_y	66
6.28	Modi vibrazionali valutati con FEMAP	69
6.33	Modi vibrazionali ottenuti con FEMAP	71
6.34	In figura si osserva anche il sistema di riferimento aerodinamico, in rosso, posizionato al bordo di attacco.	71
6.35	Le curve ottenute rispecchiano quanto trovato con il codice MUL2.	72
6.36	Soluzioni dei due codici a confronto	74
6.37	Il cerchio pieno indica la mesh utilizzata per le analisi aeroelastiche sopra riportate, ossia 30×8	75
6.38	Il cerchio pieno indica la mesh utilizzata per le analisi aeroelastiche sopra riportate, ossia 40×10	77
7.1	Schema del velivolo in 3D	78
7.2	Modello FEM dell'ala di Goland.	80
7.3	Andamento della velocità di flutter per varie posizioni dei motori lungo la corda.	81
7.4	Andamento della velocità di flutter per varie posizioni dei motori lungo l'apertura alare.	82
7.5	(a) Ala in parete sottile	83
7.6	(b) Ala in parete sottile rinforzata	83
7.10	Frequenze e smorzamenti dei primi 6 modi dell'ala incastrata	85

7.15	Modi vibrazionali dell'ala non incastrata.	86
7.19	Modi vibrazionali del velivolo incastrato.	88
7.20	Frequenze e smorzamenti dei primi 7 modi del velivolo incastrato.	89
7.27	Modi vibrazionali del velivolo libero	91
7.28	Frequenze e smorzamenti di alcuni modi del velivolo libero.	92
7.29	Modello del motore e del pilone associato.	93
7.30	Scomposizione dei materiali utilizzati nel GE CF6.	94
7.31	Modello definitivo del velivolo.	95
7.39	Frequenze e smorzamenti di alcuni modi del velivolo libero con i motori.	97
7.40	Andamento della velocità di flutter per varie posizioni dei motori lungo la corda.	99
7.41	Andamento della velocità di flutter per varie posizioni dei motori lungo l'apertura alare.	100
8.1	Errori di train, validation e di test per diverse complessità del modello, con un random state pari a 23	104
8.2	Errori di train, validation e di test per diverse complessità del modello, e per diversi random states.	105
8.3	Architettura finale della rete neurale.	106
8.7	Confronto tra le due velocità al variare dell'input x_e e per diverse posizioni y_e	108
8.11	Confronto tra le due velocità al variare dell'input y_e e per diverse posizioni x_e	109
8.12	Velocità note e predette a confronto dei vari test size per $x_e = 0$ e lungo l'apertura alare	111
8.13	Velocità note e predette a confronto dei vari test size per $y_e = 42\%$ e lungo la corda.	112
8.14	Confronto degli errori su ogni dato nei 4 test size considerati.	113
8.19	Confronto tra le due velocità al variare dell'input y_e e per diverse posizioni x_e	116
8.24	Confronto tra le due velocità al variare dell'input x_e e per diversi valori di y_e	117

Elenco delle tabelle

2.1	Funzioni di espansione nel caso di polinomi di Taylor fino al quarto ordine	11
2.2	Coordinate normalizzate dei quattro punti della sezione	13
2.3	Coordinate normalizzate dei nove punti della sezione	14
6.1	Spostamento all'estremo libero u_z in $10^{-5} m$	53
6.2	Gradi di libertà della trave discretizzata secondo le varie teorie di approssimazione.	54
6.3	Stress σ_y in Pa all'estremo vincolato.	54
6.4	Frequenze dei vari modi di vibrare in Hertz	56
6.5	Confronto tra le frequenze, in Hertz, con i risultati bibliografici	59
6.6	Confronto tra i risultati delle analisi aeroelastiche con quelli bibliografici.	62
6.7	Risultati relativi al modello TE4	63
6.8	Risultati relativi al modello 4LE9	64
6.9	Confronto dei risultati dell'analisi statica tra MUL2, utilizzando 10 elementi B4, e FEMAP.	67
6.10	Confronto tra le frequenze, in Hertz, ottenute con MUL2 e FEMAP.	67
6.11	Confronto tra le frequenze, in Hertz, ottenute con MUL2 e FEMAP.	69
6.12	Tabella riassuntiva dei risultati	73
6.13	Risultati ottenuti con il codice FEMAP.	74
6.14	Risultati dell'analisi	75
6.15	Risultati dell'analisi	76
7.1	Dimensioni del velivolo considerato.	78
7.2	Caratteristiche del materiale utilizzato.	79
7.3	Dati relativi all'ala di Goland	79
7.4	Frequenze e modi di vibrazione dei primi 6 modi dell'ala incastrata.	83
7.5	Frequenze e modi di vibrazione dell'ala libera	85
7.6	Confronto delle frequenze e dei modi di vibrare tra il velivolo incastrato con quelli dell'ala incastrata della Tabella 7.4.	87
7.7	Confronto della condizione di flutter tra il velivolo e l'ala incastrati.	89
7.8	Frequenze e modi vibrazionali dell'ala libera.	90
7.9	Confronto della condizione di flutter tra il velivolo e l'ala libera.	92
7.10	Dimensioni del motore e del pilone.	93
7.11	Proprietà medie dei materiali utilizzati nel motore considerato.	94

Elenco delle tabelle

7.12	Frequenze e modi vibrazionali del velivolo libero con i motori.	95
7.13	Confronto della condizione di flutter tra il velivolo libero con i motori e senza di essi.	98
7.14	Risultati relativi alla velocità di flutter per varie posizioni dei motori lungo la corda.	98
7.15	Risultati relativi alla velocità di flutter per varie posizioni dei motori lungo l'apertura alare.	100
8.1	Suddivisione del dataset nei tre set utili.	103
8.2	Numero di parametri che la rete allena durante il training.	106
8.3	Parametri per monitorare gli errori dei vari test size a confronto.	113
8.4	Parametri definitivi della rete neurale.	114

Capitolo 1

Introduzione

L'aeroelasticità è la materia che studia i fenomeni aeroelatici che nascono dalla mutua interazione tra un fluido e una struttura deformabile. Nella forma più generale si può dire che è l'insieme di tre discipline che interagiscono tra loro: l'aerodinamica, l'elasticità delle strutture e la dinamica. Un'illustrazione del problema è fornita dal Triangolo di Collar riportato nella Figura 1.1.

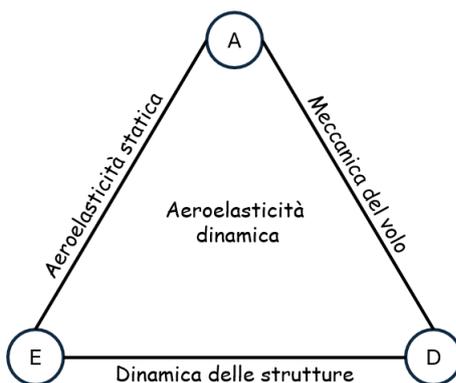


Figura 1.1: Triangolo di Collar. *A* aerodinamica, *E* elasticità, *D* dinamica.

In un problema aeroelastico l'aerodinamica si occupa di definire il carico agente sulla struttura, l'elasticità di definire lo stato tensionale e di deformazione, e la dinamica di definire gli effetti inerziali e di smorzamento. Dunque, a seconda del tipo di fenomeno da studiare vengono coinvolte due o tre di queste discipline.

- Per problemi di **meccanica del volo** si utilizzano le teorie dell'*aerodinamica* e della *dinamica*. Vengono descritti fenomeni come:
 - *risposta alla raffica*
 - *stabilità e controllo del velivolo*

- Per problemi di **dinamica delle strutture** si utilizzano le teorie dell'*elasticità* e della *dinamica*. In questo caso vengono descritti fenomeni come:
 - *frequenze libere , forzate e analisi modali*
 - *vibrazione meccanica*
 - *impatti*
- Per problemi di **aeroelasticità statica** si utilizzano le teorie dell'*aerodinamica* e dell'*elasticità delle strutture*. Vengono descritti due fenomeni in particolare:
 - *divergenza*: su un velivolo è dovuta all'azione della portanza che deforma una superficie portante in modo da incrementarne la portanza e di conseguenza farla deformare ulteriormente, fino alla distruzione.
 - *inversione dei comandi*: si verifica solo sulle superfici portanti dotate di superfici mobili per il controllo del velivolo. Se la superficie portante si deforma eccessivamente può succedere che la superficie mobile produca un effetto opposto a quanto ci si aspetti.
- Per problemi di **aeroelasticità dinamica** si utilizzano tutte e tre le teorie per studiare fenomeni come:
 - *flutter*: instabilità dinamica che si verifica a causa dell'interazione tra le forze aerodinamiche e i modi propri vibrazionali della struttura. Il punto di flutter è la velocità del velivolo per il quale lo smorzamento del sistema è nullo. In questa situazione la struttura è animata da un moto armonico non smorzato che, per velocità superiori, porta a delle oscillazioni auto indotte che possono causare il cedimento della struttura.
 - *buffeting*: instabilità associata alla turbolenza nella corrente fluida che impatta su una superficie. Si manifesta con oscillazioni irregolari che non aumentano nel tempo come per il flutter poiché non sono auto-eccitate, ma dipendono da fattori esterni.

L'aeroelasticità non è di interesse esclusivo al settore aeronautico e aerospaziale, ma tali fenomeni riguardano una qualunque struttura deformabile che interagisca con le forze generate da una corrente fluida. Si ritrovano infatti problemi aeroelastici anche nelle costruzioni civili come ponti o grattacieli.

1.1 Origini e motivazioni storiche dell'aeroelasticità nell'industria aeronautica

Nel 1903 i fratelli Wright compirono quello che viene ricordato come il primo volo controllato, sostenuto e prolungato di un velivolo propulso e guidato da un pilota a bordo. Prima di allora furono stati condotti vari tentativi, tutti senza successo. Vengono ricordati in particolare i due voli del Great Aerodrome (Figura 1.2), il velivolo propulso costruito dal pioniere dell'aviazione Samuel P. Langley sempre nel 1903.

1.1. Origini e motivazioni storiche dell'aeroelasticità nell'industria aeronautica

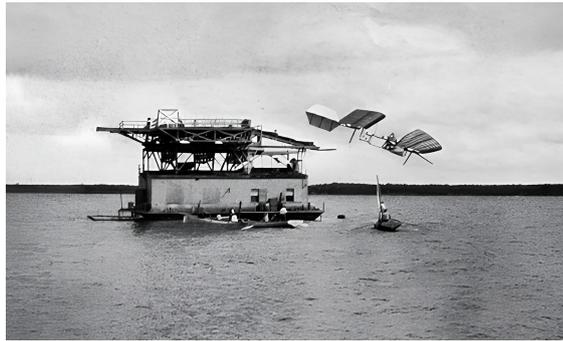


Figura 1.2: Primo tentativo del Great Aerodrome in decollo da una piattaforma sul fiume Potomac, il 7 ottobre 1903.

Entrambi furono fallimentari: la causa del primo fu assegnata alla presenza di un cavo in cui il velivolo si impigliò durante il decollo facendolo precipitare nel fiume Potomac. Per il secondo tentativo invece, che causò il cedimento dell'ala posteriore e della coda, le cause rimasero incerte ma si suppose che una maggiore rigidità all'estremità alare avrebbe potuto evitare la divergenza torsionale dell'ala [1].

Negli anni a seguire, durante la Prima Guerra Mondiale, si verificarono diversi incidenti aerei. Tra i più ricordati ci sono quelli sul velivolo Handely Page Type O/400 della RAF, Albatros D-III e il Fokker D-VIII tedeschi. Il primo fra questi è riconosciuto come il primo caso documentato di flutter; l'incidente fu causato da forti oscillazioni sulla fusoliera e la coda. L'ingegnere britannico Frederick W. Lanchester notò che le oscillazioni erano auto-indotte e non erano dovute ad un problema di risonanza, e che potevano essere attenuate aumentando la rigidità torsionale della coda [1].

A partire dagli anni '20, mentre nel mondo si registrarono sempre più incidenti causati dal flutter, soprattutto sulle superfici di controllo, cominciarono i primi studi teorici in Olanda, al *National Physical Laboratory* (NPL), condotti da Von Baumhauer e Koning, in cui dimostrarono che le oscillazioni potevano essere ridotte, o addirittura eliminate, con una massa concentrata posizionata come in Figura 1.3, [2].

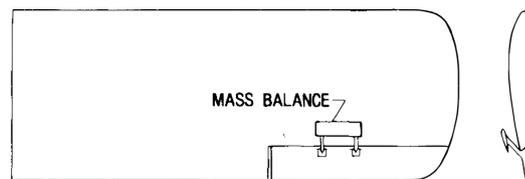


Figura 1.3: Soluzione di *mass-balance* per il flutter dell'alettone sull'idrovolante Van Berkel.

Nel 1926 Reissner pubblicò una teoria per descrivere la divergenza dell'ala e nel 1930 venne coniato, per la prima volta, il termine *aeroelasticità* dagli ingegneri Harold R. Cox e Alfred Pugsley che studiarono il problema dell'inversione dei comandi, fornendo requisiti precisi per la rigidità necessaria dell'ala.

Nel 1934, negli Stati Uniti, Theodore Theodorsen introdusse un metodo rivoluzionario per il calcolo del flutter. Con le ipotesi di flusso potenziale e validità della condizione di Kutta, studiò il problema di una placca piana oscillante sottoposta a traslazione, torsione e azionamento dell'alettone introducendo la funzione complessa $C(\kappa)$ per descrivere gli sfasamenti tra le forze aerodinamiche e il movimento della placca piana, [3]. Tale soluzione giocò un ruolo importante all'aviazione americana.

A partire da allora ci furono svariati sviluppi teorici tra cui quelli di Küssner, Von Karman e Sears riguardanti la riposta di un profilo alare investito da una raffica a gradino. Furono condotti anche i primi test in volo seguendo l'idea von Schlippe che portarono a misure empiriche della velocità di flutter. Questi furono successivamente accantonati poiché si riteneva troppo pericoloso condurre test di flutter in volo. Vennero ripresi più avanti, verso la fine degli anni '40, quando furono disponibili strumentazioni avioniche più sofisticate e ci fu una migliore comprensione teorica del problema.

Con la Seconda Guerra Mondiale la progettazione degli aerei cambiò rapidamente: maggiori velocità di volo, lunghi range, diversi aspect ratios, parti strutturali in materiale diverso (non si utilizzò più il legno ma l'acciaio) e l'introduzione di ali a freccia, portarono ad una comprensione più difficile dei problemi aeroelastici. Furono fondamentali in questa fase anche i test in galleria del vento.

Nel 1947 avvenne il primo volo supersonico, questo portò ad un maggior interesse nella ricerca delle condizioni di flutter ad alte velocità. I progressi cominciarono già qualche anno prima dove scienziati da tutta Europa e dagli Stati Uniti portarono avanti gli studi preliminari condotti da Possio negli anni '30. Il maggior contributo fu fornito dal Professor Holt Ashley nel 1956 quando pubblicò la *Piston Theory* [4], in cui mostrò come è possibile legare la pressione generata da un corpo in movimento, con la componente normale alla superficie del corpo della velocità, in modo analogo ad un fluido mosso da un pistone in uno spazio monodimensionale.

Negli anni '60 cominciarono ad essere prodotti i primi codici di calcolo ad elementi finiti che potevano essere utilizzati per casi semplici. Mentre la ricerca teorica continuò vennero perfezionati i codici di calcolo e nel 1968 la NASA rilasciò il proprio software open source NASTRAN, acronimo di *NASA STRucture ANalysis*.

1.2 Intelligenza artificiale e machine learning

Con "Intelligenza Artificiale" si intende la capacità di un sistema informatico di simulare il funzionamento di un cervello umano per eseguire dei calcoli o conseguire obiettivi, come ad esempio la previsione e analisi dei dati, il riconoscimento di immagini o di pattern, l'automazione di processi lavorativi nelle industrie e molto altro. Alcuni di questi sistemi non vengono creati con linee di codice, come per gli algoritmi tradizionali, per risolvere un certo problema, ma perseguono il loro obiettivo imparando da esperienze passate o esempi. Ciò avviene attraverso un *addestramento* del sistema con un set di dati già noti. La branca dell'intelligenza artificiale che si occupa di sistemi di apprendimento di dati viene definita "Machine Learning".

Molti esperti del settore definiscono la nascita dell'intelligenza artificiale nel 1956, [5], quando dieci fra i migliori informatici al mondo parteciparono ad un congresso sponsorizzato dalla

IBM per discutere dei loro lavori riguardo le reti neurali artificiali e nuovi linguaggi di programmazione. Si discusse anche di convogliare i loro studi per creare un computer "intelligente". Fu coniato per la prima volta il termine *intelligenza artificiale*.

I primi lavori furono puramente accademici ed erano pensati per poter giocare a dama [6] o a scacchi [7]. Erano in grado di battere molti giocatori umani, tuttavia non portò ad un grande interesse verso questi sistemi. Fu durante gli anni '70 con il sistema PROSPECTOR, che aiutò i geologi a trovare un sito ricco di molibdeno dal valore totale di 100 milioni di dollari, e con il computer scacchistico Deep Blue, che fu in grado di sconfiggere il Grande Maestro e campione del mondo Garry Kasparov in una partita ben pubblicizzata, che l'interesse verso l'intelligenza artificiale prese piede.

Al giorno d'oggi l'intelligenza artificiale è diventata di utilizzo quotidiano avendo un importanza cruciale in diversi settori grazie alla sua capacità di analizzare dati, riconoscere patterns e automatizzare processi. Nel settore industriale l'IA è capace di ottimizzare la produzione riducendo i costi e migliorando dunque l'efficienza del processo produttivo. Nel campo della sanità vengono utilizzati sistemi ad intelligenza artificiale per diagnosticare malattie e accelerare la ricerca medica.

Nella ricerca, l'IA accelera l'analisi dei dati e facilita la scoperta di nuove soluzioni. In ambito scientifico, ad esempio, l'IA consente di simulare esperimenti, analizzare grandi quantità di dati o, ad esempio, identificare nuove molecole per farmaci.

1.3 Contenuto dell'elaborato

Lo studio presentato in questo elaborato ha come obiettivo quello di realizzare una rete neurale capace di prevedere il comportamento al flutter di un velivolo per diverse disposizioni di motori lungo l'ala. Per fare ciò verranno prese in considerazione varie configurazioni di velivolo in cui i motori saranno posizionati in diverse stazioni lungo l'apertura alare, lungo la corda. Per ciascuna di esse verrà condotta un'analisi aeroelastica dinamica al fine di ottenere il punto di flutter e costruire così un dataset, sufficientemente popolato, per eseguire il *training* della rete neurale. La rete, una volta addestrata, dovrà saper determinare correttamente le proprietà aeroelastiche dinamiche anche per configurazioni dei motori per le quali non è stata addestrata.

Per creare il dataset contenente le proprietà di flutter dovranno essere condotte diverse analisi strutturali; si utilizzerà a tal fine il codice MUL2, sviluppato dall'omonimo gruppo di ricerca del Politecnico di Torino, per la realizzazione del modello ad elementi finiti, per le analisi statiche e dinamiche, e il codice commerciale NASTRAN per eseguire le analisi aeroelastiche statiche e dinamiche.

La rete neurale invece, verrà implementata con il linguaggio di programmazione PYTHON mediante l'utilizzo delle librerie TENSORFLOW e KERAS che rendono la creazione della rete semplice, veloce ed affidabile.

Prima di presentare lo studio condotto verranno illustrate le teorie di calcolo utilizzate. Nel Capitolo 2 si discuterà della teoria strutturale utilizzata, cioè la *Carrera Unified Formulation*. Il Capitolo 3 è dedicato invece ai modelli aerodinamici utilizzati, cioè il *Vortex Lattice Method*

e il *Doublet Lattice Method*. Nel Capitolo 4 si descrive il metodo utilizzato per fare comunicare il modello strutturale con il modello aerodinamico, essenziale per un'analisi aeroelastica, cioè il metodo *Infinite Plate Spline*, ma vengono descritte anche le equazioni di governo. Si precisa che questi modelli nominati sono implementati nel codice di calcolo FEM MUL2. Il Capitolo 5 fornisce una breve introduzione alle reti neurali, alle principali tipologie impiegate al giorno d'oggi, ma fornisce anche una panoramica riguardo la formulazione matematica di base per il calcolo. I capitoli successivi sono dedicati invece al lavoro svolto, a partire dal Capitolo 6 dove si analizza il funzionamento del codice mediante la conferma di risultati bibliografici riguardanti analisi statiche, dinamiche e aeroelastiche su strutture semplici. Nel Capitolo 7 si costruisce e si studia il modello del velivolo che sarà utilizzato per le analisi di flutter di interesse. Infine, nel Capitolo 8 verranno condotte le analisi di machine learning e discussi i risultati finali ottenuti.

Capitolo 2

Formulazione strutturale

In questo capitolo verranno descritti i modelli strutturali utilizzati per le analisi nei Capitoli 6 e 7. In particolare si descriveranno le teorie classiche di Eulero-Bernoulli e Timoshenko, ma anche della CUF basata su polinomi di Taylor a ordine N o di Lagrange.

Prima di procedere con la descrizione dei vari modelli si definiscono delle variabili che saranno utilizzate in seguito. Per fare ciò si fa riferimento alla Figura 2.1.

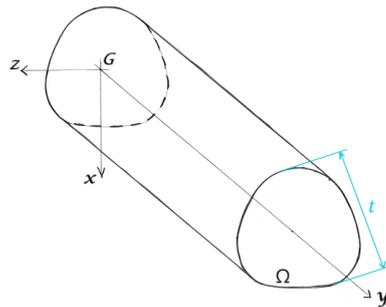


Figura 2.1: Schema di una trave con il sistema di riferimento utilizzato.

Rispetto al sistema di riferimento in Figura 2.1 si definiscono il vettore degli spostamenti nelle tre direzioni x, y, z

$$\mathbf{u} = \{u_x \quad u_y \quad u_z\}^T,$$

il vettore delle deformazioni

$$\boldsymbol{\varepsilon} = \{\varepsilon_{xx} \quad \varepsilon_{yy} \quad \varepsilon_{zz} \quad \gamma_{xz} \quad \gamma_{yz} \quad \gamma_{xy}\}^T$$

e il vettore degli sforzi

$$\boldsymbol{\sigma} = \{\sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \tau_{xz} \quad \tau_{yz} \quad \tau_{xy}\}^T.$$

Le tre grandezze non sono indipendenti le une dalle altre, ma è possibile esprimere un legame tra loro mediante l'operatore matriciale \mathbf{b} e la matrice dei coefficienti \mathbf{C} . Infatti, se si suppone di avere deformazioni non eccessive che porterebbero ad avere delle non linearità geometriche, e

dunque a dover introdurre termini di ordine superiore nelle equazioni, le componenti del vettore delle deformazioni ε si esprimono come

$$\begin{aligned}\varepsilon_{xx} &= \frac{\partial u_x}{\partial x}, & \gamma_{xz} &= \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \\ \varepsilon_{yy} &= \frac{\partial u_y}{\partial y}, & \gamma_{yz} &= \frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y} \\ \varepsilon_{zz} &= \frac{\partial u_z}{\partial z}, & \gamma_{xy} &= \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}\end{aligned}\tag{2.1}$$

Le equazioni (2.1) possono essere raggruppate nell'equazione matriciale (2.2).

$$\varepsilon = \mathbf{b} \cdot \mathbf{u}\tag{2.2}$$

dove

$$\mathbf{b} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{bmatrix}$$

Il legame tra tensioni e deformazioni si può esprimere attraverso la matrice \mathbf{C} :

$$\boldsymbol{\sigma} = \mathbf{C} \cdot \boldsymbol{\varepsilon}\tag{2.3}$$

in cui, nel caso di materiale isotropo,

$$\mathbf{C} = \begin{bmatrix} 2G + \lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2G + \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & 2G + \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & G & 0 & 0 \\ 0 & 0 & 0 & 0 & G & 0 \\ 0 & 0 & 0 & 0 & 0 & G \end{bmatrix}$$

e

$$G = \frac{E}{2(1 + \nu)}$$

$$\lambda = \frac{2E}{(1 + \nu)(1 - 2\nu)}$$

dove E è il modulo di Young, G è il modulo di taglio e ν è il modulo di Poisson.

2.1 Teorie classiche

2.1.1 Trave di Eulero Bernoulli - EBBT

La teoria della trave di Eulero Bernoulli si basa su tre ipotesi fondamentali:

1. la sezione trasversale è rigida nel suo piano; le deformazioni che la trave subisce dipendono unicamente dalla posizione lungo l'asse della trave. Riferendosi alla Figura 2.1 , sulla base di questa prima ipotesi, si può scrivere che

$$\begin{cases} \varepsilon_{xx} = \frac{\partial u_x}{\partial x} = 0 \\ \varepsilon_{zz} = \frac{\partial u_z}{\partial z} = 0 \\ \gamma_{xz} = \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} = 0 \end{cases} \quad (2.4)$$

da cui si può quindi scrivere che

$$\begin{cases} u_x(x, y, z) = u_{x1}(y) \\ u_z(x, y, z) = u_{z1}(y) \end{cases} \quad (2.5)$$

2. La sezione trasversale ruota attorno ad una superficie neutra rimanendo piana. Questo porta ad avere uno spostamento u_y planare nella sezione:

$$u_y(x, y, z) = u_{y1}(y) + \phi_z(y)x + \phi_x(y)z \quad (2.6)$$

dove ϕ_z e ϕ_x sono le rotazioni attorno agli assi z e x rispettivamente.

3. La sezione trasversale rimane perpendicolare alla superficie neutra durante la deformazione. Ciò significa che non possono avvenire deformazioni di taglio, per cui

$$\begin{cases} \gamma_{yz} = 0 \\ \gamma_{yx} = 0 \end{cases} \quad (2.7)$$

Combinando le equazioni (2.4), (2.6) e (2.7) si possono esprimere le rotazioni come le derivate degli spostamenti:

$$\begin{cases} \phi_z = -\frac{\partial u_{x1}}{\partial y} = -u_{x1,y} \\ \phi_x = -\frac{\partial u_{z1}}{\partial y} = -u_{z1,y} \end{cases} \quad (2.8)$$

Si può dunque descrivere il campo di spostamento della trave di Eulero-Bernoulli in questo modo:

$$\begin{cases} u_x(x, y, z) = u_{x1}(y) \\ u_y(x, y, z) = u_{y1}(y) - u_{x1,y}(y)x - u_{z1,y}(y)z \\ u_z(x, y, z) = u_{z1}(y) \end{cases} \quad (2.9)$$

in cui le uniche incognite sono u_{x1} , u_{y1} e u_{z1} .

2.1.2 Trave di Timoshenko - TBT

Con la teoria di Timoshenko si vuole superare la terza ipotesi di EBBT. Dunque la sezione trasversale non è più vincolata a rimanere perpendicolare alla superficie neutra ed è possibile che nella struttura deformata ci siano anche le deformazioni a taglio γ_{yx} e γ_{yz} . Poichè non è più valida l'equazione (2.7), non è più possibile esprimere le rotazioni in funzione delle derivate degli spostamenti come fatto in (2.8), che perciò restano incognite.

Il campo di moto in questo caso è

$$\begin{cases} u_x(x, y, z) = u_{x1}(y) \\ u_y(x, y, z) = u_{y1}(y) + \phi_z(y)x + \phi_x(y)z \\ u_z(x, y, z) = u_{z1}(y) \end{cases} \quad (2.10)$$

ed è composto da 5 incognite, 3 comuni a EBBT a cui si aggiungono anche le rotazioni ϕ_z e ϕ_x .

Le teorie appena esposte non sono però in grado di descrivere il campo di spostamento in presenza di un momento torcente applicato. Considerando sempre il sistema di riferimento in Figura 2.1, se si applica un momento lungo l'asse y , allora è necessario introdurre una rotazione ϕ_y che in generale sarà funzione di x, y, z . Se è valida la prima ipotesi di Eulero-Bernoulli (che vale anche per la trave di Timoshenko), la descrizione della rotazione si semplifica e risulta che $\phi_y = \phi_y(y)$.

Nella sezione trasversale il campo di spostamenti, dovuto alla sola torsione, si può scrivere come

$$\begin{cases} u_x(y, z) = z\phi_y \\ u_z(x, y) = -x\phi_y \end{cases} \quad (2.11)$$

Queste deformazioni dovranno essere sommate a (2.9) o (2.10) per avere un modello trave capace di descrivere deformazioni a taglio e torsione, con lo svantaggio però di introdurre un ulteriore grado di libertà dato da ϕ_y .

2.2 Carrera Unified Formulation - CUF

La *Carrera Unified Formulation* permette di calcolare il campo di spostamenti su una sezione con una formulazione unificata, cioè con un'unica formula indipendente dall'ordine e dal tipo di teoria utilizzata per l'approssimazione, che è invece fornito come input [8].

Seguendo la notazione di Einstein, la formulazione della CUF si può scrivere come:

$$\mathbf{u} = F_\tau \mathbf{u}_\tau \quad \tau = 1, \dots, M \quad (2.12)$$

dove F_τ sono le funzioni di espansione, \mathbf{u}_τ è il vettore degli spostamenti e M indica il numero di funzioni di espansione che è necessario utilizzare per una teoria di un certo ordine N (nella Tabella 2.1 è mostrato il legame tra le due nel caso di un'approssimazione alla Taylor).

2.2.1 Polinomio di Taylor a ordine N

La formulazione di Carrera (2.12) può essere interpretata come un polinomio tipo Taylor di ordine N , dove N è l'ordine dell'espansione ed è un parametro di input che può essere scelto a piacere. Nel caso in cui si scelga un modello con $N = 1$ allora l'approssimazione prende il nome di CLEC (*Complete Linear Expansion Case*) e il campo di spostamenti ha la seguente forma:

$$\begin{cases} u_x = u_{x1} + xu_{x2} + zu_{x3} \\ u_y = u_{y1} + xu_{y2} + zu_{y3} \\ u_z = u_{z1} + xu_{z2} + zu_{z3} \end{cases} \quad (2.13)$$

dove le funzioni di espansione sono

$$\begin{aligned} F_{\tau=1} &= 1 \\ F_{\tau=2} &= x \\ F_{\tau=3} &= z \end{aligned}$$

I campi di spostamento previsti dalle teorie di Eulero-Bernoulli (2.9) e di Timoshenko (2.10) possono essere visti come caso particolare della CLEC. Infatti, introducendo le ipotesi (2.4) e (2.6) si ottiene il campo di spostamento previsto da TBT, dove ora

$$\phi_z(y) = u_{x2}, \quad \phi_x(y) = u_{x3}$$

Inserendo anche l'ipotesi (2.7) si ottiene il campo previsto da EBBT.

A differenza delle teorie classiche, la teoria CLEC introduce dei termini con il quale è possibile calcolare anche le deformazioni nella sezione trasversale della trave, in particolare la deformazione a torsione. Questo viene descritto nel dettaglio in [9].

Nell'elaborato saranno utilizzati polinomi di Taylor non solo per $N = 1$, ma anche per $N = 2, 3, 4$. Nella Tabella 2.1 vengono mostrate le funzioni di espansione per gli ordini $N = 0, 1, 2, 3, 4$ della teoria, e il numero M di queste.

N	M	F_τ
0	1	$F_1 = 1$
1	3	$F_2 = x, F_3 = z$
2	6	$F_4 = x^2, F_5 = xz, F_6 = z^2$
3	10	$F_7 = x^3, F_8 = x^2z, F_9 = xz^2, F_{10} = z^3$
4	15	$F_{11} = x^4, F_{12} = x^3z, F_{13} = x^2z^2, F_{14} = xz^3, F_{15} = z^4$

Tabella 2.1: Funzioni di espansione nel caso di polinomi di Taylor fino al quarto ordine

Una forma compatta del campo di spostamenti valida per ogni ordine N è la seguente:

$$u_{x_i} = \underbrace{\dots}_{1, \dots, N-1} + \underbrace{\sum_{M=0}^N x^{(N-M)} z^M u_{x_i \frac{N(N+1)+M+1}{2}}}_{\text{ordine } N} \quad (2.14)$$

dove $x_i = x, y, z$.

Nell'approssimazione del campo di spostamenti con i polinomi di Taylor, il numero di incognite è dettato dall'ordine della teoria stessa. Infatti, se le incognite sono le M componenti del vettore u_τ in (2.12), dalla Tabella 2.1 si nota come queste aumentano all'aumentare dell'ordine della teoria.

Di seguito viene riportato il numero di incognite per determinare lo spostamento di un singolo nodo nelle tre direzioni x, y, z mediante i polinomi di Taylor di ordine N (TEN):

$$TE1 \longrightarrow DOF_s = 3 \cdot M = 3$$

$$TE2 \longrightarrow DOF_s = 3 \cdot M = 18$$

$$TE3 \longrightarrow DOF_s = 3 \cdot M = 30$$

$$TE4 \longrightarrow DOF_s = 3 \cdot M = 45$$

dove i valori di M sono espressi nella Tabella 2.1.

2.2.2 Polinomi di Lagrange

Nell'elaborato sono saranno utilizzati polinomi di Lagrange basati su 4 e 9 punti (LE4 e LE9 rispettivamente). Viene utilizzata una formulazione con coordinate normalizzate poiché offre diversi vantaggi. Considerando il caso di una sezione a quattro nodi, in Figura 2.2 viene mostrata prima la sezione reale della struttura e in seguito la sezione che la struttura assume in seguito all'interpolazione mediante i polinomi di Lagrange in coordinate normalizzate.

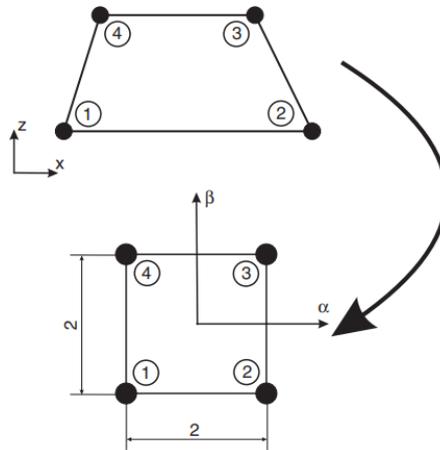


Figura 2.2: Sezione trasversale in coordinate reali e normalizzate per LE4.

In questa nuova forma i polinomi di Lagrange sono lineari nelle variabili α e β , che rappresentano le coordinate normalizzate, con l'aggiunta di un termine bilineare $\alpha\beta$.

$$F_\tau = \frac{1}{4} (1 + \alpha\alpha_\tau) (1 + \beta\beta_\tau), \quad \tau = 1, 2, 3, 4 \quad (2.15)$$

dove α_τ e β_τ sono invece le coordinate dei 4 punti della sezione riportate nella tabella seguente.

Punto	α_τ	β_τ
1	-1	-1
2	1	-1
3	1	1
4	-1	1

Tabella 2.2: Coordinate normalizzate dei quattro punti della sezione

In questo modo i polinomi di Lagrange, ossia le funzioni di espansione per il modello di Lagrange, diventano:

$$\begin{aligned}
 F_{\tau=1} &= \frac{1}{4} (1 - \alpha) (1 - \beta) \\
 F_{\tau=2} &= \frac{1}{4} (1 + \alpha) (1 - \beta) \\
 F_{\tau=3} &= \frac{1}{4} (1 + \alpha) (1 + \beta) \\
 F_{\tau=4} &= \frac{1}{4} (1 - \alpha) (1 + \beta)
 \end{aligned}$$

Dunque il campo di spostamento calcolato secondo (2.12) è

$$\begin{cases}
 u_x = F_1 u_{x1} + F_2 u_{x2} + F_3 u_{x3} + F_4 u_{x4} \\
 u_y = F_1 u_{y1} + F_2 u_{y2} + F_3 u_{y3} + F_4 u_{y4} \\
 u_z = F_1 u_{z1} + F_2 u_{z2} + F_3 u_{z3} + F_4 u_{z4}
 \end{cases} \quad (2.16)$$

In modo analogo è possibile identificare il campo di spostamenti della sezione nel caso di un polinomio di Lagrange basato su 9 punti.

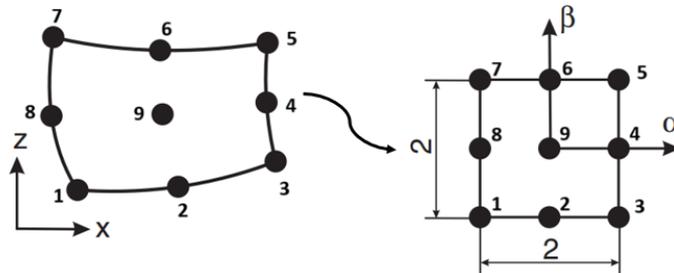


Figura 2.3: Sezione trasversale in coordinate reali e normalizzate per LE9.

I polinomi di Lagrange ora si possono scrivere come:

$$\begin{aligned}
 F_\tau &= \frac{1}{4} (\alpha^2 + \alpha\alpha_\tau) (\beta^2 + \beta\beta_\tau), & \tau &= 1, 3, 5, 7 \\
 F_\tau &= \frac{1}{2} \beta_\tau^2 (\beta^2 + \beta\beta_\tau) (1 - \alpha^2) + \frac{1}{2} \alpha_\tau^2 (\alpha^2 + \alpha\alpha_\tau) (1 - \beta^2), & \tau &= 2, 4, 6 \\
 F_\tau &= (1 - \alpha^2) (1 - \beta^2), & \tau &= 9
 \end{aligned}$$

dove le coordinate α_τ e β_τ sono riportate nella tabella di seguito.

Punto	α_τ	β_τ
1	-1	-1
2	0	-1
3	1	-1
4	1	0
5	1	1
6	0	1
7	-1	1
8	-1	0
9	0	0

Tabella 2.3: Coordinate normalizzate dei nove punti della sezione

Sostituendo i valori riportati in Tabella 2.3 i polinomi di Lagrange diventano:

$$\begin{aligned}
 F_{\tau=1} &= \frac{1}{4} (\alpha^2 - \alpha) (\beta^2 - \beta) \\
 F_{\tau=2} &= \frac{1}{2} (\beta^2 - \beta) (1 - \alpha^2) \\
 F_{\tau=3} &= \frac{1}{4} (\alpha^2 + \alpha) (\beta^2 - \beta) \\
 F_{\tau=4} &= \frac{1}{2} (\alpha^2 + \alpha) (1 - \beta^2) \\
 F_{\tau=5} &= \frac{1}{4} (\alpha^2 + \alpha) (\beta^2 + \beta) \\
 F_{\tau=6} &= \frac{1}{2} (\beta^2 + \beta) (1 - \alpha^2) \\
 F_{\tau=7} &= \frac{1}{4} (\alpha^2 - \alpha) (\beta^2 + \beta) \\
 F_{\tau=8} &= \frac{1}{2} (\alpha^2 - \alpha) (1 - \beta^2) \\
 F_{\tau=9} &= (1 - \alpha^2) (1 - \beta^2)
 \end{aligned}$$

Il campo di spostamenti secondo (2.12) è

$$\begin{cases} u_x = F_1 u_{x1} + F_2 u_{x2} + F_3 u_{x3} + F_4 u_{x4} + F_5 u_{x5} + F_6 u_{x6} + F_7 u_{x7} + F_8 u_{x8} + F_9 u_{x9} \\ u_y = F_1 u_{y1} + F_2 u_{y2} + F_3 u_{y3} + F_4 u_{y4} + F_5 u_{y5} + F_6 u_{y6} + F_7 u_{y7} + F_8 u_{y8} + F_9 u_{y9} \\ u_z = F_1 u_{z1} + F_2 u_{z2} + F_3 u_{z3} + F_4 u_{z4} + F_5 u_{z5} + F_6 u_{z6} + F_7 u_{z7} + F_8 u_{z8} + F_9 u_{z9} \end{cases} \quad (2.17)$$

Indipendentemente dal numero di punti scelto per descrivere la sezione con il polinomio di Lagrange, le incognite del problema saranno sempre gli spostamenti trasversali di tali punti nelle tre direzioni. Dunque il numero di incognite è legato direttamente a questo parametro. Ipotizzando di voler discretizzare solo la sezione frontale di una trave con m punti (così che si ha un modello LEm), allora si avranno $3m$ incognite.

$$LE4 \longrightarrow DOF_s = 3 \cdot 4 = 12$$

$$LE9 \longrightarrow DOF_s = 3 \cdot 9 = 27$$

2.3 Formulazione FEM mediante CUF

Per ottenere una formulazione finale per il calcolo degli spostamenti valida per i codici ad elementi finiti è necessario fare ricorso al principio degli spostamenti virtuali PVD (*Principle of Virtual Displacements*). Prima di applicare il principio è bene definire il campo di spostamenti di tutta la struttura mediante (2.12) e mediante le **funzioni di forma**. Infatti, con la formulazione di Carrera si calcola lo spostamento nodale; le funzioni di forma si occupano invece di interpolare questi spostamenti lungo tutta la struttura.

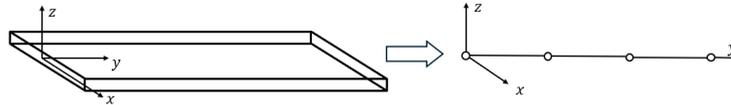


Figura 2.4: Trave discretizzata in elementi B2

Ad esempio, considerando la trave in Figura 2.4 discretizzata con un certo numero di elementi beam a due nodi (elementi B2), gli spostamenti nodali secondo (2.12), utilizzando una teoria di Taylor al primo ordine ($N = 1, \tau = 1, 2, 3$), sono:

$$\mathbf{u}_i = F_\tau \mathbf{u}_{\tau i} \quad (2.18)$$

dove

$$\mathbf{u}_{\tau i} = \left\{ u_{x1} \quad u_{y1} \quad u_{z1} \quad u_{x2} \quad u_{y2} \quad u_{z2} \quad u_{x3} \quad u_{y3} \quad u_{z3} \right\}_i^T \quad (2.19)$$

Il campo di spostamenti lungo la trave è invece

$$\mathbf{u} = N_i \mathbf{u}_i \quad (2.20)$$

dove sono state utilizzate le funzioni di forma N_i . Poiché si sta considerando la trave discretizzata con elementi monodimensionali e a due nodi, sono necessarie due funzioni di forma espresse mediante polinomi di primo grado in una sola variabile adimensionale:

$$\begin{cases} N_1 = \frac{1}{2}(1-r) \\ N_2 = \frac{1}{2}(1+r) \end{cases} \quad (2.21)$$

con $-1 \leq r \leq +1$ coordinata adimensionale lungo l'elemento.

Combinando l'equazione (2.18) nella (2.20) si ottiene una formulazione finale:

$$\mathbf{u} = N_i(y) F_\tau(x, z) \mathbf{u}_{\tau i} \quad (2.22)$$

che può essere esplicitata nelle tre componenti x, y, z :

$$\begin{cases} u_x = N_i(y) F_\tau(x, z) u_{x\tau i} \\ u_y = N_i(y) F_\tau(x, z) u_{y\tau i} \\ u_z = N_i(y) F_\tau(x, z) u_{z\tau i} \end{cases}$$

Grazie alle funzioni di forma è dunque possibile determinare lo spostamento in qualunque punto della trave, anche al di fuori dei nodi. Di seguito viene mostrato un calcolo facendo riferimento alla Figura 2.5.

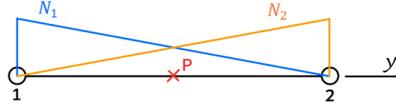


Figura 2.5: Punto generico lungo l'asse della trave

Il punto P è in una posizione intermedia ai nodi 1 e 2. Con l'equazione (2.22) lo spostamento in quel punto è

$$\mathbf{u}(x_p, y_p, z_p) = N_1(y_p) F_\tau(x_p, z_p) \mathbf{u}_{\tau 1} + N_2(y_p) F_\tau(x_p, z_p) \mathbf{u}_{\tau 2} \quad (2.23)$$

Scomponendo lungo le tre direzioni ed esplicitando le funzioni di espansione, considerando la CLEC, si giunge a

$$\begin{cases} u_x = N_{1,p} u_{x1,1} + N_{1,p} x_p u_{x2,1} + N_{1,p} z_p u_{x3,1} + N_{2,p} u_{x1,2} + N_{2,p} x_p u_{x2,2} + N_{2,p} z_p u_{x3,2} \\ u_y = N_{1,p} u_{y1,1} + N_{1,p} x_p u_{y2,1} + N_{1,p} z_p u_{y3,1} + N_{2,p} u_{y1,2} + N_{2,p} x_p u_{y2,2} + N_{2,p} z_p u_{y3,2} \\ u_z = N_{1,p} u_{z1,1} + N_{1,p} x_p u_{z2,1} + N_{1,p} z_p u_{z3,1} + N_{2,p} u_{z1,2} + N_{2,p} x_p u_{z2,2} + N_{2,p} z_p u_{z3,2} \end{cases}$$

A questo punto è possibile applicare il principio degli spostamenti virtuali. Questo è definito come:

$$\delta \mathbf{u} = N_j F_s \delta \mathbf{u}_{sj}$$

da cui è possibile ottenere la deformazione virtuale con l'equazione (2.2):

$$\delta \boldsymbol{\varepsilon} = \mathbf{b} \cdot \delta \mathbf{u} = \mathbf{b} N_j F_s \delta \mathbf{u}_{sj}$$

Secondo il PVD il legame tra le forze virtuali interne, esterne e di inerzia è

$$\delta L_{int} = \delta L_{ext} - \delta L_{ine} \quad (2.24)$$

Il lavoro virtuale delle forze interne si può scrivere come

$$\delta L_{int} = \int_{\mathcal{V}} \delta \boldsymbol{\varepsilon}^T \cdot \boldsymbol{\sigma} d\mathcal{V} = \int_{\mathcal{V}} (\mathbf{b} N_j F_s \delta \mathbf{u}_{sj}) \cdot (\mathbf{C} \mathbf{b} N_i F_\tau u_{\tau i}) d\mathcal{V} \quad (2.25)$$

dove è stata utilizzata l'equazione (2.3) per esprimere il vettore $\boldsymbol{\sigma}$. Gli indici j, s fanno riferimento al sistema congruente di spostamenti e deformazioni, mentre gli indici i, τ fanno riferimento al sistema equilibrato di forze esterne e tensioni interne. L'ultima equazione può essere ancora elaborata.

$$\delta L_{int} = \delta \mathbf{u}_{sj}^T \int_{\mathcal{V}} F_s N_j \mathbf{b}^T \mathbf{C} \mathbf{b} N_i F_\tau d\mathcal{V} \mathbf{u}_{\tau i} = \delta \mathbf{u}_{sj}^T \mathbf{k}^{\tau sij} \mathbf{u}_{\tau i} \quad (2.26)$$

Il termine $\mathbf{k}^{\tau sij}$ è detto **nucleo fondamentale** e rappresenta una matrice costruita iterando su quattro indici; il suo assemblaggio è omissso in questo elaborato.

Il lavoro virtuale delle forze di inerzia si può esprimere invece come

$$\delta L_{ine} = \int_{\mathcal{V}} \delta \mathbf{u}^T \rho \ddot{\mathbf{u}} d\mathcal{V}$$

in cui ρ è la densità del materiale e $\ddot{\mathbf{u}}$ è l'accelerazione. Esplicitando i termini di spostamento e spostamento virtuale si può costruire la matrice di massa in modo analogo a quanto fatto con il nucleo fondamentale. Il risultato di questo procedimento è riportato di seguito:

$$\delta L_{ine} = \delta \mathbf{u}_{sj}^T \mathbf{m}^{\tau sij} \ddot{\mathbf{u}}_{\tau i} \quad (2.27)$$

Infine, il lavoro virtuale delle forze esterne \mathbf{P} , applicate nel punto x_p, y_p, z_p è

$$\delta L_{ext} = \delta \mathbf{u}_s^T \cdot \mathbf{P} = N_j(y_p) F_s(x_p, z_p) \delta \mathbf{u}_{sj}^T \cdot \mathbf{P} \quad (2.28)$$

Per l'assemblaggio del nucleo fondamentale $\mathbf{k}^{\tau sij}$ e l'espressione della matrice di massa $\mathbf{m}^{\tau sij}$ si rimanda a [8].

2.4 Equazioni di governo

Verranno ora richiamate le equazioni fondamentali per la risoluzione dei problemi strutturali utilizzate anche in questo elaborato, dal codice di calcolo MUL2.

2.4.1 Analisi statica - SOL 101

Viene condotta nel caso in cui si vogliono determinare gli spostamenti della struttura, il suo stato tensionale e le sue deformazioni. In questo caso, le uniche forze coinvolte sono i carichi esterni e le reazioni elastiche interne. L'equazione (2.24) si riduce a

$$\delta L_{int} = \delta L_{ext} \quad (2.29)$$

dove i lavori sono

$$\begin{aligned} \delta L_{int} &= \delta \mathbf{U}^T \mathbf{K} \mathbf{U} \\ \delta L_{ext} &= \delta \mathbf{U}^T \mathbf{P} \end{aligned}$$

La matrice \mathbf{K} è la matrice di rigidità, non il nucleo fondamentale, poiché le funzioni di forma e di espansione sono contenute nel vettore degli spostamenti \mathbf{U} . \mathbf{P} è invece il vettore delle forze esterne applicate.

Sostituendo le espressioni per i due lavori nella (2.29) si ottiene:

$$\delta \mathbf{U}^T \mathbf{K} \mathbf{U} = \delta \mathbf{U}^T \mathbf{P}$$

ed infine

$$\mathbf{K} \mathbf{U} = \mathbf{P} \quad (2.30)$$

Poiché si è interessati agli spostamenti, l'equazione (2.30) si risolve invertendo \mathbf{K} , il che è possibile solo se non sono consentiti spostamenti rigidi della struttura:

$$\mathbf{U} = \mathbf{K}^{-1} \mathbf{P} \quad (2.31)$$

2.4.2 Analisi dinamica (vibrazioni libere) - SOL 103

In questa analisi, poiché si vogliono studiare le frequenze libere della struttura, non ci sono carichi esterni applicati, per cui

$$\delta L_{ext} = 0$$

e applicando il PVD si ottiene:

$$\delta L_{int} = -\delta L_{ine} \quad (2.32)$$

Si ha cioè un bilancio tra le forze inerziali e quelle elastiche interne alla struttura. Le prime vengono espresse come

$$\delta L_{ine} = \delta \mathbf{U}^T \mathbf{M} \ddot{\mathbf{U}}$$

Sostituendo le espressioni per i lavori si ha

$$\delta \mathbf{U}^T \mathbf{M} \ddot{\mathbf{U}} + \delta \mathbf{U}^T \mathbf{K} \mathbf{U} = 0$$

e semplificando il vettore degli spostamenti trasposto:

$$\mathbf{M} \ddot{\mathbf{U}} + \mathbf{K} \mathbf{U} = 0 \quad (2.33)$$

La soluzione all'equazione (2.33) può essere trovata risolvendo un problema agli autovalori una volta ipotizzata una soluzione di tipo armonico, cioè

$$\mathbf{U} = \bar{\mathbf{U}} e^{i\omega t} \rightarrow \dot{\mathbf{U}} = -\omega^2 \bar{\mathbf{U}} e^{i\omega t}$$

Sostituendo la soluzione e la sua derivata seconda nell'equazione (2.33) si ottiene:

$$\omega^2 \mathbf{M} - \mathbf{K} = 0 \tag{2.34}$$

Capitolo 3

Formulazione aerodinamica

L'aeroelasticità studia l'interazione reciproca tra una corrente fluida e una struttura deformabile. Un modello strutturale che descrive la deformazione della struttura, dovuta alle forze aerodinamiche, è già stato presentato nel Capitolo 2. Resta da descrivere dunque un modello per ottenere le forze aerodinamiche; a tal scopo si farà uso di un particolare metodo ai pannelli. Sarà prima fornita una breve panoramica sui metodi ai pannelli.

3.1 Metodi ai pannelli

La teoria della linea portante di Prandtl (*Lifting Line Theory* LLT) fornisce risultati accurati per un'ala tridimensionale, tuttavia ha delle limitazioni dovute in parte alla forma dell'ala. Infatti i risultati forniti dalla LLT valgono finché la superficie portante ha un alto Aspect Ratio [10] e piccoli angoli di freccia. Per geometrie più complesse, con ali a freccia o a delta, al giorno d'oggi si fa riferimento a metodi ai pannelli basati tutti sulle ipotesi di ali sottili, piccoli angoli di attacco e un flusso stazionario, inviscido, irrotazionale e incompressibile.

L'idea di base è quella di dividere il corpo aerodinamico in un reticolo di pannelli. Si può riassumere il metodo in tre punti principali [11]:

1. definire la geometria dei pannelli che costituiscono il reticolo;
2. collocare nei *load point* di ogni pannello una singolarità di tipo pozzo/sorgente, vortice o doppietta e specificarne l'intensità;
3. specificare la condizione al bordo applicata a tutti i *control point* di ogni pannello.

A seconda di come vengono scelti questi tre punti si hanno dei metodi numerici diversi che si differenziano per affidabilità, stabilità e costo computazionale.

3.2 Vortex Lattice Method - VLM

Nel VLM la singolarità considerata è un filamento vorticoso a ferro di cavallo che passa lungo il quarto anteriore del pannello, in corrispondenza del load point, e si estende fino all'infinito a valle.

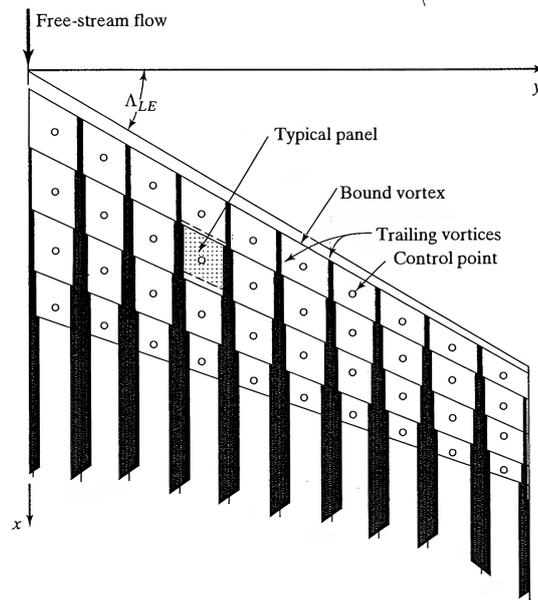


Figura 3.1: Distribuzione dei pannelli e dei filamenti vorticosi su un'ala a freccia

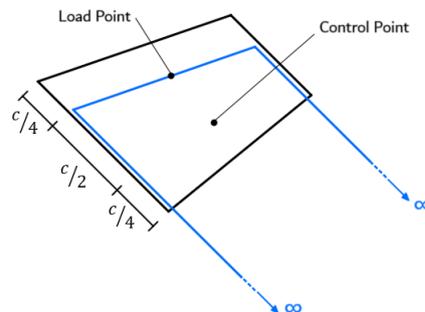


Figura 3.2: Generico pannello utilizzato nel VLM. Sono mostrati la posizione del filamento vorticoso, il load point e il control point.

Il metodo si basa sulle seguenti ipotesi:

- si considera un'aerodinamica stazionaria;
- si considera un flusso potenziale per cui vale l'equazione di Laplace per il potenziale della velocità ϕ : $\nabla^2 \phi = 0$. Ciò significa che si considera un flusso inviscido, irrotazionale e incompressibile;
- lo spessore dell'ala è trascurato, dunque i pannelli discretizzano la superficie media dell'ala;

- la condizione al bordo applicata ai control points di ogni pannello è di tangenza a parete.

Considerando ancora l'ipotesi di un'ala piana e soggetta a piccoli angoli di attacco della corrente, la condizione al bordo si può scrivere come

$$\mathbf{V}_i^{jT} \cdot \mathbf{n}_i = 0 \quad (3.1)$$

dove \mathbf{V}_i^j è la velocità indotta dal vortice del pannello j -esimo, sul control point del pannello i -esimo; mentre \mathbf{n}_i è il versore normale alla superficie del pannello i -esimo (Figura 3.4).

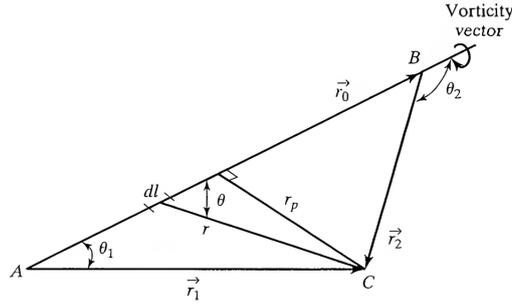


Figura 3.3: Velocità indotta in un punto C da un filamento vorticoso

In generale, la velocità indotta in un punto C distante r da un tratto infinitesimo di filamento vorticoso, di intensità Γ e di lunghezza $d\ell$, si può calcolare con la *legge di Biot-Savart*:

$$d\mathbf{V} = \frac{\Gamma}{4\pi r^3} (d\ell \times \mathbf{r})$$

In modo analogo si può ottenere la velocità indotta \mathbf{V}_i^j . Il contributo complessivo dato dalla somma di tutti i N_{AP} pannelli e dalla velocità di free stream, considerata parallela all'asse x mostrato in Figura 3.1, è:

$$\mathbf{V}_i = \mathbf{V}_\infty + \sum_{j=1}^{N_{AP}} \mathbf{C}_i^j \Gamma_j \quad (3.2)$$

dove il termine

$$\mathbf{C}_i^j = \frac{1}{4\pi} \int_{\ell} \frac{d\ell \times \mathbf{r}}{r^3}$$

è il coefficiente di influenza e dipende dalla geometria del filamento vorticoso poiché i pannelli possono avere area diversa tra di loro, e dalla loro distanza rispetto al control point del pannello i -esimo.

L'equazione per la condizione al bordo (3.1), se applicata a tutti i pannelli, può essere riscritta come:

$$\left(\mathbf{V}_\infty + \sum_{j=1}^{N_{AP}} \mathbf{C}_i^j \Gamma_j \right)^T \cdot \mathbf{n}_i = 0 \iff V_\infty \mathbf{i}^T \cdot \mathbf{n}_i + \sum_{j=1}^{N_{AP}} \Gamma_j \mathbf{C}_i^{jT} \cdot \mathbf{n}_i = 0 \quad (3.3)$$

Il termine C_i^j può essere scomposto nelle tre direzioni, ma con l'ipotesi di piccole perturbazioni si ha che l'unica componente non trascurabile è quella in direzione z . Definendo w_i' la componente in direzione z della velocità indotta solo dai vortici, allora si può riscrivere l'ultima equazione come

$$V_\infty \mathbf{i}^T \cdot \mathbf{n}_i + w_i' \cdot \mathbf{n}_i = 0$$

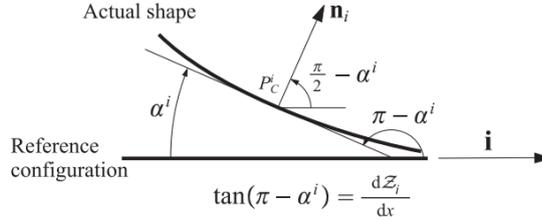


Figura 3.4: Rappresentazione del pannello i -esimo nella sua forma deformata, reale, e nella forma non deformata presa come riferimento

Per svolgere i prodotti scalari è utile fare riferimento alla Figura 3.4 dove si vedono facilmente che

$$\mathbf{i} \cdot \mathbf{n}_i = \cos\left(\frac{\pi}{2} - \alpha^i\right) = \sin \alpha^i$$

e che

$$\mathbf{k}^T \cdot \mathbf{n}_i = \cos \alpha^i$$

Sostituendo i prodotti scalari allora si ha:

$$\begin{aligned} V_\infty \sin \alpha^i + w_i' \cos \alpha^i = 0 &\implies V_\infty \tan \alpha^i + w_i' = 0 \iff \\ \iff -V_\infty \tan(\pi - \alpha^i) + w_i' = 0 &\implies V_\infty \frac{dz_i}{dx} + w_i' = 0 \end{aligned}$$

dove la derivata nell'ultima equazione rappresenta la pendenza del pannello deformato. In conclusione, definendo la velocità trasversale adimensionale $w_i = w_i'/V_\infty$ la condizione al bordo è

$$\frac{dz_i}{dx} = w_i \quad (3.4)$$

e rappresenta un legame tra la condizione di deformazione della struttura e il campo di moto aerodinamico.

È possibile legare la velocità indotta sul generico pannello i -esimo con i salti di pressione sul pannello j -esimo. Per fare ciò si ricorda l'espressione della portanza per unità di lunghezza sul pannello j -esimo lungo Δx_j

$$L_j' = \frac{1}{2} \rho_\infty V_\infty^2 \Delta x_j \Delta p_j$$

Secondo il teorema Kutta-Joukowski la portanza, per unità di lunghezza, si esprime come

$$L_j' = \rho_\infty V_\infty \Gamma_j$$

Uguagliando le ultime due espressioni si trova

$$\Gamma_j = \frac{1}{2} \Delta x_j V_\infty \Delta p_j$$

Sostituendo l'espressione di Γ_j nell'equazione (3.3) si ha

$$V_\infty \mathbf{i}^T \cdot \mathbf{n}_i + \sum_{j=1}^{N_{AP}} \frac{1}{2} \Delta x_j V_\infty \Delta p_j \mathbf{C}_i^{jT} \cdot \mathbf{n}_i = 0 \quad (3.5)$$

Come visto in precedenza il prodotto scalare $\mathbf{i}^T \cdot \mathbf{n}_i = \sin \alpha^i$ e, per l'ipotesi di piccole perturbazioni $\sin \alpha^i \approx \tan \alpha^i$ e, ancora, $\tan \alpha^i \approx -\frac{dz_i}{dx} = -w_i$, l'ultima equazione può essere riscritta come

$$w_i = \sum_{j=1}^{N_{AP}} \frac{1}{2} \Delta x_j \Delta p_j \mathbf{C}_i^{jT} \cdot \mathbf{n}_i = 0 \quad i = 1, \dots, N \quad (3.6)$$

3.3 Doublet Lattice Method - DLM

Nelle applicazioni che prevedono il DLM la superficie aerodinamica è considerata oscillante con una certa frequenza; questo porta ad avere un'aerodinamica instazionaria. È pertanto necessario utilizzare come singolarità, oltre alla linea di vortici, anche una linea di doppiette. I vortici contribuiscono alla parte stazionaria, le doppiette a quella instazionaria. Come evidenziato da [12], il VLM è un caso particolare del DLM in cui la frequenza delle oscillazioni della superficie è nulla.

Le ipotesi che governano questo metodo sono le stesse che governano il VLM a patto di considerare un'aerodinamica instazionaria, invece che stazionaria, e utilizzare come singolarità anche le doppiette oltre che i vortici. Nonostante l'equivalenza le espressioni matematiche per i due metodi sono molto diverse.

Il punto di partenza per il DLM consiste in una relazione che lega la velocità trasversale indotta sui pannelli, con la differenza di pressione in punti diversi della superficie. Questa relazione è stata fornita in [13]. Si definiscono prima le grandezze adimensionali

$$w = \frac{w'}{V_\infty}, \quad \Delta p = \frac{\Delta \mathcal{P}}{\frac{1}{2} \rho_\infty V_\infty^2}$$

$\Delta \mathcal{P}$ è il salto di pressione misurato in Pa . Si suppone ancora che l'oscillazione della superficie si possa descrivere con il modello di oscillatore armonico. Questo induce delle perturbazioni alla velocità e alla pressione:

$$w = \bar{w} e^{i\omega t}, \quad \Delta p = \overline{\Delta p} e^{i\omega t}$$

dove ω è la frequenza angolare delle oscillazioni.

Il legame tra la velocità trasversale misurata nel punto (x, y) dovuta ai salti di pressione nel punto (ξ, η) , si esprime infine attraverso un integrale:

$$\bar{w}(x, y) = \frac{1}{8\pi} \int_A \overline{\Delta p}(\xi, \eta) \mathcal{K}(x_0, y_0, \omega, M) \quad (3.7)$$

dove \mathcal{K} è detta **funzione kernel** e dipende dalla pulsazione ω , dal numero di Mach M e dalla distanza reciproca tra (ξ, η) e (x, y) attraverso (x_0, y_0) definiti come:

$$\begin{cases} x_0 = x - \xi \\ y_0 = y - \eta \end{cases}$$

L'espressione per la funzione kernel nel caso di una superficie planare è

$$\mathcal{K} = \frac{K_1 T_1}{r^2} e^{-i \frac{\omega x_0}{V_\infty}} \quad (3.8)$$

dove l'espressione per il parametro K_1 viene fornita in [14] ed è

$$\begin{aligned} K_1 &= I_1 + \frac{Mr}{R} \frac{e^{-ik_1 u_1}}{\sqrt{1+u_1^2}} \\ I_1 &= \int_{u_1}^{\infty} \frac{e^{-ik_1 u_1}}{(1+u^2)^{3/2}} du \\ r = y_0, \quad u_1 &= \frac{MR - x_0}{\beta^2 r}, \quad k_1 = \frac{\omega r}{V_\infty}, \quad R = \sqrt{x_0^2 + \beta^2 r^2} \\ \beta &= \sqrt{1 - M^2} \\ T_1 &= 1 \end{aligned}$$

Per descrivere il DLM si segue ora la procedura di Rodden in [15] che approssima il legame tra velocità trasversale e pressione con un'equazione polinomiale al quarto ordine.

Per prima cosa viene riscritta l'equazione (3.7) in una forma numerica:

$$\bar{w}_i = \frac{1}{8\pi} \sum_{j=1}^{N_{AP}} \overline{\Delta p_j} \Delta x_j \int_{-e_j}^{+e_j} \mathcal{K}_{ij} d\hat{\eta}_j \quad (3.9)$$

in cui si è considerato il pannello j -esimo lungo $2e_j$ nella direzione locale $\hat{\eta}_j$ del pannello. Facendo comparire la **matrice di influenza aerodinamica** \mathbf{A}^D si può scrivere ancora

$$\bar{w} = \sum_{j=1}^{N_{AP}} A_{ij}^D \overline{\Delta p_j} \quad (3.10)$$

con

$$A_{ij}^D = \frac{\Delta x_j}{8\pi} \int_{-e_j}^{+e_j} \mathcal{K}_{ij} d\hat{\eta}_j \quad (3.11)$$

Sostituendo l'espressione della funzione kernel (3.8) si può riscrivere

$$A_{ij}^D = \frac{\Delta x_j}{8\pi} \int_{-e_j}^{+e_j} \frac{K_1 T_1}{r^2} e^{-i \frac{\omega x_0}{V_\infty}} d\hat{\eta}_j \quad (3.12)$$

L'ultima espressione può essere rivista come la somma di due contributi, uno stazionario calcolato con i VLM e uno instazionario:

$$A_{ij}^D = A_{0ij}^D + A_{1ij}^D \quad (3.13)$$

dove:

- il termine stazionario che deriva dal VLM è

$$A_{0ij}^D = \frac{1}{2} \Delta x_j \mathbf{C}_i^{jT} \cdot \mathbf{n}_i \quad (3.14)$$

- il termine instazionario è

$$A_{1ij}^D = \frac{\Delta x_j}{8\pi} \int_{-e_j}^{+e_j} \frac{\left(e^{-\frac{i\omega x_0}{V_\infty}} K_1 - K_{10} \right) T_1}{r^2} d\hat{\eta}_j \quad (3.15)$$

in cui K_{10} è la componente stazionaria di K_1 .

Quest'ultimo termine può essere approssimato con un'equazione polinomiale al quarto ordine:

$$Q_1(\hat{\eta}_j) = A_1 \hat{\eta}_j^2 + B_1 \hat{\eta}_j + C_1 + D_1 \hat{\eta}_j^3 + E_1 \hat{\eta}_j^4 \approx \left(e^{-\frac{i\omega x_0}{V_\infty}} K_1 - K_{10} \right) T_1$$

Per calcolare i cinque coefficienti A_1, B_1, C_1, D_1, E_1 si impone il passaggio del polinomio in cinque posizioni note lungo il pannello, cioè in $-e_j, -e_j/2, 0, e_j/2, e_j$:

$$\begin{aligned} A_1 &= -\frac{Q_1(-e_j) - 16Q_1(-e_j/2) + 30Q_1(0) - 16Q_1(e_j/2) + Q_1(e_j)}{6e_j^2} \\ B_1 &= \frac{Q_1(-e_j) - 8Q_1(-e_j/2) + Q_1(e_j/2) - Q_1(e_j)}{6e_j} \\ C_1 &= Q_1(0) \\ D_1 &= -2\frac{Q_1(-e_j) - 2Q_1(-e_j/2) + 2Q_1(0) - Q_1(e_j)}{3e_j^3} \\ E_1 &= 2\frac{Q_1(-e_j) - 4Q_1(-e_j/2) + 6Q_1(0) - 4Q_1(e_j) + Q_1(e_j)}{3e_j^4} \end{aligned}$$

In questo modo è possibile riscrivere l'equazione (3.15):

$$\begin{aligned} A_{1ij}^D &= \frac{\Delta x_j}{8\pi} \left[(\bar{y}_i^2 A_1 + \bar{y}_i B_1 + C_1 + \bar{y}_i^3 D_1 + \bar{y}_i^4 E_1) F + \right. \\ &\quad \left. + \left(\bar{y}_i A_1 + \frac{1}{2} B_1 + \frac{3}{2} \bar{y}_i^2 D_1 + 2\bar{y}_i^3 E_1 \right) \log \frac{(\bar{y}_i - e_j)^2}{(\bar{y}_i + e_j)^2} \right. \\ &\quad \left. + 2e_j \left(A_1 + 2\bar{y}_i D_1 + \left(3\bar{y}_i^2 + \frac{1}{3} e_j^2 \right) E_1 \right) \right] \quad (3.16) \end{aligned}$$

in cui

$$F = \int_{-e_j}^{e_j} \frac{d\hat{\eta}_j}{(\hat{y}_i - \hat{\eta}_j)^2} = \frac{2e_j}{y_i^2 - e_j^2}$$

Capitolo 4

Formulazione aeroelastica

In questo capitolo viene presentata la formulazione utilizzata per accoppiare il modello strutturale con il modello aerodinamico, e le equazioni di governo per il problema del flutter. Il primo punto prevede la discussione del modello di spline IPS (*Infinite Plate Spline*) descritto in precedenza da [16]. Per le equazioni di governo sarà necessario definire le matrici generalizzate.

4.1 Il metodo delle spline

Per semplificare l'accoppiamento aeroelastico si possono valutare gli spostamenti della mesh strutturale non nei nodi della sezione trasversale, ma su una superficie di riferimento comune a tutti i punti. Questa viene scelta per comodità in $z = 0$. Dati gli spostamenti nei nodi \mathbf{U} , calcolati ad esempio con la formulazione di Carrera (2.12), per collegarli agli spostamenti \mathbf{Z} valutati nei punti, detti *pseudo-strutturali*, della superficie di riferimento, si può scrivere:

$$\mathbf{Z} = \mathbf{Y}\mathbf{U} \quad (4.1)$$

dove $[\mathbf{U}] \rightarrow N_{DOFs} \times 1$, $[\mathbf{Z}] \rightarrow N_{ps} \times 1$ e $[\mathbf{Y}] \rightarrow N_{ps} \times N_{DOFs}$. In accordo con [16], le componenti del vettore \mathbf{Z} si possono scrivere come

$$Z_i = a_0 + a_1x_i + a_2y_i + \sum_{j=1}^{N_{ps}} F_j r_{ij}^2 \log(r_{ij}^2) \quad (4.2)$$

cioè come combinazione lineare di un'espansione lineare e il contributo delle spline. F_j sono i coefficienti delle spline, a_0 , a_1 , a_2 sono i coefficienti della componente lineare e

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}.$$

I coefficienti delle spline inoltre devono rispettare le seguenti condizioni:

$$\begin{aligned} \sum_{j=1}^{N_{ps}} F_j &= 0 \\ \sum_{j=1}^{N_{ps}} F_j x_j &= 0 \\ \sum_{j=1}^{N_{ps}} F_j y_j &= 0 \end{aligned} \quad (4.3)$$

Ponendo $K_{ij} = r_{ij}^2 \log(r_{ij}^2)$ l'equazione e ricordando le condizioni (4.3), l'equazione (4.2) si può scrivere in forma matriciale per tutti i nodi:

$$\begin{Bmatrix} 0 \\ 0 \\ 0 \\ Z_1 \\ Z_2 \\ \vdots \\ Z_{N_{ps}} \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & \cdots & 1 \\ 0 & 0 & 0 & x_1 & x_2 & \cdots & x_{N_{ps}} \\ 0 & 0 & 0 & y_1 & y_2 & \cdots & y_{N_{ps}} \\ 1 & x_1 & y_1 & 0 & K_{12} & \cdots & K_{1N_{ps}} \\ 1 & x_2 & y_2 & K_{21} & 0 & \cdots & K_{2N_{ps}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N_{ps}} & y_{N_{ps}} & K_{N_{ps}1} & K_{N_{ps}2} & \cdots & 0 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ F_1 \\ F_2 \\ \vdots \\ F_{N_{ps}} \end{Bmatrix} \quad (4.4)$$

L'equazione (4.4) può essere riscritta in forma compatta notando che la matrice è facilmente scomponibile come una matrice a blocchi composta da 4 matrici:

$$\begin{aligned} \mathbf{Z}^* &= [0 \quad 0 \quad 0 \quad Z_1 \quad Z_2 \quad \cdots \quad Z_{N_{ps}}]^T \\ \mathbf{P} &= [a_0 \quad a_1 \quad a_2 \quad F_1 \quad F_2 \quad \cdots \quad F_{N_{ps}}]^T \\ \mathbf{R} &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_{N_{ps}} \\ y_1 & y_2 & \cdots & y_{N_{ps}} \end{bmatrix} \\ \mathbf{K} &= \begin{bmatrix} 0 & K_{12} & \cdots & K_{1N_{ps}} \\ K_{21} & 0 & \cdots & K_{2N_{ps}} \\ \vdots & \vdots & \ddots & \vdots \\ K_{N_{ps}1} & K_{N_{ps}2} & \cdots & 0 \end{bmatrix} \end{aligned}$$

Con questa nuova notazione si può scrivere

$$\mathbf{Z}^* = \begin{bmatrix} \mathbf{0} & \mathbf{R} \\ \mathbf{R}^T & \mathbf{K} \end{bmatrix} \cdot \mathbf{P} \iff \mathbf{Z}^* = \mathbf{G} \cdot \mathbf{P} \quad (4.5)$$

La matrice \mathbf{G} è nota una volta note le coordinate della griglia dei punti, gli spostamenti dei nodi contenuti in \mathbf{Z}^* , tramite l'equazione (4.1), sono noti una volta scelto il modello strutturale. Dunque è possibile calcolare i coefficienti delle spline, contenuti in \mathbf{P} , invertendo l'ultima equazione:

$$\mathbf{P} = \mathbf{G}^{-1} \cdot \mathbf{Z}^*$$

4.2 Pendenza nei *Control Points*

Come visto nel Capitolo 3 la pendenza nei punti di controllo è legata alla condizione al bordo di tangenza a parete. È importante dunque trovare una formulazione matematica adeguata per descriverla.

In un generico punto di controllo k di coordinate $(\mathcal{X}_k, \mathcal{Y}_k)$, lo spostamento calcolato con (4.2) è

$$\mathcal{Z}_k = a_0 + a_1 \mathcal{X}_1 + a_2 \mathcal{Y}_1 + \sum_{j=1}^{N_{ps}} F_j \mathcal{K}_{kj}^2 \log(\mathcal{R}_{kj}^2)$$

Per imporre la condizione di tangenza se ne deve calcolare la derivata rispetto alla direzione del flusso, sia questa la direzione x :

$$\left. \frac{\partial \mathcal{Z}_k}{\partial x} \right|_{(\mathcal{X}_k, \mathcal{Y}_k)} = a_1 + \sum_{j=1}^{N_{ps}} F_j \left. \frac{\partial [\mathcal{K}_{kj}^2 \log(\mathcal{R}_{kj}^2)]}{\partial x} \right|_{(\mathcal{X}_k, \mathcal{Y}_k)} = a_1 + \sum_{j=1}^{N_{ps}} F_j \mathcal{DK}_{kj}$$

Poiché ci sono tanti punti di controllo quanti sono i pannelli aerodinamici, cioè N_{AP} , si riscrive l'equazione della derivata per tutti i pannelli in forma matriciale.

$$\begin{Bmatrix} \frac{\partial \mathcal{Z}_1}{\partial x} \\ \frac{\partial \mathcal{Z}_2}{\partial x} \\ \vdots \\ \frac{\partial \mathcal{Z}_{N_{AP}}}{\partial x} \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \mathcal{DK}_{11} & \mathcal{DK}_{12} & \cdots & \mathcal{DK}_{1N_{ps}} \\ 0 & 1 & 0 & \mathcal{DK}_{21} & \mathcal{DK}_{22} & \cdots & \mathcal{DK}_{2N_{ps}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 0 & \mathcal{DK}_{N_{AP}1} & \mathcal{DK}_{N_{AP}2} & \cdots & \mathcal{DK}_{N_{AP}N_{ps}} \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ F_1 \\ F_2 \\ \vdots \\ F_{N_{ps}} \end{Bmatrix} \quad (4.6)$$

che in forma compatta diventa

$$\frac{\partial \mathbf{Z}}{\partial x} = \mathbf{D} \cdot \mathbf{P}$$

in cui $[\mathbf{Z}] \rightarrow N_{AP} \times 1$, $[\mathbf{D}] \rightarrow N_{AP} \times (N_{ps} + 3)$ e $[\mathbf{P}] \rightarrow (N_{ps} + 3) \times 1$. Come per la matrice \mathbf{G} dell'equazione (4.5), anche la matrice \mathbf{D} è nota una volta nota la griglia dei punti e la posizione dei control points.

L'ultima equazione può essere riscritta legando direttamente le pendenze nei punti di controllo con gli spostamenti nodali della sezione trasversale della struttura, cioè i gradi di libertà

del sistema che sono contenuti nel vettore U . Per fare ciò si sostituisce l'espressione di P trovata in precedenza:

$$\frac{\partial \mathbf{Z}}{\partial x} = \mathcal{D} \cdot \mathbf{G}^{-1} \cdot \mathbf{Z}^*$$

Il vettore \mathbf{Z}^* coincide con \mathbf{Z} se si eliminano le prime tre righe, in cui gli elementi sono nulli. Si può definire anche una matrice \mathcal{S} che coincide con \mathbf{G}^{-1} senza le prime tre colonne. Così facendo l'ultima espressione diventa, senza che cambi la soluzione:

$$\frac{\partial \mathbf{Z}}{\partial x} = \mathcal{D} \cdot \mathcal{S} \cdot \mathbf{Z}$$

A questo punto si introduce l'espressione (4.1) e si giunge ad un'espressione finale:

$$\frac{\partial \mathbf{Z}}{\partial x} = \mathcal{D} \cdot \mathcal{S} \cdot \mathbf{Y} \cdot \mathbf{U} \iff \frac{\partial \mathbf{Z}}{\partial x} = \mathbf{A} \cdot \mathbf{U}$$

4.3 Spostamenti nei *Load Points*

Per il calcolo degli spostamenti nei punti di applicazione del carico si procede in modo analogo al calcolo delle pendenze nei punti di controllo, descritto nella sezione precedente. In questo caso si definisce una matrice $\tilde{\mathcal{D}}^*$ che lega gli spostamenti ai coefficienti delle spline e che dipende dalle coordinate dei load points:

$$\tilde{\mathbf{Z}} = \tilde{\mathcal{D}}^* \cdot \mathbf{P}$$

dove

$$\tilde{\mathcal{D}}^* = \begin{bmatrix} 1 & \tilde{\mathcal{X}}_1 & \tilde{\mathcal{Y}}_1 & \tilde{\mathcal{K}}_{11} & \cdots & \tilde{\mathcal{K}}_{1N_{ps}} \\ 1 & \tilde{\mathcal{X}}_2 & \tilde{\mathcal{Y}}_2 & \tilde{\mathcal{K}}_{21} & \cdots & \tilde{\mathcal{K}}_{2N_{ps}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \tilde{\mathcal{X}}_{N_{AP}} & \tilde{\mathcal{Y}}_{N_{AP}} & \tilde{\mathcal{K}}_{N_{AP}1} & \cdots & \tilde{\mathcal{K}}_{N_{AP}N_{ps}} \end{bmatrix}$$

Seguendo una procedura analoga a quella precedente si riscrive tale equazione in modo da far comparire i gradi di libertà del sistema:

$$\tilde{\mathbf{Z}} = \tilde{\mathbf{A}} \cdot \mathbf{U}$$

4.4 Equazioni di governo per il flutter

L'equazione di governo per i problemi aeroelastici espressa nel dominio del tempo è

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = \mathbf{F}(t) \quad (4.7)$$

in cui compaiono la matrice di massa \mathbf{M} , la matrice di smorzamento \mathbf{C} , la matrice di rigidità \mathbf{K} , il vettore dei carichi aerodinamici applicati \mathbf{F} e il vettore degli spostamenti \mathbf{u} con le sue derivate prima e seconda nel tempo. È un'equazione differenziale al secondo ordine ed è di difficile

risoluzione. Per risolverla in modo più semplice si può applicare la trasformata di Laplace per passare nel dominio delle frequenze:

$$\mathcal{L}(s) = \int_{-\infty}^{\infty} \ell(t) e^{-st} dt$$

dove $s = \sigma + i\omega$ è il parametro di Laplace scomponibile in parte reale e parte immaginaria. Nel dominio delle frequenze non si ha più un'equazione differenziale ma algebrica:

$$\left[\mathbf{M}s^2 + \mathbf{C}s + \mathbf{K} - \frac{1}{2}\rho V_{\infty}^2 \mathbf{Q}(s) \right] \hat{\mathbf{u}}(s) \quad (4.8)$$

dove il vettore delle forze aerodinamiche è stato espresso come

$$\mathbf{F} = \frac{1}{2}\rho V_{\infty}^2 \mathbf{Q}(s) \hat{\mathbf{u}}(s) = 0$$

e si sono considerate oscillazioni armoniche descritte come

$$\mathbf{u}(t) = \hat{\mathbf{u}}e^{st} = \hat{\mathbf{u}}e^{(\sigma+i\omega)t} = \hat{\mathbf{u}}e^{\sigma t} (\cos \omega t + i \sin \omega t)$$

L'equazione (4.8) può essere adimensionalizzata introducendo il parametro di Laplace adimensionale p definito come

$$p = \frac{b}{V_{\infty}} s = \frac{b}{V_{\infty}} (\sigma + i\omega) = g + i\kappa$$

in cui b è una lunghezza di riferimento (nel caso di un profilo alare è pari a metà della corda) e κ è la frequenza ridotta, legata a g come $g = \zeta\kappa$ con ζ *decay rate*. Il parametro di Laplace adimensionale è dunque

$$p = \zeta\kappa + i\kappa$$

L'equazione del moto nel dominio delle frequenze scritta in forma adimensionale diventa

$$\left[\left(\frac{V_{\infty}}{b} \right)^2 \mathbf{M}p^2 + \frac{V_{\infty}}{b} \mathbf{C}p + \mathbf{K} - \frac{1}{2}\rho V_{\infty}^2 \mathbf{Q}(i\kappa) \right] \hat{\mathbf{u}}(p) = 0 \quad (4.9)$$

4.4.1 Matrici generalizzate

Poiché le matrici che compaiono nell'equazione (4.9) hanno dimensione $N_{DOF} \times N_{DOF}$, l'equazione è computazionalmente onerosa da risolvere. Un modo per alleggerire il costo computazionale è quello di utilizzare l'approccio modale in cui si considera il vettore degli spostamenti come dato dalla sovrapposizione dei modi e delle loro ampiezze, incognite:

$$\mathbf{u} = \Phi \cdot \mathbf{q}$$

dove $[\Phi] \rightarrow N_{DOF} \times N_{modes}$ contiene le forme modali e $[\mathbf{q}] \rightarrow N_{modes} \times 1$ contiene le ampiezze dei modi.

Con questo approccio si possono definire le matrici generalizzate che hanno la proprietà di essere matrici quadrate di dimensione $N_{modes} \times N_{modes}$ e di essere diagonali. Si precisa che,

solitamente, $N_{modes} \ll N_{DOF}$ poiché sono necessari solamente i primi modi vibrazionali per descrivere lo spostamento \mathbf{u} .

La matrice di massa generalizzata $\tilde{\mathbf{M}}$ è

$$\tilde{\mathbf{M}} = \Phi^T \mathbf{M} \Phi.$$

In modo analogo si possono ottenere anche le matrici generalizzate di smorzamento e rigidità. I termini diagonali in questo caso sono, rispettivamente:

$$\tilde{\mathbf{C}}_{ii} = 2\xi\omega_i\tilde{\mathbf{M}}_{ii}$$

$$\tilde{\mathbf{K}}_{ii} = \omega_i^2\tilde{\mathbf{M}}_{ii}$$

dove ω_i è la frequenza associata alla forma modale i -esimo e ξ è lo smorzamento strutturale del sistema.

La matrice aerodinamica generalizzata, data una certa frequenza ridotta κ , si può scrivere come

$$Q_{ij}(i\kappa) = \sum_{N=1}^{N_{AP}} \Delta p_j^N(i\kappa) \mathcal{Z}_i^N S_{AP}^N$$

dove

- $\Delta p_j^N(i\kappa)$ è il salto di pressione sul pannello N -esimo indotto dal pannello j -esimo. Questo viene fornito dal DLM.
- \mathcal{Z}_i^N è i -esimo set di spostamenti trasversali del pannello N -esimo forniti dalla griglia strutturale e mappati su quella aerodinamica per mezzo del metodo delle spline visto in precedenza.
- S_{AP}^N è l'area del pannello N -esimo.

La forma finale delle equazioni che saranno risolte è dunque

$$\left[\left(\frac{V_\infty}{b} \right)^2 \tilde{\mathbf{M}} p^2 + \frac{V_\infty}{b} \tilde{\mathbf{C}} p + \tilde{\mathbf{K}} - \frac{1}{2} \rho V_\infty^2 \tilde{\mathbf{Q}}(i\kappa) \right] \hat{\mathbf{q}}(p) = 0 \quad (4.10)$$

Il metodo utilizzato in questo elaborato per risolvere l'equazione matriciale è il *pk-method* descritto in [17] che prevede una risoluzione iterativa per ogni velocità considerata in un certo range.

Capitolo 5

Cenni di reti neurali artificiali

Una rete neurale artificiale è un modello di machine learning costruita in modo tale da risolvere un certo problema similmente a come farebbe un cervello umano, dove la rete è costituita da neuroni collegati tra loro da assoni. A livello biologico, un neurone è in grado di assorbire una certa quantità di cariche elettriche; superato il limite di accumulo l'assone trasferisce la quantità in eccesso ad altri neuroni collegati. Questo processo è detto *sinapsi*.

In modo analogo, una rete neurale artificiale è composta da oggetti matematici irriducibili detti neuroni, o nodi, interconnessi tra loro. Ciascuno di essi riceve in input un segnale, lo elabora con apposite funzioni matematiche, e lo invia agli altri neuroni ad esso collegati. L'elaborazione del segnale è fatta tenendo a mente l'obiettivo che deve perseguire la rete neurale. Si parla infatti di *addestramento* della rete per ottenere un errore accettabile.

La figura seguente mostra un'analogia tra una rete neurale biologica e una artificiale [18].

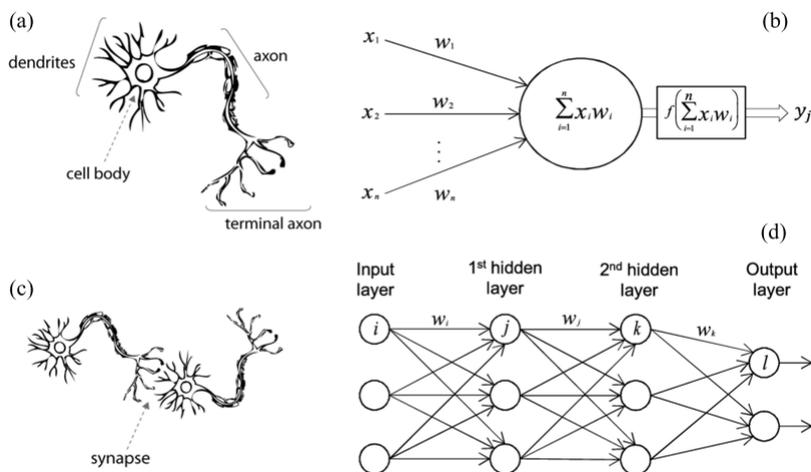


Figura 5.1: Neurone biologico confrontato con un neurone artificiale. (a) neurone biologico; (b) neurone artificiale; (c) sinapsi biologica; (d) sinapsi artificiale

5.1 Il problema della regressione

Il problema della regressione consiste nel trovare una relazione funzionale tra una o più variabili indipendenti x , e una variabile dipendente y . È utile iniziare la trattazione sulle reti neurali da questo problema in quanto rappresenta la base per le reti più complesse; in particolare la regressione lineare è il modello di rete neurale artificiale più semplice possibile.

5.1.1 Regressione lineare

Seguendo la notazione di [19] possiamo dire che, data la popolazione di dati $(x^{(i)}, y^{(i)})$, con $i = 1, \dots, n$, può essere descritta grazie alla seguente equazione:

$$\hat{y} = wx + b \quad (5.1)$$

dove w è detto peso e b è detto bias. La struttura di una rete neurale che descrive la regressione lineare tra una due variabili (una indipendente e una dipendente) è rappresentata nell'immagine seguente.

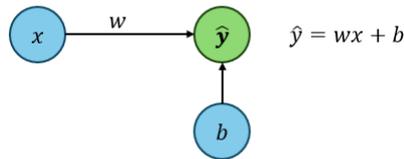


Figura 5.2: Schema del modello lineare con un input (una variabile indipendente).

Dunque, data una qualsiasi $x^{(i)}$, allora $\hat{y}(x^{(i)}) = \hat{y}^{(i)} \approx y^{(i)}$. L'uguaglianza non è esatta ma è affetta da errori che vengono tenuti in conto dalla funzione di perdita (*loss function*). Questa può essere definita in vari modi, uno fra questi, molto comune, è l'errore quadratico medio:

$$L(w, b) = \frac{1}{2} \frac{1}{n} \sum_{i=1}^n (\hat{y}^{(i)} - y^{(i)})^2 = \frac{1}{2} \frac{1}{n} \sum_{i=1}^n [(wx^{(i)} + b) - y^{(i)}]^2 \quad (5.2)$$

La funzione di perdita descrive quindi la qualità dell'approssimazione dei dati esatti con la retta (5.1) o dalla rete in Figura 5.2. Poiché L dipende dalla scelta dei parametri w e b è possibile ottenere una coppia di questi valori che la minimizzino. Questo processo di ottimizzazione viene svolto con il **metodo del gradiente**, un algoritmo che segue i seguenti passi:

1. Si sceglie un punto di partenza per la coppia di parametri w, b :

$$\begin{Bmatrix} w \\ b \end{Bmatrix}_{(0)}$$

2. Si calcola il gradiente della funzione di perdita rispetto ai parametri del passo precedente:

$$\nabla L_{(0)} = \begin{Bmatrix} \frac{\partial L}{\partial w} \\ \frac{\partial L}{\partial b} \end{Bmatrix}_{(0)}$$

L'operatore gradiente definisce la direzione di massima pendenza della funzione L al variare dei parametri w, b ; bisogna quindi rivalutare i parametri procedendo lungo quella direzione.

3. Si aggiornano i pesi calcolati in precedenza con il gradiente ottenuto procedendo di un passo α , detto *learning rate*, nella direzione di massima decrescita della loss function:

$$\begin{Bmatrix} w \\ b \end{Bmatrix}_{(1)} = \begin{Bmatrix} w \\ b \end{Bmatrix}_{(0)} - \alpha \begin{Bmatrix} \frac{\partial L}{\partial w} \\ \frac{\partial L}{\partial b} \end{Bmatrix}_{(0)}$$

4. Si ripete l'algoritmo finché la funzione di perdita è più piccola di una certa tolleranza o finché non si raggiunge il limite massimo di iterazioni.

Varianti del metodo del gradiente

Il metodo del gradiente può essere lento a convergere e non è detto che lo faccia al punto di minimo assoluto della funzione di perdita. Può capitare che converga ad un punto di minimo locale o ad un punto di sella. Esistono pertanto delle varianti rispetto al metodo originario che vengono utilizzate spesso nelle reti neurali, alcune tra queste vengono presentate di seguito.

- **Batch gradient descent:** vengono considerati tutti i dati di training prima di aggiornare i pesi. Si calcola la funzione di perdita i -esima di ciascun dato del set di dati, se ne calcola la derivata rispetto ai pesi e se ne fa la media. Questo valore sarà utilizzato per aggiornare i pesi.

$$\nabla L_{(m)} = \frac{1}{n} \sum_{i=1}^n \nabla L_i$$

$$\begin{Bmatrix} w \\ b \end{Bmatrix}_{(m+1)} = \begin{Bmatrix} w \\ b \end{Bmatrix}_{(m)} - \alpha \nabla L_{(m)}, \quad m = 1, 2, \dots, m$$

dove m è l'epoca corrente e L_i è la loss function calcolata con il dato i -esimo.

- **Stochastic gradient descent:** si considera un solo dato per volta per calcolare il gradiente. In questo modo i pesi vengono aggiornati continuamente in ogni epoca: con n dati all'interno del dataset i pesi vengono calcolati n volte per ogni epoca.

$$\nabla L_{(m)} = \nabla L_i$$

$$\begin{Bmatrix} w \\ b \end{Bmatrix}_{(m+1)} = \begin{Bmatrix} w \\ b \end{Bmatrix}_{(m)} - \alpha \nabla L_{(m)}, \quad m = 1, 2, \dots, m \times n$$

- **Mini-batch gradient descent:** è un metodo ibrido ai due precedenti. Invece di aggiornare i pesi dopo ogni dato, come il metodo precedente, i pesi vengono aggiornati dopo che è stato elaborato un sottogruppo di dati del dataset (*mini-batch*) di dimensione $m_b < m$ e

tale che $m/m_b \in \mathbb{N}$. In questo modo si calcolano i nuovi parametri non ad ogni nuovo dato considerato, ma ad ogni batch.

$$\nabla L_{(m)} = \frac{1}{n} \sum_{i=1}^n \nabla L_i$$

$$\begin{Bmatrix} w \\ b \end{Bmatrix}_{(m+1)} = \begin{Bmatrix} w \\ b \end{Bmatrix}_{(m)} - \alpha \nabla L_{(m)}, \quad m = 1, 2, \dots, \frac{m}{m_b} \times n$$

Il problema della regressione lineare si può estendere al caso in cui la variabile dipendente y dipenda da più variabili indipendenti x_1, \dots, x_d come in Figura 5.3.

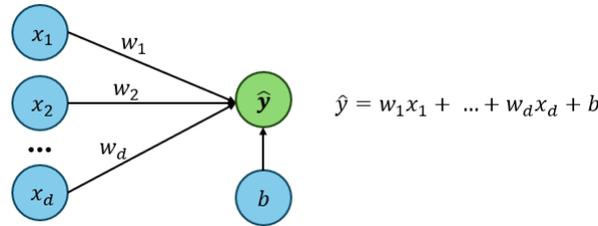


Figura 5.3: Schema del modello lineare con più input

In questo caso la relazione prevede l'output come una combinazione lineare dei vari input x_k moltiplicati ciascuno per il rispettivo peso w_k e sommato al bias b :

$$\hat{y} = w_1 x_1 + \dots + w_d x_d + b$$

o in forma compatta

$$\hat{y} = \mathbf{w}^T \cdot \mathbf{x} + b \tag{5.3}$$

e la loss function diventa:

$$L(w_1, \dots, w_d, b) = \frac{1}{2} \frac{1}{n} \sum_{i=1}^n \left[\left(b + \sum_{k=1}^d w_k x_k^{(i)} \right) - y^{(i)} \right]^2 \tag{5.4}$$

5.1.2 Regressione logistica

Immaginando di avere un set di dati che possono essere ordinati in due classi (classificazione binaria), la regressione logistica si occupa di determinare la *probabilità* che il dato i -esimo della popolazione possa far parte della prima o della seconda classe. Per esprimere matematicamente il concetto si assegnano alle due classi i valori 0 e 1. Un problema di questo tipo non può essere descritto con una relazione lineare tra le variabili coinvolte; è necessario utilizzare una funzione che tenda asintoticamente ai due valori discreti. La funzione ideale sarebbe dunque una funzione

a gradino, ma per tener conto delle incertezze che si possono verificare in fase di distinzione di $x^{(i)}$ in una delle due classi si utilizzano le funzioni logistiche, come ad esempio la **funzione sigmoide** definita come:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (5.5)$$

il cui grafico è rappresentato nella figura seguente ed è confrontato con una funzione a gradino.

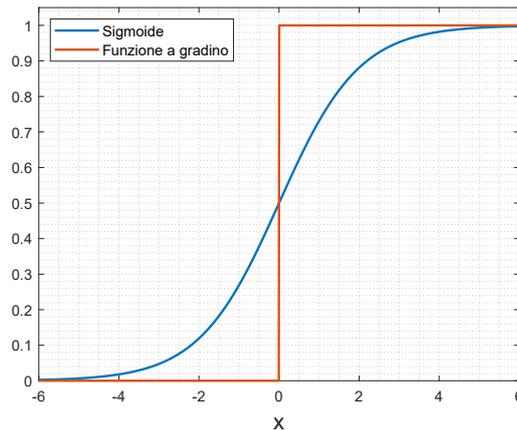


Figura 5.4: Confronto tra una funzione a gradino e la curva sigmoide

Più le incertezze sulla classificazione sono piccole e più la sigmoide si avvicina alla funzione a gradino. Nell'ambito della regressione logistica, per adattare la funzione ai dati della popolazione si agisce su dei parametri, come per la regressione lineare. A tal scopo è utile riscrivere l'equazione (5.5) in modo da far comparire i parametri:

$$\sigma(x) = \frac{1}{1 + e^{-(wx+b)}} \quad (5.6)$$

Il termine di bias b opera una traslazione lungo x della funzione, mentre il peso w esprime l'incertezza sulla classificazione di $x^{(i)}$: infatti, più w è grande, più piccole sono le incertezze e più la sigmoide si avvicina alla funzione a gradino

In un generico problema di regressione logistica costituito da d parametri si può allora scrivere:

$$\hat{y} = \sigma = \frac{1}{1 + e^{-(b+w_1x_1+\dots+w_dx_d)}} \quad (5.7)$$

Come per la regressione lineare, anche per quella logistica si può definire una funzione di perdita, che in questo caso però, data la diversa natura della funzione di approssimazione dei dati, è definita in modo diverso. Si fa pertanto uso del concetto di **cross-entropy** per definire la loss function. In un caso di classificazione binaria si ha:

$$L(w_1, \dots, w_d, b) = -\frac{1}{n} \sum_{i=1}^n \left[y^{(i)} \log(\sigma^{(i)}) + (1 - y^{(i)}) \log(1 - \sigma^{(i)}) \right] \quad (5.8)$$

Capitolo 5. Cenni di reti neurali artificiali

Se il dato i -esimo viene classificato nella classe a cui è attribuito il valore 1, cioè se $y^{(i)} = 1$, allora L si riduce a

$$L = -\frac{1}{n} \sum_{i=1}^n \log(\sigma^{(i)})$$

se ancora, la funzione sigmoide classifica correttamente il dato, cioè $\sigma^{(i)} = 1$, allora $L = 0$; se invece la previsione di σ è totalmente sbagliata si ha $L \rightarrow +\infty$, cioè l'errore che si commette tende all'infinito.

Se invece $y^{(i)} = 0$ allora

$$L = -\frac{1}{n} \sum_{i=1}^n \log(1 - \sigma^{(i)})$$

e $L = 0$ se anche $\sigma^{(i)} = 0$; altrimenti, se $\sigma^{(i)} = 1$ allora $L \rightarrow +\infty$.

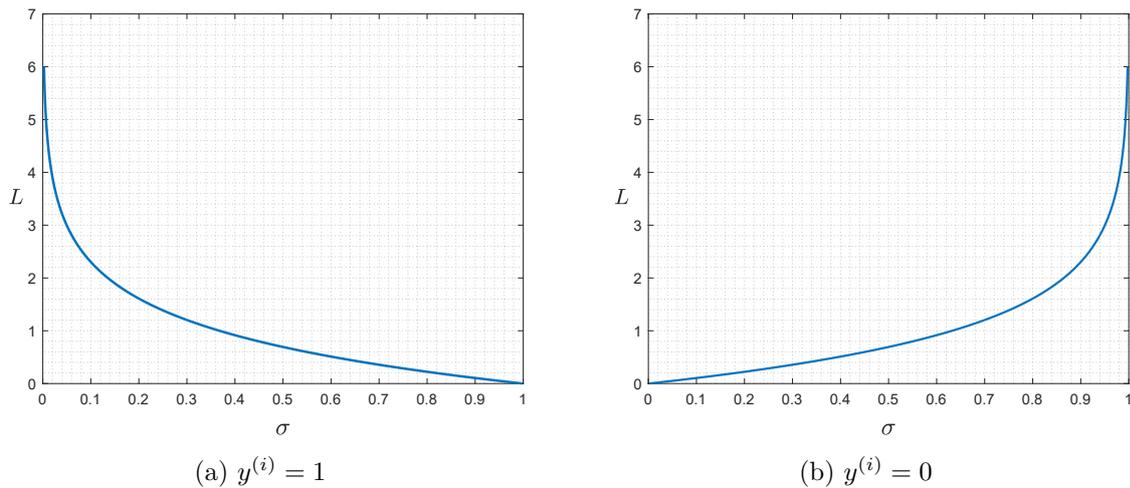


Figura 5.5: Loss function per i dati delle due classi.

Come si vede dalla Figura 5.5 se l'incertezza sulla classificazione del dato da parte di σ è bassa, più la loss function è piccola; viceversa, al crescere dell'incertezza cresce l'errore commesso.

La funzione di perdita così definita ammette un unico minimo, globale, ossia è convessa; in questo modo è possibile ottimizzarla con il metodo del gradiente.

Il neurone di una rete neurale che esegue una regressione di questo tipo si può rappresentare come in Figura 5.6.

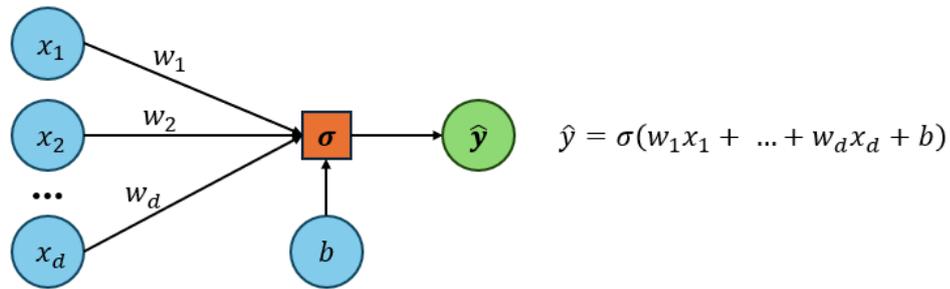


Figura 5.6: Schema del neurone che esegue la regressione logistica a partire da d input.

5.2 Assemblaggio di una rete neurale

Come già detto visto in precedenza la regressione lineare e non lineare costituiscono la base per le reti neurali più complesse. Nelle Figure 5.2, 5.3 e 5.6 sono rappresentati schemi di un singolo neurone. Considerando più neuroni e collegando tra loro input e output si possono creare delle vere e proprie reti. Nell'immagine seguente è rappresentata una rete *feedforward* in cui l'informazione va da sinistra verso destra, senza retroazioni (in questo modo l'output \hat{y} dipende solo dall'input).

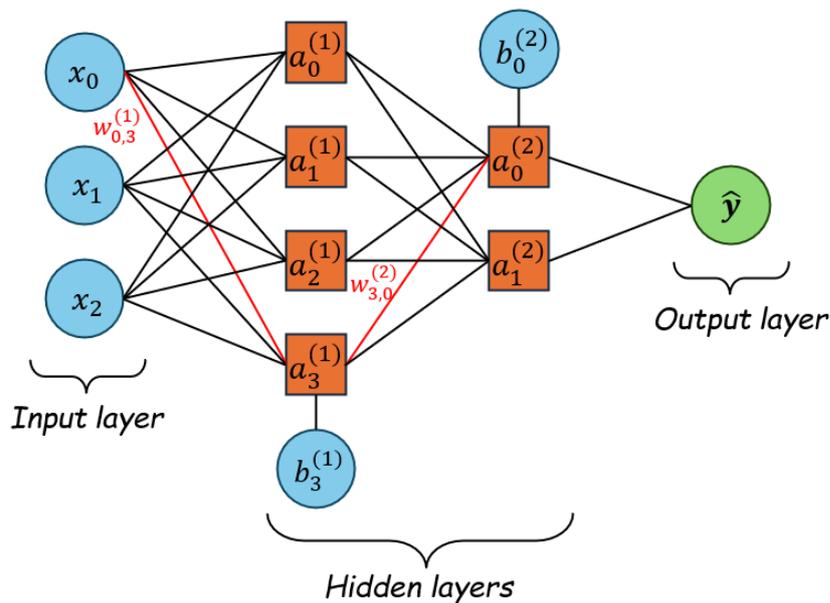


Figura 5.7: Schema di una rete neurale profonda

La rete viene detta *Deep Neural Network* DNN poiché è composta da un layer di input, uno di output e più layer interni, in questo caso due. Si possono utilizzare diverse funzioni per la regressione non lineare, nella raffigurazione è indicata una generica funzione a invece della sigmoide σ . Si forniranno maggiori dettagli a riguardo di ciò in 5.4

5.2.1 Notazione matematica

Per implementare il funzionamento della rete in un codice numero è utile riscrivere le relazioni che la descrivono con delle equazioni matematiche in forma matriciale. Per fare ciò, di seguito si utilizza la notazione di TENSORFLOW, cioè la libreria open source scritta in linguaggio PYTHON.

Il layer di input e i bias si esprimono come vettori riga

$$\mathbf{x} = \{x_0, x_1, \dots, x_{N-1}\} \in \mathbb{R}^{1 \times N} \quad (5.9)$$

$$\mathbf{b}^{(\ell)} = \{b_0^{(\ell)}, b_1^{(\ell)}, \dots, b_{n_\ell-1}^{(\ell)}\} \in \mathbb{R}^{1 \times n_\ell} \quad (5.10)$$

in cui si sono considerati N input, e n_ℓ neuroni nel layer ℓ -esimo.

I pesi si esprimono sotto forma di matrice

$$\mathbf{w}^{(\ell)} = \begin{bmatrix} w_{0,0}^{(\ell)} & \dots & w_{0,n_{\ell-1}}^{(\ell)} \\ w_{1,0}^{(\ell)} & \dots & w_{1,n_{\ell-1}}^{(\ell)} \\ \vdots & \ddots & \vdots \\ w_{n_{\ell-1}-1,0}^{(\ell)} & \dots & w_{n_{\ell-1}-1,n_{\ell-1}}^{(\ell)} \end{bmatrix} \in \mathbb{R}^{n_{\ell-1} \times n_\ell}$$

dove il generico elemento $w_{i,j}^{(\ell)}$ indica il peso dell'output del nodo i del layer $\ell - 1$ che arriva in input al nodo j del layer ℓ .

Prendendo come esempio la Figura 5.7 le relazioni per il calcolo di \hat{y} sono:

$$\mathbf{x} = \{x_0, x_1, x_2\}$$

$$\mathbf{b}^{(1)} = \{0, 0, 0, b_3^{(1)}\}$$

$$\mathbf{w}^{(1)} = \begin{bmatrix} w_{0,0}^{(1)} & w_{0,1}^{(1)} & w_{0,2}^{(1)} & w_{0,3}^{(1)} \\ w_{1,0}^{(1)} & w_{1,1}^{(1)} & w_{1,2}^{(1)} & w_{1,3}^{(1)} \\ w_{2,0}^{(1)} & w_{2,1}^{(1)} & w_{2,2}^{(1)} & w_{2,3}^{(1)} \end{bmatrix}$$

Gli output del primo layer interno sono dunque:

$$\mathbf{a}^{(1)} = \mathbf{x} \cdot \mathbf{w} + \left(\mathbf{b}^{(1)}\right)^T$$

Esplicitando l'ultima equazione si ha

$$\begin{cases} a_0^{(1)} = w_{0,0}^{(1)}x_0 + w_{1,0}^{(1)}x_1 + w_{2,0}^{(1)}x_2 \\ a_1^{(1)} = w_{0,1}^{(1)}x_0 + w_{1,1}^{(1)}x_1 + w_{2,1}^{(1)}x_2 \\ a_2^{(1)} = w_{0,2}^{(1)}x_0 + w_{1,2}^{(1)}x_1 + w_{2,2}^{(1)}x_2 \\ a_3^{(1)} = w_{0,3}^{(1)}x_0 + w_{1,3}^{(1)}x_1 + w_{2,3}^{(1)}x_2 + b_3^{(1)} \end{cases} \quad (5.11)$$

Per il secondo layer interno, considerando che ora gli input corrispondono agli output del layer precedente, e che

$$*b^{(1)} = \{b_0^{(2)}, 0\}$$

si ha

$$\mathbf{a}^{(2)} = \left(\mathbf{a}^{(1)}\right)^T \cdot \mathbf{w} + \left(\mathbf{b}^{(2)}\right)^T$$

Esplicitando nuovamente l'ultima equazione

$$\begin{cases} a_0^{(2)} = w_{0,0}^{(2)}a_0^{(1)} + w_{1,0}^{(2)}a_1^{(1)} + w_{2,0}^{(2)}a_2^{(1)} + w_{3,0}^{(2)}a_3^{(1)} + b_0^{(2)} \\ a_1^{(2)} = w_{0,1}^{(2)}a_0^{(1)} + w_{1,1}^{(2)}a_1^{(1)} + w_{2,1}^{(2)}a_2^{(1)} + w_{3,1}^{(2)}a_3^{(1)} \end{cases} \quad (5.12)$$

Il valore calcolato dalla rete neurale che dovrà approssimare il generico dato della popolazione $y^{(i)}$ è dunque:

$$\hat{y}^{(i)} = w_{0,out}^{(3)}a_0^{(2)} + w_{1,out}^{(3)}a_1^{(2)} \quad (5.13)$$

È facile notare come il problema si sia complicato molto rispetto a (5.3); l'output ora è dato dalla somma di funzioni composte non lineari.

5.3 Algoritmo di forward & backward propagation

Con il termine *forward propagation* si intende il processo di calcolo e memorizzazione delle variabili interne alla rete in modo ordinato dal layer di input a quello di output (da sinistra a destra nella Figura 5.8), in modo da poter valutare la loss function.

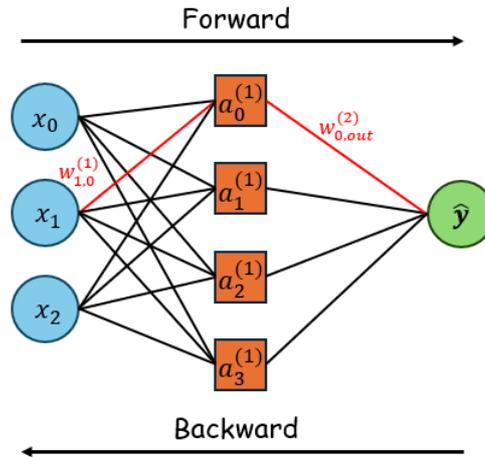


Figura 5.8: DNN ad un singolo layer nascosto

Ad esempio, considerando un problema semplice con un singolo layer nascosto come quello in Figura 5.8, un tipico algoritmo di forward propagation può essere il seguente: dato il layer di input $\mathbf{x} \in \mathbb{R}^3$, le variabili intermedie memorizzate corrispondono alle variabili del layer interno \mathbf{h} :

$$\mathbf{h} = a \left(\mathbf{w}^{(1)} \cdot \mathbf{x} \right)$$

dove $\mathbf{w}^{(1)} \in \mathbb{R}^{3 \times 4}$ contiene i pesi del layer di input. L'output della rete sarà invece

$$\hat{y} = \mathbf{w}^{(2)} \cdot \mathbf{h}$$

Noto questo è possibile calcolare la funzione di perdita L , ad esempio con (5.2).

Per *backward propagation* si intende invece il procedimento per il calcolo del gradiente della loss function. Le variabili interne alla rete ora, le derivate parziali rispetto ai pesi, vengono calcolate in modo inverso, cioè dal layer di output a quello di input. Per il calcolo del gradiente, che sarà utilizzato per aggiornare i pesi, come descritto in precedenza 5.1.1, si utilizza la regola della catena.

Per studiare L al variare di uno dei pesi di input, ad esempio il peso $w_{1,0}^{(1)}$, si applica la regola della catena in ordine dall'output verso l'input:

$$\frac{\partial L}{\partial w_{1,0}^{(1)}} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial a} \cdot \frac{\partial a}{\partial w_{1,0}^{(1)}}$$

5.4 Funzioni di attivazione

Nella Figura 5.7 è rappresentata una generica funzione di attivazione a . In precedenza si è descritta la funzione sigmoide ma esistono altre funzioni che sono spesso utilizzate. Le più comuni sono:

- **Funzione Softmax:** generalizzazione della sigmoide che normalizza i k elementi di un vettore \mathbf{z} nell'intervallo $(0, 1)$ e la cui somma è 1.

$$s : \mathbb{R}^k \rightarrow \left\{ z \in \mathbb{R}^k : s(z_i) > 0, \sum_{i=1}^k s(z_i) = 1 \right\}$$

$$s_i(\mathbf{z}) = \frac{e^{z_i}}{\sum_{i=1}^k e^{z_k}}, \quad i = 1, \dots, k$$

Tipicamente viene utilizzata in problemi di classificazione multi-classe non discussi in questo elaborato.

- **Tangente iperbolica:** simile alla funzione sigmoide ma centrata in 0 e limitata nell'intervallo $(-1, 1)$.

$$\tanh = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

- **ReLU:** Rectified Linear Unit, grazie alla sua semplicità e affidabilità è una tra le più utilizzate.

$$\text{ReLU} = \max\{0, x\}$$

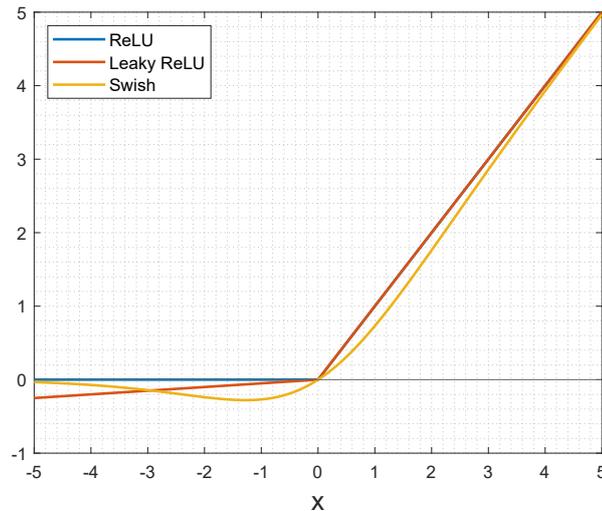


Figura 5.9: ReLU, Leaky ReLU e Swish

La funzione restituisce un valore nullo se l'input è negativo, viceversa, se l'input è positivo, lo ritrasmette come output: il suo unico compito è quello di spegnere i neuroni che ricevono un input negativo. Questo garantisce anche una maggiore efficienza dal punto di vista del costo computazionale poiché alcuni neuroni potrebbero non essere attivi. Tuttavia potrebbe innescarsi il problema della ReLU morente (*dying ReLU*) che si verifica nel momento in cui alcuni neuroni restano spenti per tutto il corso del calcolo. Per questo motivo

sono state introdotte delle funzioni migliorative come Leaky Relu o PReLU mostrate nella Figura 5.9.

Un'ulteriore osservazione è da fare a riguardo della non derivabilità della funzione. Poiché la derivata della funzione compare all'interno dell'algoritmo di backpropagation, è importante definire un valore finito per la derivata in $x = 0$. Si sceglie di attribuire a tale derivata il valore nullo, così come è per i valori negativi della funzione.

- **Leaky ReLU:** miglioramento della funzione ReLU capace di risolvere il problema del ReLU morente. Ai valori di input negativi viene restituito un output non nullo ma molto piccolo.

$$\text{Leaky ReLU} = \begin{cases} \alpha x, & x \leq 0 \\ x, & x > 0 \end{cases}$$

dove α fornisce la pendenza della retta per x negativi ed è un parametro stabilito a priori.

- **PReLU:** si differenzia da Leaky Relu in quanto α viene ora stabilito durante il training della rete.
- **Swish:** si ottiene moltiplicando la funzione sigmoide per x

$$\text{Swish} = x \cdot \sigma = \frac{x}{1 + e^{-x}}$$

Per $x \rightarrow -\infty$ e per $x \rightarrow +\infty$ la funzione si avvicina alla Relu. Tuttavia presenta due importanti differenze rispetto ad essa: non è monotona ed è continua in $x = 0$. Si pensa che queste siano queste due proprietà a garantire risultati migliori nelle reti profonde. Maggiori dettagli a riguardo della funzione Swish si possono trovare in [20].

In generale, si può dire che le funzioni di attivazione inseriscono la non linearità nel problema, il che aiuta la rete a riconoscere pattern più complessi, che una semplice regressione lineare non riuscirebbe a rilevare. In questo modo si riesce a soddisfare il **Teorema di approssimazione universale**, secondo cui una rete neurale artificiale è in grado di riconoscere una qualunque funzione non lineare e continua [21].

La scelta della funzione di attivazione è cruciale per il funzionamento della rete e può avere un impatto significativo sulle sue prestazioni. Non esiste una regola generale per la scelta ma deve essere fatta tenendo conto di vari aspetti come:

- il tipo di problema (se di classificazione o regressione ad esempio)
- il tipo di dati
- la profondità della rete
- il codominio delle funzioni di attivazione
- la velocità di apprendimento
- il problema del *vanishing/exploding gradient*

In particolare, il problema del *vanishing gradient* si verifica nel momento in cui le derivate delle funzioni di attivazione sono così piccole che durante l'algoritmo di backpropagation non si riesce ad aggiornare i pesi a sufficienza. Di conseguenza si avrà una convergenza lenta o assente. Questo è tipico per funzioni come *tanh* o la sigmoide dove per input elevati (in modulo) vanno in saturazione.

Al contrario, l'*exploding gradient* si verifica nel momento in cui le derivate delle funzioni di attivazione sono così grandi che le variazioni sui pesi sono eccessive. La rete in questo caso diventa instabile.

Per mitigare questi due effetti è importante scegliere la corretta funzione di attivazione.

5.5 Underfitting & overfitting

In un caso generico di regressione lineare o non lineare, per stimare la bontà della rete neurale che si sta costruendo è necessario valutare l'errore, calcolato con la funzione di perdita, non sui dati di training, ma su dei dati di test che il modello non ha mai visto. Infatti, lo scopo ultimo della rete è quello di determinare la relazione che lega un set di dati qualsiasi. L'errore sulla previsione di questi dati è definito come **errore di generalizzazione** e dipende da come è stato eseguito il training.

Come messo in evidenza da [22] il modello di rete può essere troppo semplice o troppo complesso e, a seconda di questo, si può avere un errore di generalizzazione più o meno grande. Questo viene mostrato nelle due immagini di seguito.

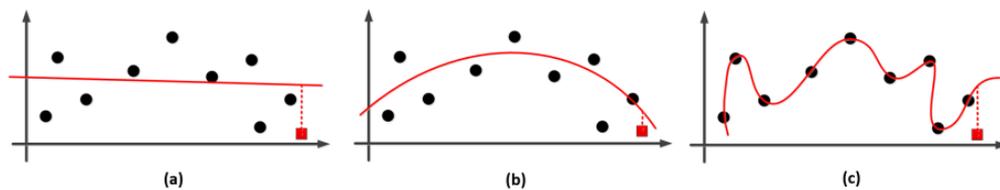


Figura 5.10: Previsioni di tre diversi modelli di rete. I pallini neri sono dati di training, i quadrati rossi sono dati di test

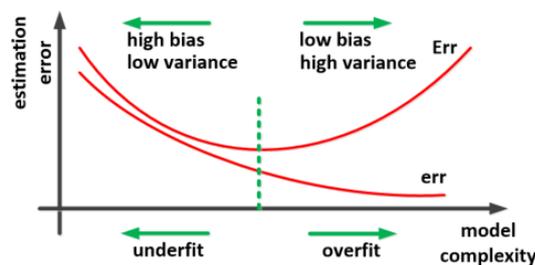


Figura 5.11: Errori del modello al variare della sua complessità

- (a) **Underfitting**: il modello contiene pochi parametri, cioè è troppo semplice. In questo caso si ha un elevato errore (*bias*);
- (b) Fitting adeguato: viene rappresentato sufficientemente bene il set di dati e l'errore di generalizzazione *Err* è minimo;
- (c) **Overfitting**: il modello contiene troppi parametri, dunque è troppo complicato. Il modello ha ora una elevata varianza (*variance*).

Dalla Figura 5.11 si trova conferma che all'aumentare della complessità del modello l'errore sul set di training, indicato come *err*, diminuisce.

5.6 Altri tipi di rete

Quanto discusso fin'ora sono concetti basilari su qualunque tipo di rete neurale artificiale. Ora verranno presentate due tipologie di reti diverse tra loro ma molto interessanti e comuni nelle applicazioni al giorno d'oggi.

5.6.1 Convolutional Neural Networks - CNNs

Grazie agli studi condotti da Hubel e Wiesel negli anni '50 si scoprì che molti neuroni nella corteccia visiva vengono stimolati solamente da una regione molto ristretta all'interno di tutto il campo visivo e che le diverse regioni dei neuroni si sovrappongono fino a ricoprire tutta la scena visiva [23]. Venne dimostrato anche che alcuni neuroni rispondono se nel loro campo ricettivo ci sono linee orizzontali, mentre altri rispondono a linee con altra inclinazione. Inoltre, notarono che alcuni neuroni hanno una regione ricettiva più ampia e che li consente di reagire a pattern più complicati dati anche dalla combinazione di pattern più semplici (come la direzione di linee). Queste osservazioni portarono all'idea che i neuroni sono solamente collegati ad altri neuroni nelle loro strette vicinanze. In Figura 5.12 è mostrato un esempio di come il singolo neurone di un layer non sia collegato a tutti i neuroni dei layers adiacenti, ma solo ad alcuni neuroni nelle sue vicinanze, e di come i campi ricettivi dei neuroni possano sovrapporsi [24].

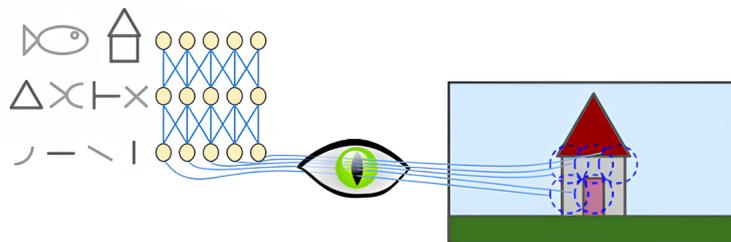


Figura 5.12: Campi ricettivi nella corteccia visiva del cervello.

Le CNNs basano il loro funzionamento sul comportamento dei neuroni nella corteccia visiva e utilizzano implicitamente degli input bidimensionali. Questa è una prima caratteristica

che li differenzia dalle reti feedforward (oltre ai neuroni *fully connected* caratteristici solo di quest'ultima tipologia di rete).

Architettura di una CNN

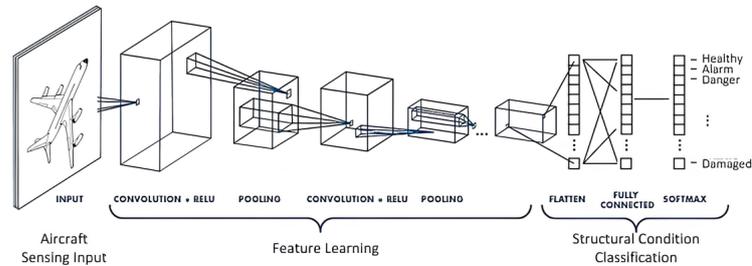


Figura 5.13: Architettura di una CNN impiegata per monitorare lo stato di salute della struttura di un velivolo.

Le reti neurali convoluzionali sono formate dai seguenti blocchi principali: layer convoluzionale in cui si estraggono le principali features dell'input mediante l'applicazione di filtri, pooling layer, una rete neurale fully connected e infine l'output [25].

- **Filtri:** sono delle matrici addestrate per riconoscere alcune feature nell'input. Vengono applicate mediante l'operazione di convoluzione, che consiste nel moltiplicare scalarmente una porzione dell'input per il filtro stesso. L'operazione viene ripetuta fino a ricoprire tutto l'input, così da avere un input filtrato. Le dimensioni dei filtri e il numero di questi sono parametri da assegnare.

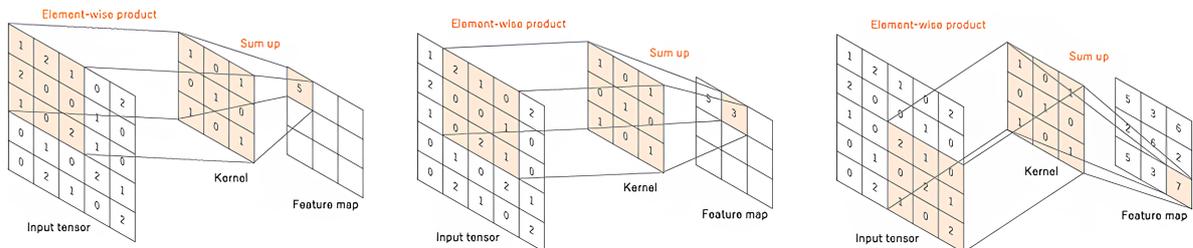


Figura 5.14: Operazione di convoluzione tra la matrice di input e il filtro.

- **Layer convoluzionale:** una volta che si sono applicati tutti i filtri all'input si ottengono una serie di mappe, una per filtro, contenenti le principali feature dell'input originario. Queste possono subire un'ulteriore trasformazione mediante l'applicazione di funzioni non lineari, tipicamente ReLU. Si ottiene in output da questo processo un volume di neuroni le cui dimensioni dipendono da quelle dell'input originario, da quelle del filtro e dal numero di filtri applicati. La creazione di questo volume è la fase più cruciale di tutto il processo di funzionamento delle CNNs. È possibile agire su alcuni iperparametri in questa fase:

- dimensione del filtro: determina il campo ricettivo dei neuroni che non sono fully connected, come spiegato in precedenza
- **stride**: distanza tra due posizioni successive del filtro durante la convoluzione. Tipicamente pari a 1 ma può essere anche più grande così da ridurre le dimensioni delle feature maps
- **padding**: come mostrato nella Figura 5.14 il centro del filtro (o kernel) non riesce a sovrapporsi agli elementi laterali della matrice di input. Questo fa perdere una parte delle informazioni dell'input e le features maps risultano più piccole dopo ogni convoluzione. Una tecnica che si utilizza è quella di inserire una riga o colonna per ogni bordo. In questo modo le dimensioni dell'input filtrato rimangono invariate (Figura 5.15).

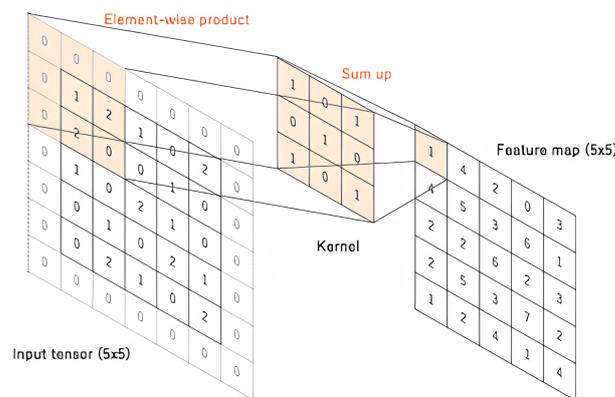


Figura 5.15: Convoluzione con la tecnica di zero-padding applicata all'input.

- **Pooling layer:** applica una riduzione delle dimensioni delle features maps così da avere meno parametri da calcolare, un modello meno complesso e quindi di ridurre il rischio di overfitting. Tipicamente se ne inserisce uno subito a valle di un layer convoluzionale come viene mostrato in Figura 5.13. Una tecnica spesso utilizzata è quella del **max pooling** con un filtro di dimensioni 2×2 e uno stride di 2 che estrae, per ogni passaggio del filtro, il valore massimo dall'input (Figura 5.16) dimezzando le dimensioni originarie.

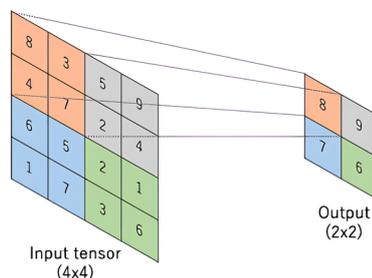


Figura 5.16: Operazione di max pooling con un filtro 2×2 .

- **Flatten layer:** si usa per rendere l'output dei layer convoluzionali un vettore unidimensionale da dare in input alla rete fully connected. La lunghezza del vettore è pari, in numero, al volume dell'ultimo stack di neuroni che ha subito il pooling.
- **Fully connected layer o dense layer:** una volta che le principali features dell'input sono state estratte dalla rete convoluzionale è necessario prendere decisioni o classificare l'input. Questo viene gestito da una rete neurale densa che, sulla base delle features apprese, collega ogni neurone permettendo una combinazione completa.

5.6.2 Physics-Informed Neural Networks - PINNs

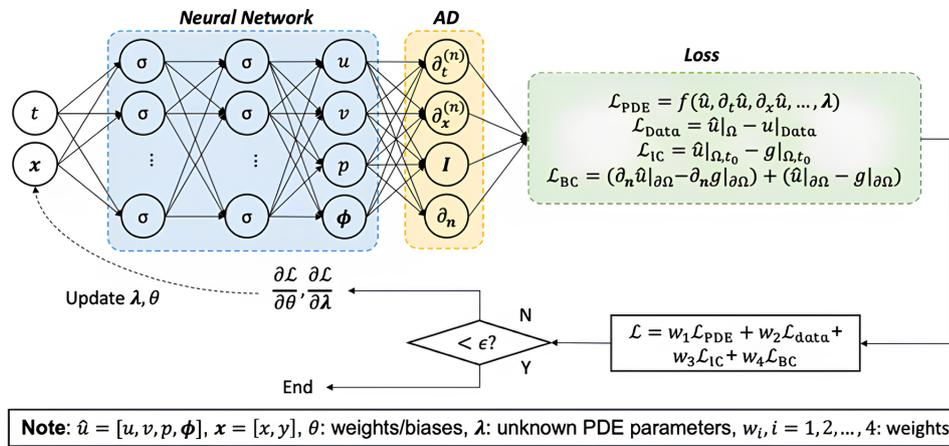


Figura 5.17: Architettura di una PINN

Questo tipo di rete neurale pone il suo funzionamento sulla validità del teorema di approssimazione universale [21]. L'obiettivo è, nel caso più complicato, quello di approssimare la soluzione di un sistema di equazioni differenziali non lineari introducendo le equazioni stesse all'interno della rete.

L'architettura delle PINN, mostrata in Figura 5.17, prevede l'utilizzo combinato di due reti: una prima feedforward fully connected che calcola la soluzione approssimata del sistema di equazioni differenziali, e una seconda rete che prende l'output della prima e calcola il residuo, cioè la loss function, implementando le equazioni di governo del problema fisico e il calcolo differenziale tramite la regola della catena. In questo modo il calcolo della derivata è esatto, e non viene approssimata con metodi numerici che introducono errori di troncamento e di discretizzazione. Questo le differenzia molto dai codici numerici tradizionali che si basano proprio sull'utilizzo di schemi diversi per l'approssimazione di derivate ed integrali. Inoltre, i codici numerici hanno bisogno di una mesh spaziale, e di un rate temporale in cui calcolare la soluzione, mentre quella ottenuta con le PINNs può spaziare in ogni punto del dominio fisico e in quello temporale.

La loss function, chiamata anche residuo in questo caso, si può pensare come data da diversi

contributi, [26]:

$$L = \theta_1 L_{PDE} + \theta_2 L_{data} + \theta_3 L_{IC} + \theta_4 L_{BC}$$

in cui L_{PDE} calcola la differenza tra il risultato ottenuto con la PINN e il risultato atteso, L_{data} misura la differenza tra i valori predetti e quelli di training, L_{IC} e L_{BC} misurano invece l'errore commesso sulle condizioni iniziali e al bordo, e θ_i con $i = 1, \dots, i = 4$ sono i pesi dei rispettivi errori.

Una volta che è stata valutata la loss function si controlla se questa soddisfa la condizione di essere inferiore rispetto ad una certa tolleranza: in caso positivo allora si accetta la soluzione calcolata dalla PINN, ma in caso negativo è necessario aggiornare i pesi della rete e iterare il processo. In questo modo il problema di risoluzione del sistema di PDE si trasforma in un problema di ottimizzazione della funzione di perdita.

È da notare che secondo questo approccio le PINNs sono **non supervisionate** in quanto non hanno bisogno di dati già noti da simulazioni o esperimenti (non effettuano alcuna regressione). È però possibile utilizzarle anche in modo **supervisionato**, cioè sfruttando dati già noti, ad esempio nel caso in cui si voglia risolvere un problema mal definito in cui manca una condizione al bordo o un'equazione di stato da un set di equazioni differenziali alle derivate parziali [27].

Capitolo 6

Risultati preliminari

Si prende in esame la trave a sezione quadrata incastrata in $y = 0$ e caricata in $y = L$ in Figura 6.1, già studiata in [28].

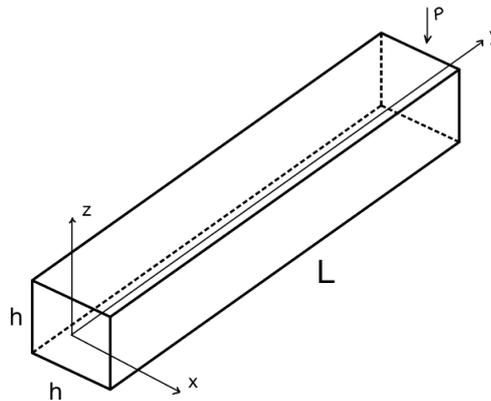


Figura 6.1: Trave analizzata

Le dimensioni caratteristiche sono $h = 0.20 \text{ m}$ e $L = 2 \text{ m}$, mentre il materiale isotropo è definito da $E = 75 \text{ GPa}$, $\nu = 0.33$, $\rho = 2700 \text{ kg/m}^3$. La forza applicata nel baricentro della sezione è $\mathbf{P} = -50\hat{\mathbf{k}} \text{ N}$.

6.1 Analisi statica

Si valuta lo spostamento u_z dell'estremo libero con diverse approssimazioni.

Eulero-Bernoulli

L'equazione della linea elastica si può scrivere come

$$u_z(y) = \frac{Py^2}{6EI_x} (y - 3L)$$

e valutato all'estremo libero, in corrispondenza di $y = L$ si ha uno spostamento pari a

$$u_z(L) = -1.3333 \cdot 10^{-5} \text{ m}$$

Timoshenko

In questo caso l'equazione della linea elastica si può scrivere come

$$u_z(y) = -\frac{Py}{\kappa GA} - \frac{PL}{EI_x} \left(\frac{y^2}{2L} - \frac{y^3}{6L} \right)$$

dove $\kappa = 0.8276$, in accordo con [29], è il coefficiente correttivo del taglio, il modulo di taglio G per un materiale isotropo è

$$G = \frac{E}{2(1 + \nu)} = 28.20 \text{ GPa}$$

e $A = h^2 = 0.04 \text{ m}^2$ è la sezione trasversale della trave.

Lo spostamento all'estremo libero, in corrispondenza di $y = L$ è pari a

$$u_z(L) = -1.3331 \cdot 10^{-5} \text{ m}$$

Codice MUL2

Si utilizzano diversi modelli per descrivere gli spostamenti, tra cui EBBT, TBT, Taylor e Lagrange. I risultati saranno confrontati con [28].

La sezione della trave, nel piano xz , viene descritta con un elemento L9, cioè a 9 nodi disposti come nella figura di seguito.

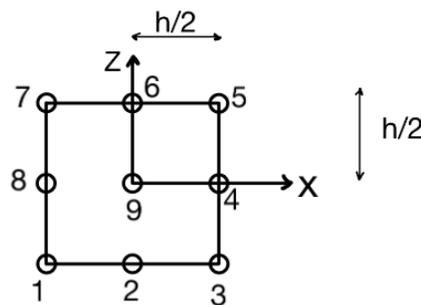


Figura 6.2: Elemento L9

e il carico è applicato al nodo 9. Solo nel caso LE4 si utilizzano gli elementi L4, e il carico in questo caso viene diviso come in figura.

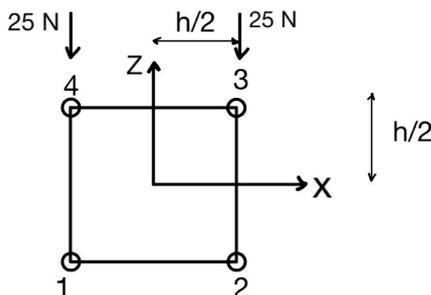


Figura 6.3: Elemento L4

I risultati per lo spostamento nel punto di applicazione della forza, cioè in $(x, y, z) = (0, L, 0)$ sono riportati in tabella.

Elemento	N elementi	EBBT Eulero-Bernoulli	TBT Timoshenko	TAYLOR				LAGRANGE	
				TE1	TE2	TE3	TE4	LE4	LE9
B2	2	-1.250	-1.259	-1.259	-1.073	-1.075	-1.075	-0.967	-1.073
	10	-1.330	-1.339	-1.339	-1.293	-1.295	-1.295	-1.099	-1.293
	40	-1.333	-1.333	-1.342	-1.324	-1.326	-1.327	-1.114	-1.325
B3	2	-1.333	-1.342	-1.342	-1.252	-1.254	-1.254	-1.082	-1.252
	10	-1.333	-1.342	-1.342	-1.320	-1.322	-1.323	-1.113	-1.320
	40	-1.3333	-1.3422	-1.3422	-1.3276	-1.3295	-1.3308	-1.1152	-1.3280
B4	2	-1.333	-1.342	-1.342	-1.293	-1.296	-1.296	-1.101	-1.294
	10	-1.333	-1.342	-1.342	-1.325	-1.327	-1.328	-1.115	-1.326
	40	-1.333	-1.342	-1.342	-1.328	-1.330	-1.332	-1.115	-1.329

Tabella 6.1: Spostamento all'estremo libero u_z in $10^{-5} m$.

I risultati sono stati ottenuti con l'integrazione SELI attiva, così da evitare gli errori numerici che si verificano con l'utilizzo di elementi B2.

Si sono utilizzati elementi a 2, 3 e 4 nodi (rispettivamente B2, B3 e B4). Considerando i risultati analitici come riferimento, per ciascuna teoria si vede un miglioramento dei risultati all'aumentare del numero di elementi utilizzati per descrivere la trave, e al numero di nodi di ciascun elemento. Inoltre, l'approssimazione con il modello di Taylor migliora all'aumentare dell'ordine di accuratezza (da TE1 a TE4).

Il modello di Lagrange descrive il campo di spostamenti come

$$u_z(x, y, z) = \sum_{i=1}^N F_i u_{z_i}(x, y, z)$$

dove è il N numero di nodi nella sezione. Con una sezione a 4 nodi (LE4) non viene descritto accuratamente lo spostamento, al contrario di LE9.

Capitolo 6. Risultati preliminari

Di seguito vengono riportati anche i DOFs di ciascun modello utilizzato, calcolati con la formula:

$$DOFs = M [(N_{NE} - 1) N_{BE} + 1]$$

dove M è il numero di incognite per ciascun nodo, N_{NE} è il numero di nodi per ciascun elemento e N_{BE} è il numero di elementi beam utilizzati.

Elemento	N elementi	EBBT Eulero-Bernoulli	TBT Timoshenko	TAYLOR				LAGRANGE	
				TE1	TE2	TE3	TE4	LE4	LE9
B2	2	9	15	27	54	90	135	36	81
	10	33	55	99	198	330	495	132	297
	40	123	205	369	738	1230	1845	492	1107
B3	2	15	25	45	90	150	225	60	135
	10	63	105	189	378	630	945	252	567
	40	243	405	729	1458	2430	3645	972	2187
B4	2	21	35	63	126	210	315	84	189
	10	93	155	279	558	930	1395	372	837
	40	363	605	1089	2178	3630	5445	1452	3267

Tabella 6.2: Gradi di libertà della trave discretizzata secondo le varie teorie di approssimazione.

Dalla Tabella 6.2 si osserva come le incognite aumentano all'aumentare del numero di nodi e di elementi utilizzati. I modelli più accurati, come quelli di Taylor al terzo e quarto ordine o il modello di Lagrange a 9 nodi, hanno un elevato costo computazionale.

Infine, si riportano gli stress σ_y valutati all'incastro, nel punto $(x, y, z) = (0, 0, h/2)$.

Elemento	N elementi	EBBT Eulero-Bernoulli	TBT Timoshenko	TAYLOR				LAGRANGE	
				TE1	TE2	TE3	TE4	LE4	LE9
B2	2	56244	56244	56244	67963	68146	68146	30764	67921
	10	71243	71243	71243	90185	90744	90749	39848	89356
	40	74055	74055	74055	91006	92155	92644	40182	87023
B3	2	74992	74992	74992	90126	90587	90579	40895	89775
	10	74992	74992	74992	88272	89206	89319	40063	84559
	40	74660	74660	74660	86052	88520	90453	38915	79120
B4	2	75690	75690	75690	86653	87350	87329	40187	85488
	10	74406	74406	74406	85460	86989	87627	38958	79271
	40	74549	74549	74549	84779	88965	90747	38743	77559

Tabella 6.3: Stress σ_y in Pa all'estremo vincolato.

Se si confrontano i risultati della Tabella 6.3 con quanto calcolato con la formula di Navier, che in questo caso si riduce ad essere

$$\sigma_y = \frac{PL}{I_x}z = 75000 \text{ Pa},$$

allora si nota che i risultati per EBBT, TBT, TE1 e LE9 si avvicinano molto. I modelli di Taylor più onerosi invece, nonostante forniscano un'approssimazione accurata dello spostamento, sovrastimano lo stress.

Si riportano di seguito due immagini da cui è possibile osservare la struttura deformata, in seguito al carico applicato, e la distribuzione degli stress lungo la trave. Queste sono state ottenute con una discretizzazione a 10 elementi B4, e un calcolo degli spostamenti con TE4.

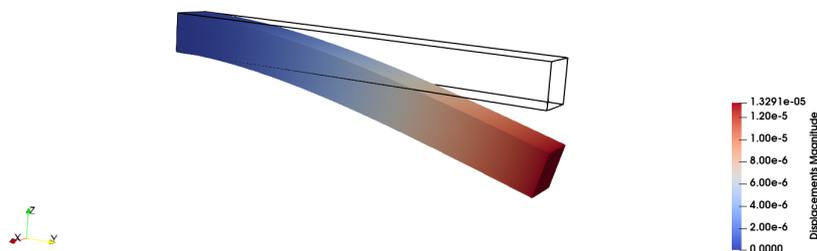


Figura 6.4: Trave deformata. In seguito al carico applicato la trave si inflette. La freccia massima si riscontra nell'estremità libera, mentre all'incastro lo spostamento è nullo.

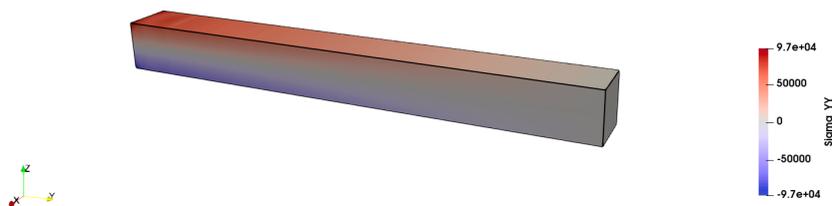


Figura 6.5: Distribuzione dello sforzo σ_y sulla trave. Dalla colorazione si può confermare l'andamento a farfalla dello stress.

La distribuzione dello sforzo lungo la trave, mostrata in Figura 6.5 rispetta la formula di Navier, infatti, la superficie superiore, a $z = +h/2$, è tesa e vede dunque una $\sigma_y > 0$, mentre la superficie inferiore, a $z = -h/2$, è compressa e vede $\sigma_y < 0$. È evidenziata anche la superficie neutra, che data la geometria della struttura sarà a $z = 0$. Si può osservare ancora come lo sforzo massimo sia all'incastro, mentre verso l'estremità libera è via via inferiore fino ad annullarsi.

6.2 Analisi dinamica

Si considera la trave discretizzata con 10 elementi B4. Si eseguono diverse analisi dinamiche cambiando di volta in volta le teorie di approssimazione, così da poter mettere a confronto le frequenze proprie e i modi vibrazionali.

Nella Tabella 6.4 seguente sono riportate le frequenze dei primi 10 modi vibrazionali calcolate con le varie teorie di approssimazione.

Modo	EBBT	TBT	TE1	TE2	TE3	TE4	LE4	LE9
1	42.49	42.27	42.27	42.63	42.58	42.57	46.44	42.62
2	42.49	42.27	42.27	42.63	42.58	42.57	46.44	42.62
3	263.30	254.60	254.60	256.58	255.16	255.08	403.94	256.49
4	263.30	254.60	254.60	256.58	255.16	255.08	278.05	256.49
5	658.82	658.82	403.94	403.94	403.94	372.09	278.05	403.94
6	724.37	673.83	478.47	661.88	661.76	661.67	662.18	661.88
7	724.37	673.83	673.83	678.58	671.31	671.05	730.57	678.23
8	1384.8	1230.6	673.83	678.58	671.31	671.05	730.57	678.23
9	1384.8	1230.6	1211.9	1211.9	1211.9	1116.5	1211.9	1211.9
10	1976.7	1885.1	1230.6	1238.4	1218.5	1217.8	1323.8	1237.4

Tabella 6.4: Frequenze dei vari modi di vibrare in Hertz

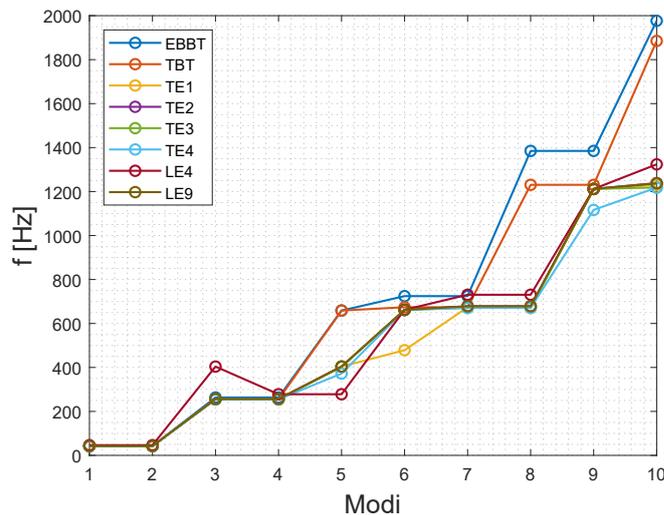
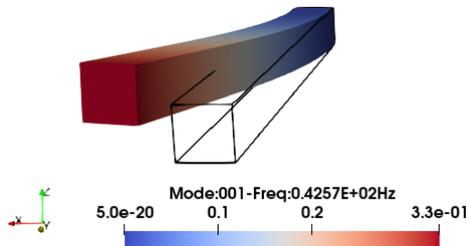
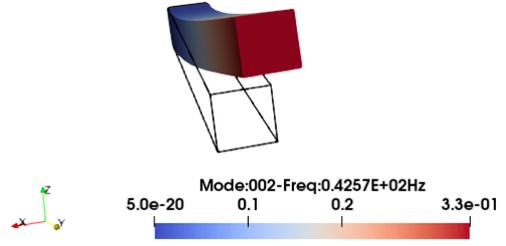


Figura 6.6: Rappresentazione grafica delle frequenze calcolate con i vari modelli

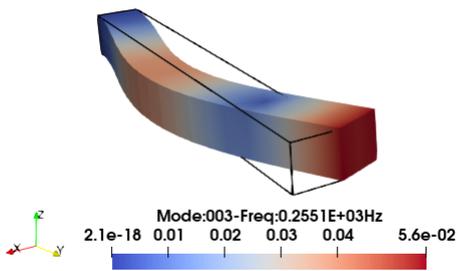
I modi di vibrare sono stati visualizzati mediante il software PARAVIEW. Di seguito si riportano i modi calcolati utilizzando un modello TE4.



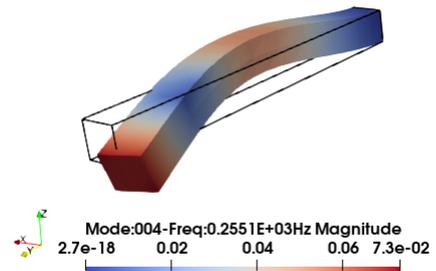
(a) Modo 1



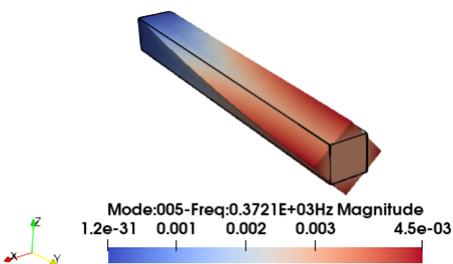
(b) Modo 2



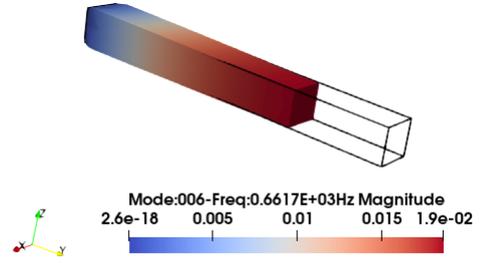
(c) Modo 3



(d) Modo 4



(e) Modo 5



(f) Modo 6

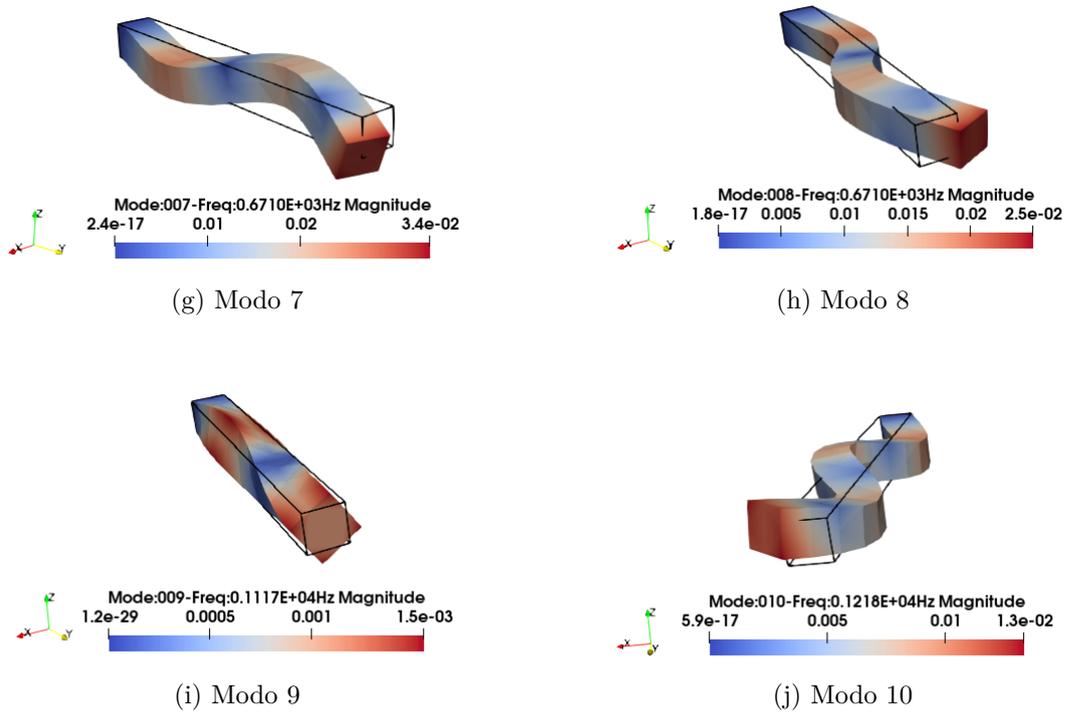


Figura 6.11: Modi vibrazionali della trave.

I modi che si ottengono sono flessionali, per basse frequenze, poi è presente un modo torsionale, che non viene previsto dalle teorie classiche EBBT e TBT, e per frequenze più alte si hanno modi flessionali a più semionde.

6.3 Analisi aeroelastica dinamica: flutter

Per le analisi aeroelastiche si fa riferimento alla piastra isotropa in Figura 6.12, già studiata da Santori in [30],

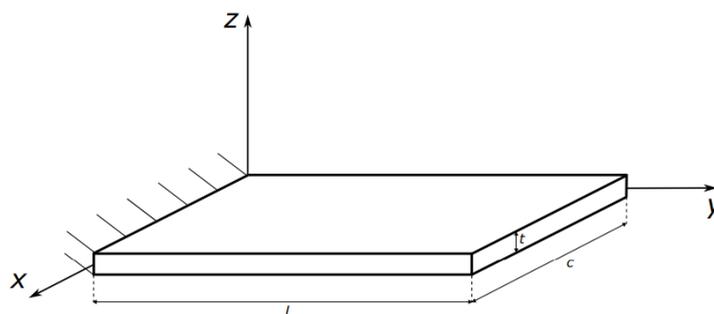


Figura 6.12: Piastra analizzata

le cui dimensioni sono $L = 0.305 \text{ m}$, $c = 0.076 \text{ m}$ e $t = 0.001 \text{ m}$. Il materiale isotropo invece è caratterizzato da $E = 73800 \text{ MPa}$, $G = 27600 \text{ GPa}$ e $\rho = 2768 \text{ kg/m}^3$.

Prima di procedere con l'analisi aeroelastica viene seguita un'analisi dinamica così da studiare come la struttura agisce in un caso dinamico. Si otterranno i primi 10 modi flessionali e le rispettive frequenze.

Si eseguono due analisi con il codice MUL2 utilizzando due teorie accurate per il calcolo degli spostamenti, cioè TE4 e LE9.

Dati i risultati in [30] si decide di discretizzare la piastra con una mesh strutturale data da 20 elementi B4, lungo l'asse y , e 4 elementi L9 nel piano xz . La disposizione di questi elementi, necessaria per il calcolo mediante l'approssimazione di Lagrange, è rappresentata nella Figura 6.13.

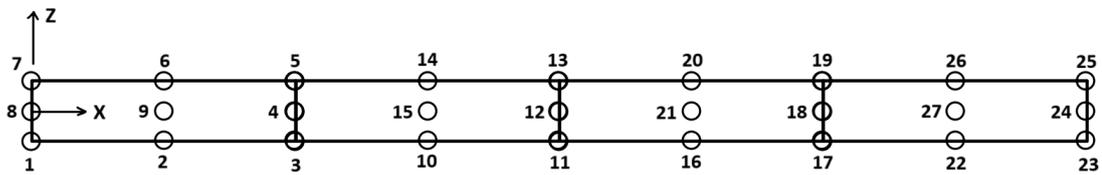


Figura 6.13: Sezione trasversale della piastra schematizzata con 4 elementi L9.

I risultati per l'analisi dinamica sono riportati nella tabella di seguito.

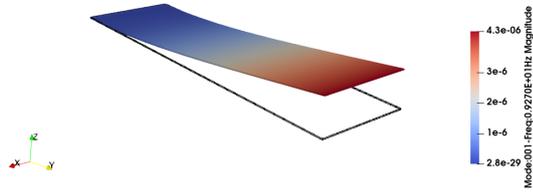
Modo	TE4	4LE9	Bibliografia 4LE9
1	9.27	9.33	9.17
2	57.92	58.35	57.31
3	73.01	72.97	72.96
4	162.61	163.59	161.00
5	226.02	225.80	225.67
6	319.93	321.19	317.06
7	398.74	398.16	397.69
8	529.43	530.48	526.32
9	601.33	600.12	599.16
10	653.61	654.02	653.56

Tabella 6.5: Confronto tra le frequenze, in Hertz, con i risultati bibliografici

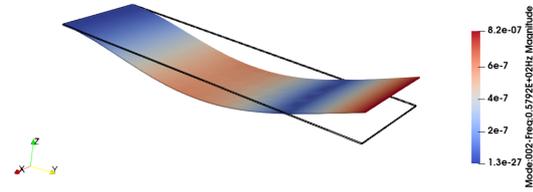
I risultati trovati si avvicinano molto a quelli ottenuti da Santori in [30]. La differenza nei risultati con il modello di Lagrange a quattro elementi è dovuta ad una diversa disposizione di questi nella sezione della piastra.

Capitolo 6. Risultati preliminari

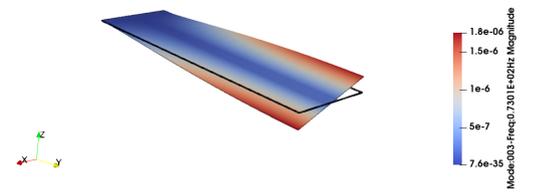
Anche i modi flessionali trovati sono in accordo con quelli riportati in [30]. Di seguito vengono mostrati solo i primi 4 modi, ottenuti utilizzando una teoria TE4 con il codice MUL2.



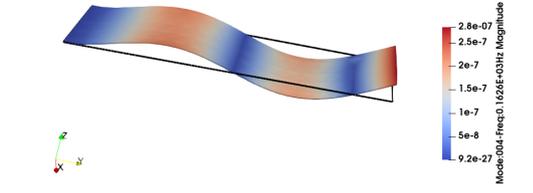
(a) Modo 1



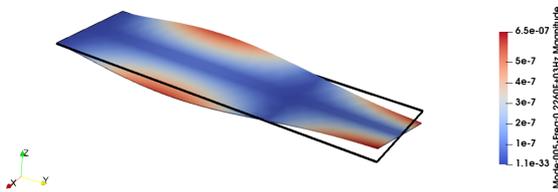
(b) Modo 2



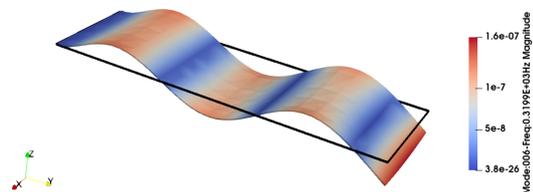
(c) Modo 3



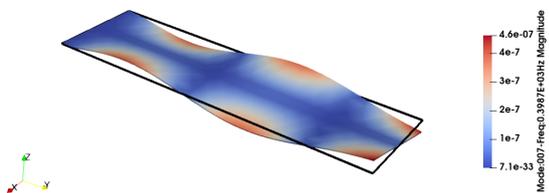
(d) Modo 4



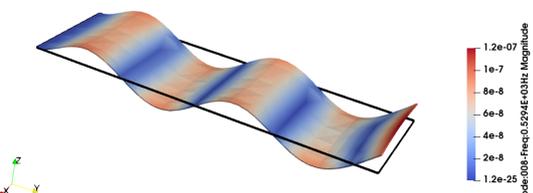
(c) Modo 5



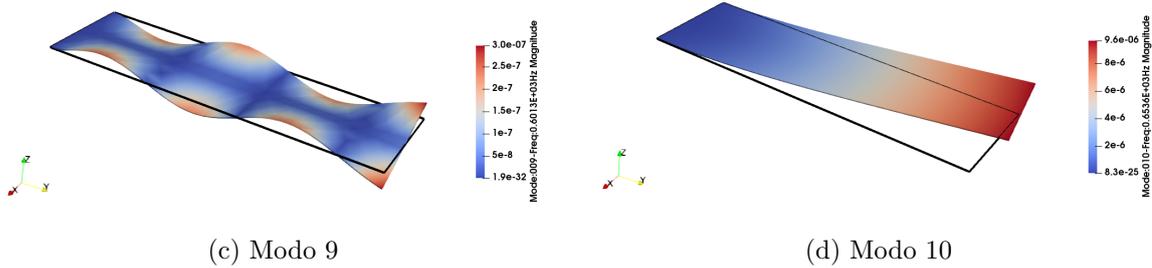
(d) Modo 6



(c) Modo 7



(d) Modo 8



Si può procedere ora con l'analisi aeroelastica.

Per eseguire l'analisi con il codice MUL2 è necessario specificare i parametri aeroelastici nel file di input `NASTRAN_FLUTTER.dat`, tra cui:

- densità di riferimento: $\rho = 1.225 \text{ kg/m}^3$
- simmetria: nel piano xz
- lunghezza della corda di riferimento: $c = 0.076 \text{ m}$.
- coordinate dei punti P1 e P2, dai cui vengono costruite le corde di estremità della piastra (*tip* e *root*)
- dimensione della mesh aerodinamica, in termini di numero di pannelli lungo l'apertura (asse y) e lungo la corda (asse x): 30×8
- range di variazione della densità (mantenuta costante al valore di riferimento)
- range di variazione del numero di Mach (mantenuto e nullo, per studiare un caso incompressibile)
- range di variazione della velocità, fatta variare in 40 intervalli tra 2 m/s e 150 m/s

Oltre a ciò, si aggiungono 50 punti, circa, equispaziati nel file `POSTPROCESSING.dat` al fine di valutare gli spostamenti del modello strutturale e trasferirli al modello aerodinamico. Questi punti sono necessari dunque per la creazione della spline.

Si esegue l'analisi con l'approssimazione di Lagrange a 4 elementi, come in Figura 6.13; si ottiene un file `flutter.f06` da cui si estraggono i risultati in termini di smorzamento e frequenza al variare della velocità nel range prefissato, per ogni modo. Questi sono riportati nei due grafici in Figura 6.19.

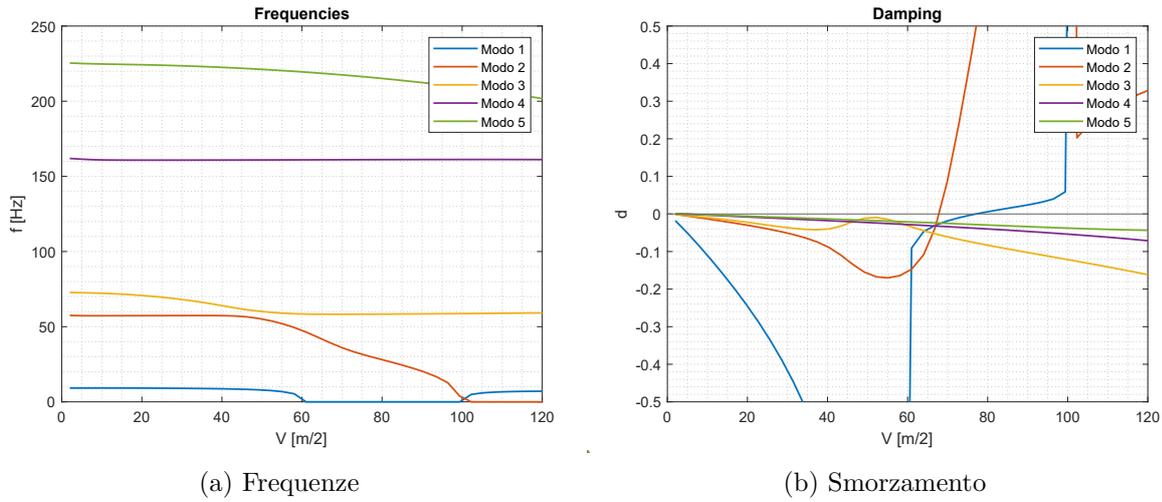


Figura 6.19: Risultati dell'analisi di flutter per i primi 5 modi vibrazionali.

Il primo modo ad andare in instabilità, poiché lo smorzamento da negativo diventa positivo, è il secondo, ad una velocità di $V_F = 67.782 \text{ m/s}$ e una frequenza $f_F = 38.584 \text{ Hz}$.

È possibile individuare anche la condizione di divergenza, caratterizzata da uno smorzamento che da negativo diventa positivo ad una frequenza nulla. Ciò si ottiene per $V_D = 77.149 \text{ m/s}$.

Ripetendo l'analisi con un' approssimazione di Taylor al quarto ordine, cioè TE4, si ottiene sempre l'instabilità dinamica per il secondo modo, e una velocità di divergenza per il primo modo ad una velocità maggiore. I risultati sono riassunti nella Tabella di seguito.

	TE4	4LE9	Bibliografia 4LE9
$V_F \text{ [m/s]}$	67.819	67.782	67.750
$f_F \text{ [Hz]}$	38.495	38.584	38.577
$V_D \text{ [m/s]}$	78.050	77.148	79.154

Tabella 6.6: Confronto tra i risultati delle analisi aeroelastiche con quelli bibliografici.

I risultati ottenuti con le due teorie forniscono risultati in accordo con quelli bibliografici in [30].

6.4 Analisi aeroelastica statica

Si considera sempre la piastra in Figura 6.12. L'obiettivo è quello di valutare la risposta della struttura, in termini di spostamento e rotazione al tip, in $y = L$, a diverse velocità e ad un angolo di attacco costante.

La torsione della struttura viene valutata una volta noti gli spostamenti del bordo di attacco,

punto A, e del bordo di fuga, punto B all'estremo libero, mediante la seguente formula:

$$\vartheta = \arctan\left(\frac{u_{za} - u_{zb}}{c}\right)$$

L'analisi viene condotta considerando 20 elementi B4 lungo l'asse della trave e due modelli diversi per gli spostamenti: TE4 e 4LE9. I risultati saranno confrontati anche con un'analisi eseguita dal codice commerciale per verificarne l'affidabilità.

Nei file input `NASTRAN_FLUTTER.dat` è necessario specificare i parametri aeroelastici, come per l'analisi di flutter, inserendo però un valore velocità, invece che un range, e un angolo di attacco. Si è scelto di eseguire lo studio con $AoA = 1^\circ$. I risultati sono riportati nelle tabelle di seguito.

V [m/s]	u_{za} [mm]	u_{zb} [mm]	ϑ [°]
10	0.725	0.695	0.023
20	0.725	0.695	0.023
30	7.698	7.388	0.238
40	16.08	15.45	0.491
50	32.01	30.76	0.958
60	67.29	64.74	1.963
70	188.0	181.1	5.311
75	477.5	460.2	13.06
78	1968	1898	43.39
80	-2551	-2460	-50.53
85	-422.5	-407.9	-11.14
90	-253.9	-245.2	-6.599

Tabella 6.7: Risultati relativi al modello TE4

V [m/s]	u_{za} [mm]	u_{zb} [mm]	ϑ [°]
0	0.717	0.687	0.023
20	3.05	2.92	0.097
30	7.61	7.30	0.241
40	15.91	15.26	0.498
50	31.75	30.48	0.977
60	67.65	65.00	2.038
70	197.3	189.7	5.778
75	571.6	550.1	16.12
78	6901	6643	73.85
80	-1154	-1112	-29.86
85	-345.8	-333.2	-9.59
90	-219.8	-211.9	-6.02

Tabella 6.8: Risultati relativi al modello 4LE9

Dall'analisi condotta si ottiene una velocità di divergenza compresa tra i 78 m/s e i 80 m/s .

6.5 Confronto con Femap

Poiché si è in assenza di un riferimento bibliografico, gli stress e l'analisi modale sono state confrontate con il codice commerciale FEMAP. Oltre a queste si riporta anche il caso di analisi statica per completezza della trattazione.

La trave a sezione quadrata è stata schematizzata come una superficie nel piano xy dotata di uno spessore nella direzione z pari alla larghezza della trave nella direzione x , cioè 0.20 m . Definito il materiale, è stata creata una mesh sulla superficie con 40 elementi sul lato lungo y , e 10 elementi lungo x , sul lato corto, per un totale di 400 elementi CQUAD.

Alla trave così schematizzata è stato applicato un vincolo di incastro, in corrispondenza dei nodi in $y = 0$, e una forza $\mathbf{F} = -50\hat{\mathbf{k}} N$, in $y = L$, in corrispondenza del nodo centrale. Quest'ultimo coincide proprio con il baricentro della sezione.

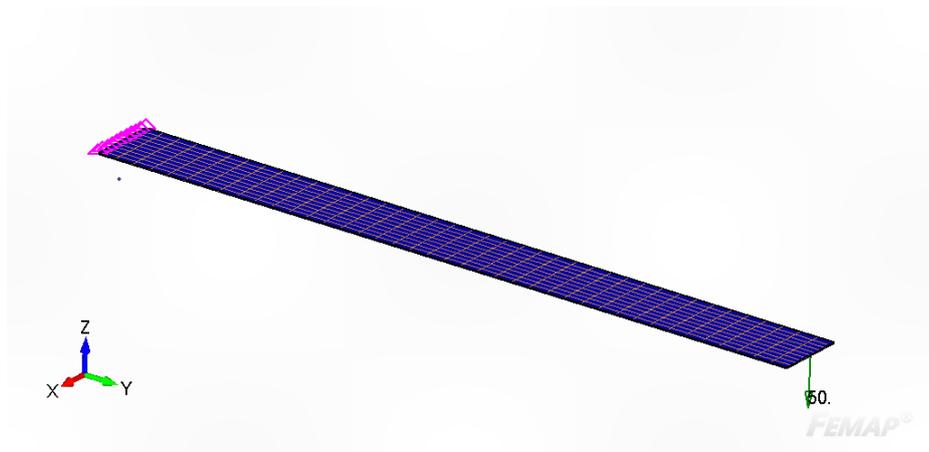


Figura 6.20: Modello della trave su FEMAP

6.5.1 Analisi statica

Si sono valutati lo spostamento all'estremo libero nelle coordinate $(x, y, z) = (0, 2, 0)$, e lo stress all'incastro nelle coordinate $(x, y, z) = (0, 0, 0.10)$, rispetto al sistema di riferimento in Figura 6.1. I risultati sono calcolati con il solutore MSC NASTRAN che li ha salvati in un file .f06. Lo spostamento ottenuto è

$$u_z = -1.337201 \cdot 10^{-5} \text{ m}$$

Nell'immagine seguente viene mostrata la trave deformata.

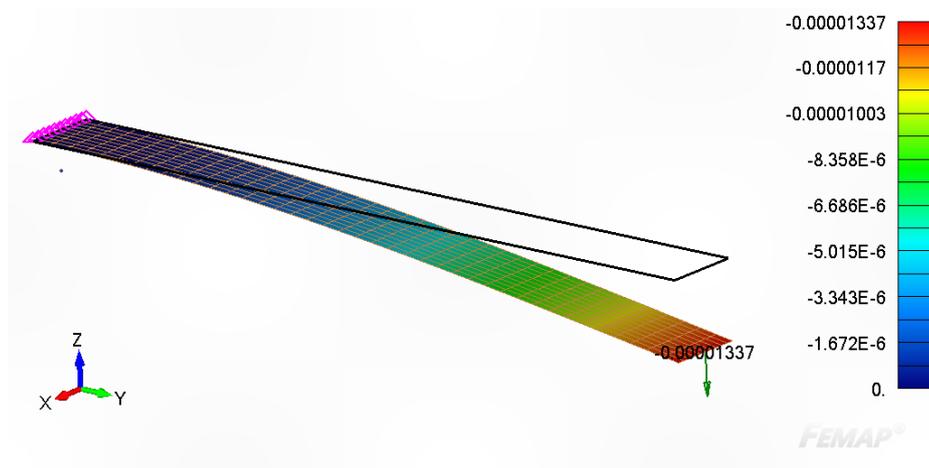


Figura 6.21: Deformata della trave su FEMAP

Lo stress lungo la trave è rappresentato nella Figura 6.22.

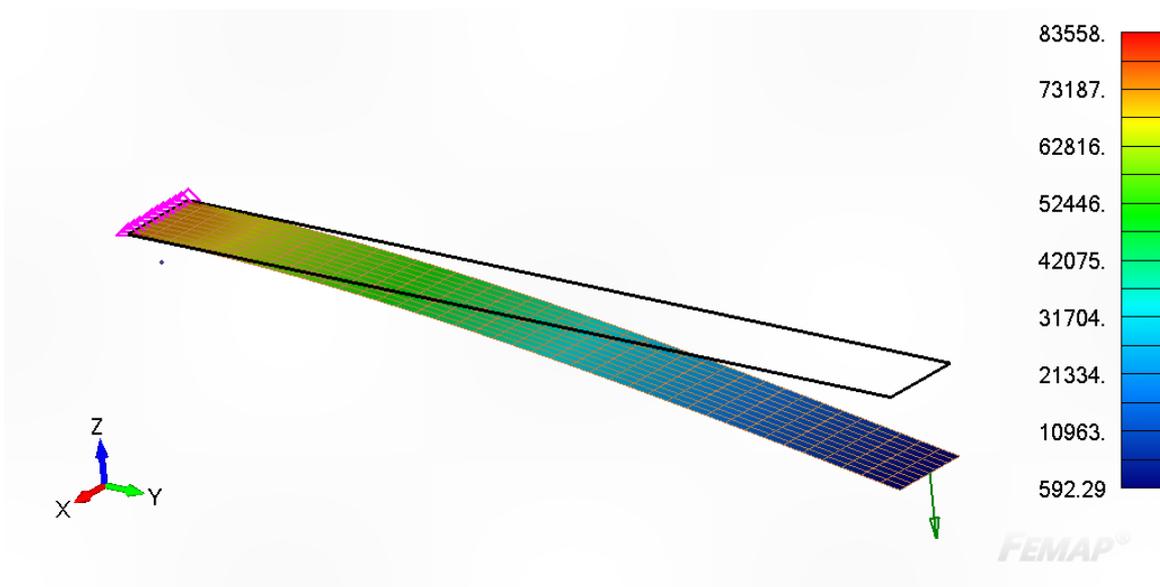


Figura 6.22: Andamento dello stress σ_y sulla superficie superiore della trave.

In particolare, si vuole determinare il valore dello stress nel nodo all'incastro in comune con i due elementi evidenziati in rosso nell'immagine seguente.

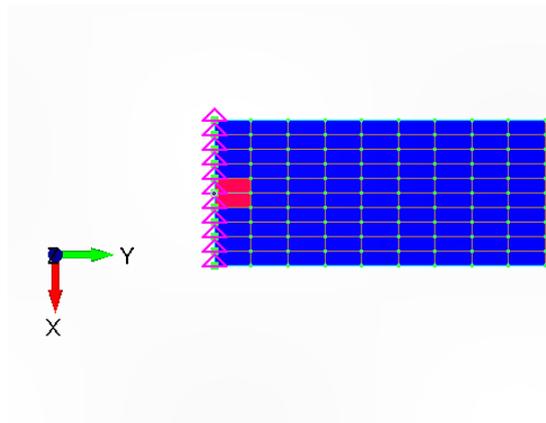


Figura 6.23: Ingrandimento della trave in cui vengono mostrati, in rosso, gli elementi 5 e 6 in cui viene valutato lo stress σ_y

NASTRAN stampa sul file `.f06` il valore dello stress nel centro e nei 4 nodi dell'elemento CQUAD. Il risultato viene calcolato come media dello stress nel nodo comune ai due elementi all'incastro:

$$\sigma_y = \frac{(\sigma_y)_5 + (\sigma_y)_6}{2} = 72970 \text{ Pa}$$

I risultati delle analisi statiche condotte con i due codici sono riassunte nella Tabella 6.9.

Modelli	u_z [$10^{-5} m$]	σ_y [Pa]
EBBT	-1.333	74406
TBT	-1.342	74406
TE1	-1.342	74406
TE2	-1.325	85460
TE3	-1.327	86989
TE4	-1.328	87627
LE4	-1.115	38958
LE9	-1.326	79271
Femap	-1.3372	72969

Tabella 6.9: Confronto dei risultati dell'analisi statica tra MUL2, utilizzando 10 elementi B4, e FEMAP.

Si può osservare come il codice commerciale si avvicini molto ai valori calcolati da MUL2 sia per gli spostamenti che per lo stress.

6.5.2 Analisi dinamica

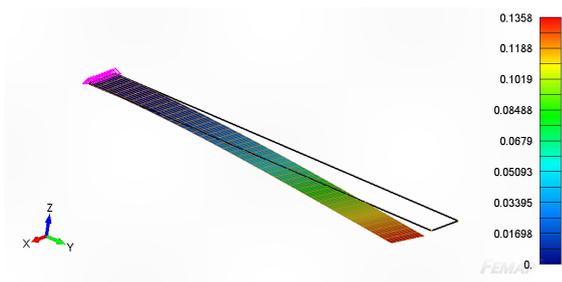
Con l'analisi dinamica si ottengono le frequenze e i modi di vibrare riportati di seguito.

Modo	Femap	TE4	4LE9	ε_{TE4} [%]	ε_{4LE9} [%]
1	42.31	42.57	42.62	0.61	0.73
2	42.45	42.57	42.62	0.28	0.40
3	253.66	255.08	256.49	0.56	1.12
4	256.57	255.08	256.49	0.58	0.03
5	516.69	372.09	403.94	27.99	21.82
6	659.84	661.67	661.88	0.28	0.31
7	667.73	671.05	678.23	0.50	1.57
8	681.43	671.05	678.23	1.52	0.47
9	1212.55	1116.5	1211.9	7.92	0.05
10	1247.25	1217.8	1237.4	2.36	0.79

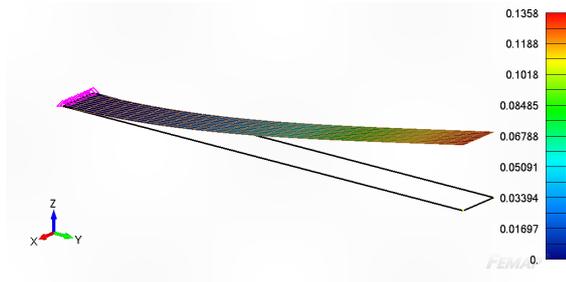
Tabella 6.10: Confronto tra le frequenze, in Hertz, ottenute con MUL2 e FEMAP.

In Tabella 6.10 sono riportate le frequenze calcolate con il codice commerciale, affiancate a quelle di due teorie accurate per il calcolo degli spostamenti, cioè TE4 e LE9. Prese le prime come riferimento, si sono affiancati anche gli errori relativi di ogni frequenza.

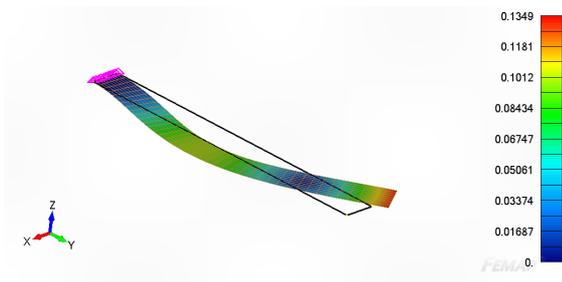
Di seguito sono riportati anche i primi 10 modi di vibrare della struttura valutati con il codice commerciale.



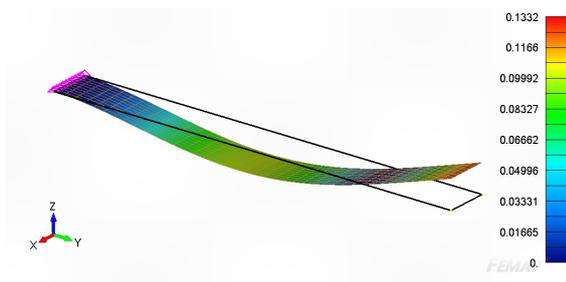
(a) Modo 1



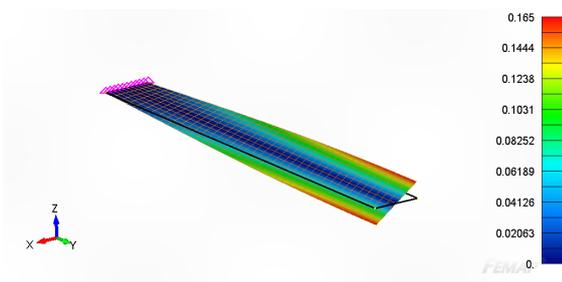
(b) Modo 2



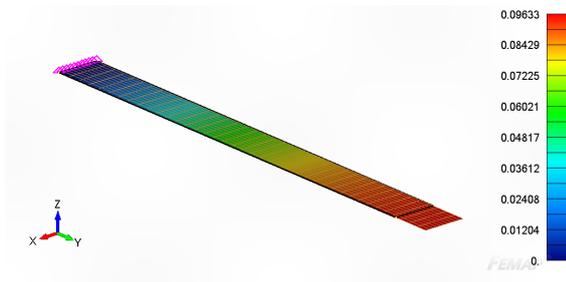
(c) Modo 3



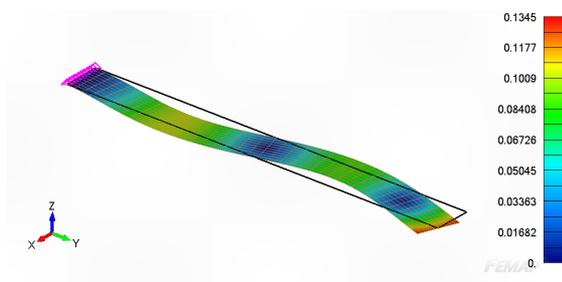
(d) Modo 4



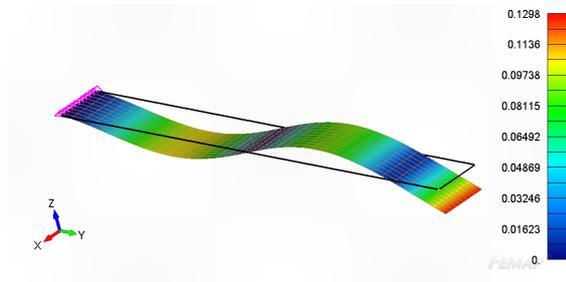
(e) Modo 5



(f) Modo 6



(g) Modo 7



(h) Modo 8

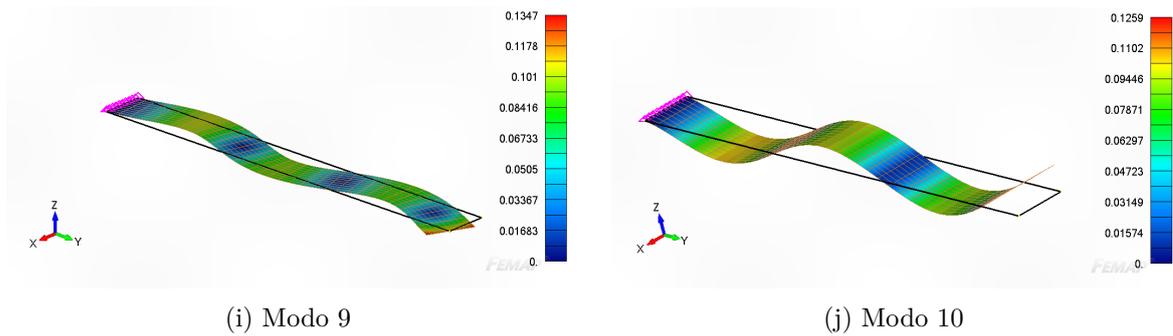


Figura 6.28: Modi vibrazionali valutati con FEMAP

I modi si avvicinano molto a quelli calcolati con il codice MUL2 con una teoria TE4. Questa è un'ulteriore validazione dei risultati calcolati in precedenza.

6.5.3 Analisi aeroelastica dinamica: flutter

La piastra viene schematizzata in modo analogo a quanto fatto con la trave: si crea una superficie nel piano xy e le si assegna uno spessore, pari a 0.001 m , in direzione z . Si crea il materiale, la proprietà e una mesh strutturale di 40×10 elementi CQUAD con un vincolo di incastro al lato in $y = 0$.

Come fatto con il codice MUL2 si esegue prima un'analisi dinamica. Le frequenze e i modi vibrazionali ottenuti verranno confrontati con quanto trovato nel paragrafo 6.3.

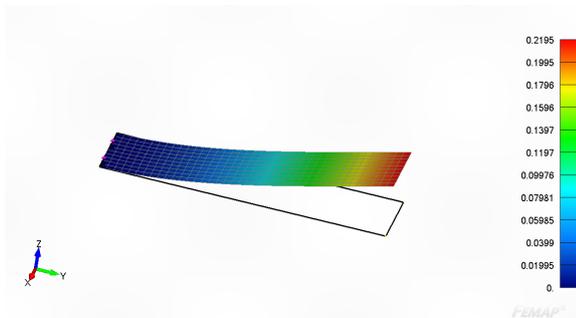
Modo	Femap	TE4	4LE9	ε_{TE4} [%]	ε_{4LE9} [%]
1	9.14	9.27	9.33	1.45	2.12
2	57.10	57.92	58.35	1.44	2.19
3	71.90	73.01	72.97	1.55	1.49
4	160.31	162.61	163.59	1.43	2.04
5	222.05	226.02	225.80	1.79	1.69
6	315.42	319.93	321.19	1.43	1.83
7	390.27	398.74	398.16	2.17	2.02
8	523.03	529.43	530.48	1.22	1.42
9	586.13	601.33	600.12	2.59	2.39
10	652.55	653.61	654.02	0.16	0.22

Tabella 6.11: Confronto tra le frequenze, in Hertz, ottenute con MUL2 e FEMAP.

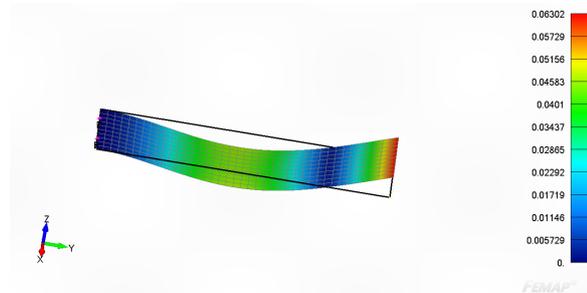
Nella Tabella 6.11 si è affiancato anche l'errore commesso tra i modelli di Taylor e Lagrange e quelli del codice commerciale. Si nota come questi sono tutti contenuti, per cui si considera affidabile la soluzione trovata.

Capitolo 6. Risultati preliminari

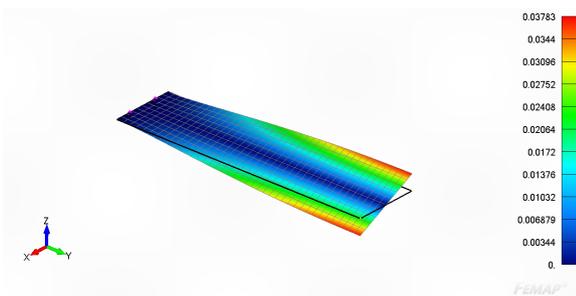
Anche i modi flessionali trovati con le analisi svolte precedentemente. Nella Figura di seguito vengono mostrati solo i primi 4 modi vibrazionali.



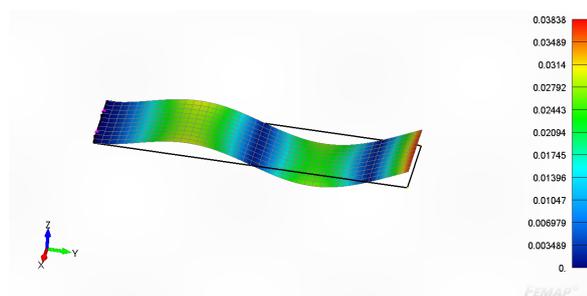
(a) Modo 1



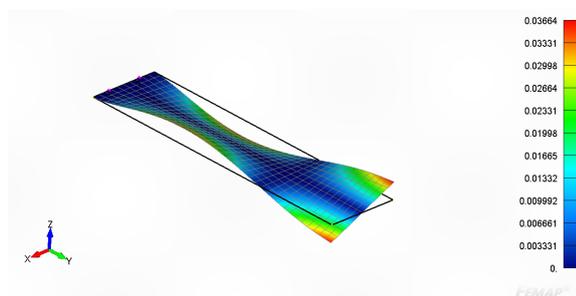
(b) Modo 2



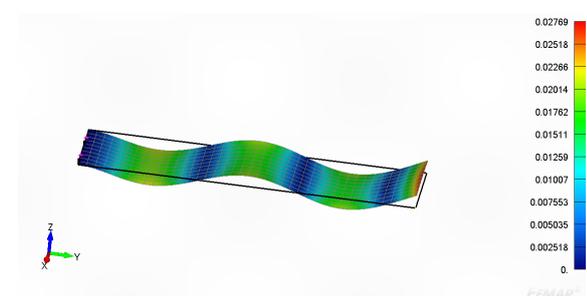
(c) Modo 3



(d) Modo 4



(c) Modo 5



(d) Modo 6

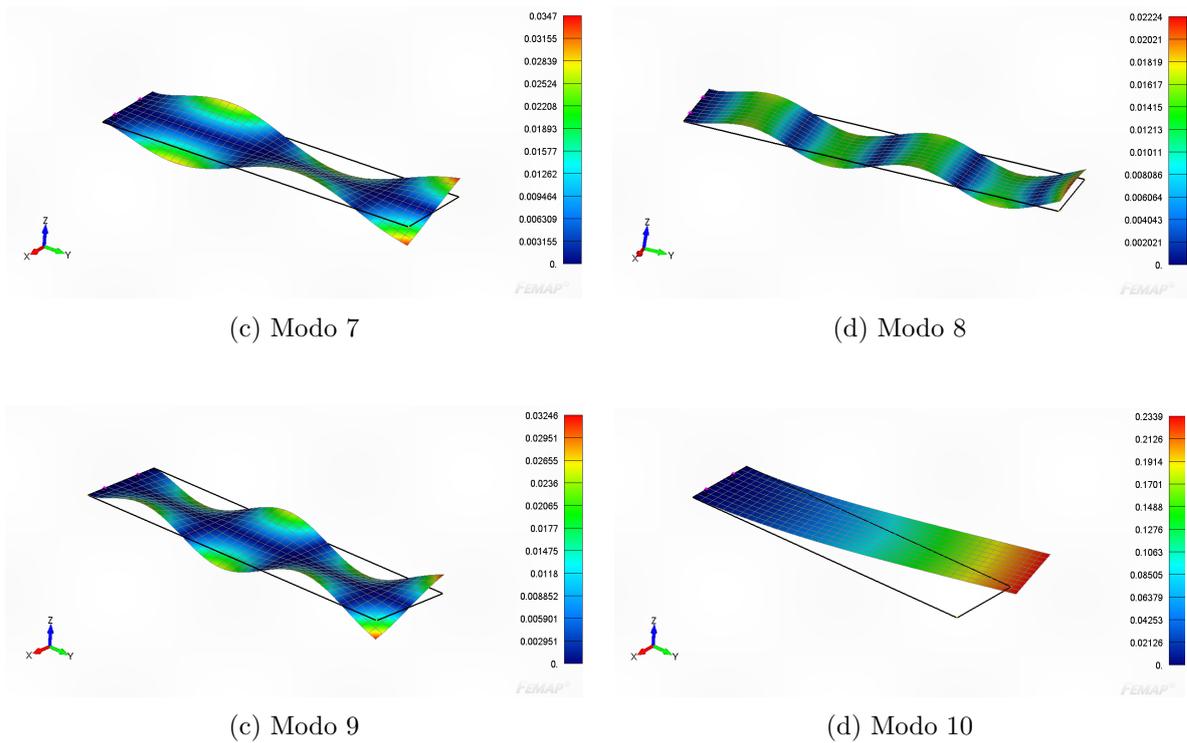


Figura 6.33: Modi vibrazionali ottenuti con FEMAP

Sul modello così creato si aggiungono la mesh aerodinamica e la spline. La prima, viene realizzata con 30×8 pannelli una volta fornito al codice le coordinate dei punti P1 e P2, rispetto al sistema di riferimento aerodinamico, e la lunghezza delle corde a queste coordinate. Per la spline invece è necessario fornire i nodi strutturali e i pannelli aerodinamici. Nella Figura 6.34 si mostrano la mesh aerodinamica e la spline ottenute.

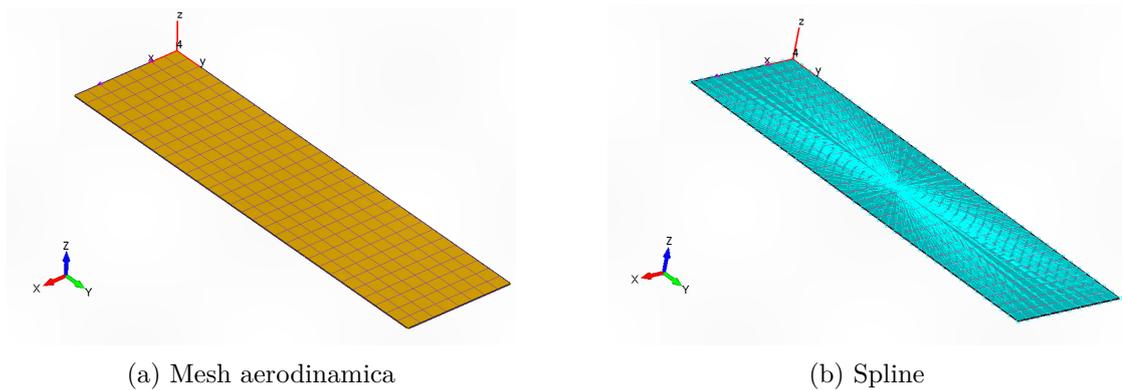


Figura 6.34: In figura si osserva anche il sistema di riferimento aerodinamico, in rosso, posizionato al bordo di attacco.

Capitolo 6. Risultati preliminari

Per eseguire l'analisi di flutter è necessario introdurre alcuni parametri aeroelastici. Questi vengono riportati di seguito

- lunghezza di riferimento: 0.076 m corda della struttura
- densità di riferimento: 1.225 kg/m^3
- simmetria: nel piano xz
- range di variazione del numero di Mach con la frequenza ridotta: $M(\kappa) = 0$ (caso incompressibile)
- range di variazione della densità (mantenuta costante al valore di riferimento)
- range di variazione della velocità: tra 2 m/s e 120 m/s
- metodo di risoluzione del flutter: PKNL (più efficiente del PK).

Si ottengono dunque i risultati in Figura 6.35.

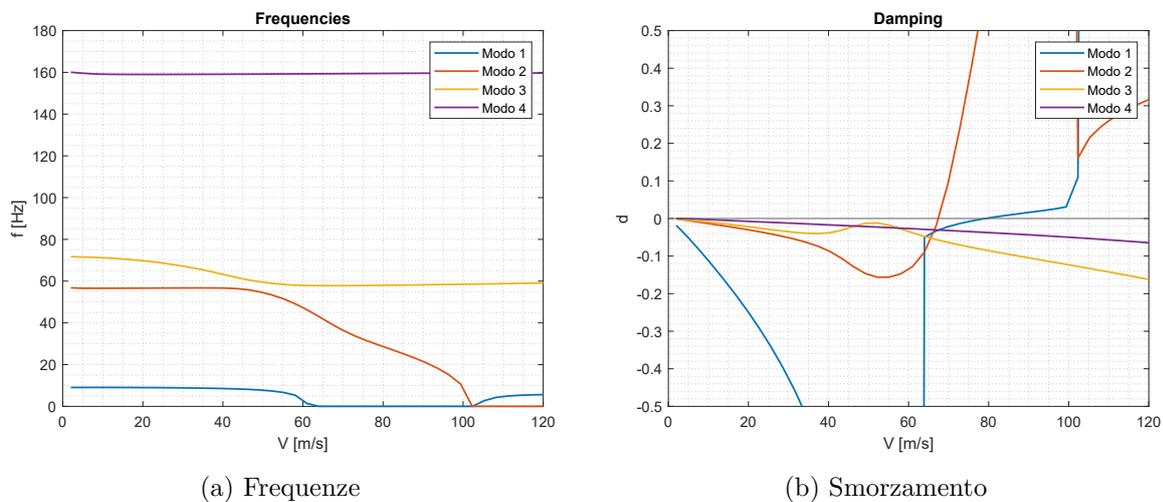


Figura 6.35: Le curve ottenute rispecchiano quanto trovato con il codice MUL2.

La condizione di flutter avviene per $V_F = 67.393 \text{ m/s}$ e $f_F = 39.184 \text{ Hz}$. Anche in questo caso è possibile identificare la condizione di divergenza per $V_D = 79.224 \text{ m/s}$.

Di seguito viene riportata una tabella dove si confrontano i risultati trovati per l'analisi di flutter con i due codici utilizzati.

	Femap	TE4	4LE9
V_F [m/s]	67.393	67.819	67.782
f_F [Hz]	39.184	38.495	38.584
V_D [m/s]	79.224	78.050	77.148

Tabella 6.12: Tabella riassuntiva dei risultati

I due codici producono risultati simili ed in linea con quelli trovati in bibliografia.

6.5.4 Analisi aeroelastica statica

Utilizzando sempre lo stesso modello si vuole eseguire ora l'analisi aeroelastica statica per confermare i risultati del paragrafo 6.4.

Come per l'analisi di flutter è necessario introdurre alcuni parametri aeroelastici, tra cui:

- lunghezza di riferimento: 0.076 m corda della struttura
- apertura alare: 0.610 m
- superficie alare di una semicorda: $c \cdot L = 0.076 \cdot 0.305 = 0.02318 \text{ m}^2$.
- simmetria: nel piano xz
- angolo di attacco, costante: $\alpha = 1^\circ$
- numero di Mach: $M = 0$
- pressione dinamica: $\frac{1}{2}\rho V^2$, cambia per ogni simulazione condotta

Si ottengono i risultati riportati nella tabella di seguito.

V [m/s]	q [Pa]	u_{za} [mm]	u_{zb} [mm]	ϑ [°]
10	61.25	0.744	0.713	0.023
20	245	3.16	3.03	0.098
30	551.25	7.91	7.59	0.242
40	980	16.58	15.92	0.497
50	1531.25	33.18	31.89	0.971
60	2205	70.43	67.78	1.997
70	3001.25	197.8	190.7	5.385
75	3445.3125	504.1	486.2	13.22
78	3726.45	2067	1994	43.51
80	3920	-2529	-2442	-49.00
85	4425.3125	-447.9	-432.8	-11.22
90	4961.25	-274.9	-265.9	-6.744

Tabella 6.13: Risultati ottenuti con il codice FEMAP.

Anche in questo caso si ottiene una velocità di divergenza tra i 78 m/s e 80 m/s . I risultati in Tabella 6.13 si avvicinano molto a quelli delle Tabelle 6.7 e 6.8. Di seguito si riporta un grafico dove vengono messi i risultati a confronto.

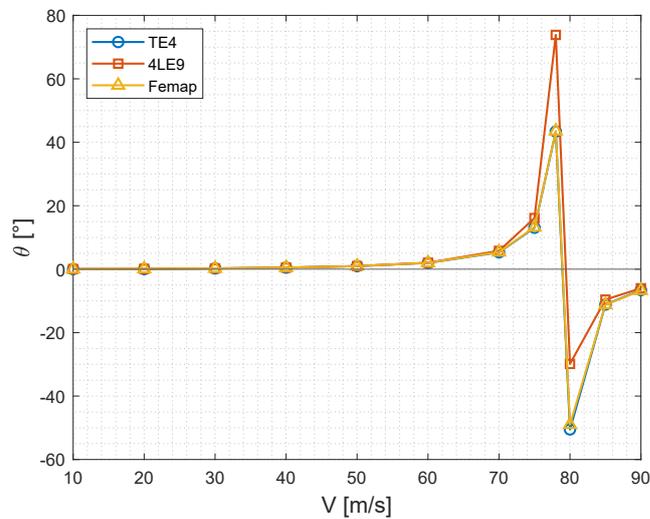


Figura 6.36: Soluzioni dei due codici a confronto

Come si vede dalla Figura, i risultati sono molto simili tra loro ad eccezione della velocità di divergenza, punto di singolarità, in cui il modello di Lagrange prevede una deformazione maggiore.

6.6 Analisi di convergenza

Si esegue ora uno studio volto a giustificare le dimensioni delle mesh strutturali e aerodinamiche scelte. Per fare ciò, si vuole mostrare l'effetto del numero di elementi CQUAD e del numero di pannelli aerodinamici sul risultato di flutter (velocità e frequenza).

6.6.1 Convergenza mesh aerodinamica

Si mantiene fissa la mesh strutturale e pari a quella utilizzata fin'ora, quindi 40 elementi in direzione y e 10 in direzione x , per un totale di 400 elementi CQUAD. Viene modificato invece il numero di pannelli partendo da una discretizzazione a pochi elementi, fino ad una con molti elementi. Per ciascuna configurazione si è eseguita un'analisi aeroelastica dinamica e si sono riassunti i risultati nella tabella seguente.

Mesh	V_F [m/s]	f_F [Hz]
15×4	65.903	40.761
20×5	66.644	39.923
25×7	67.166	39.357
30×8	67.393	39.184
35×10	67.596	38.999
40×15	67.785	38.809
45×20	67.898	38.713

Tabella 6.14: Risultati dell'analisi

I risultati riportati nella Tabella 6.14 sono riportati graficamente nella Figura 6.37.

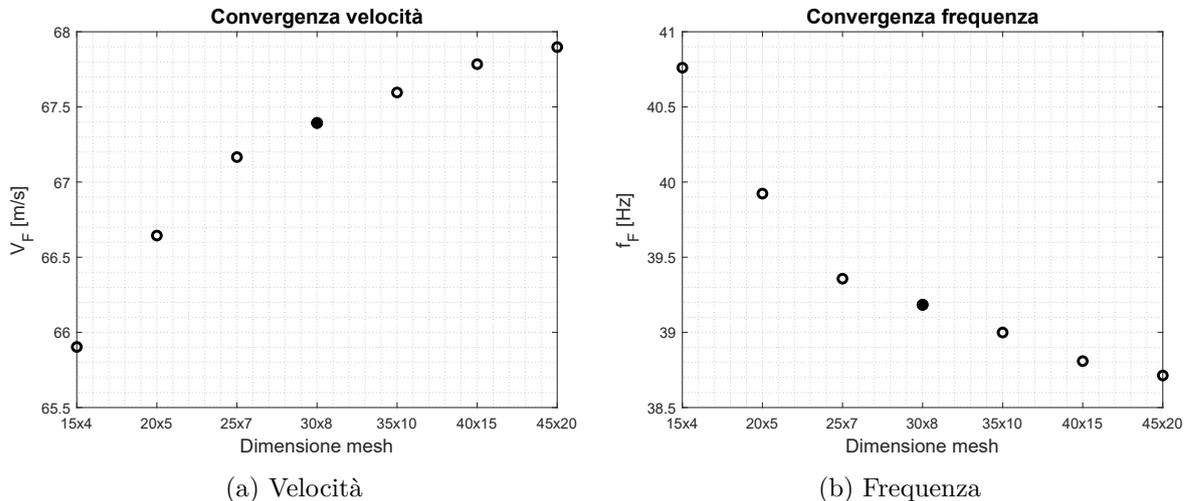


Figura 6.37: Il cerchio pieno indica la mesh utilizzata per le analisi aeroelastiche sopra riportate, ossia 30×8 .

L'analisi di convergenza è stata eseguita aumentando il numero di elementi da 60 a 900. Si è ottenuto un range di variazione di velocità di circa 2 m/s e di frequenza di circa 2 Hz .

Aumentando il numero di elementi della mesh si ottengono soluzioni che convergono ad un asintoto orizzontale, dunque più accurate. Rispetto al caso di partenza, diminuire il numero di elementi porta a soluzioni meno precise; aumentarli porta invece a soluzioni migliori, ma al costo di avere un costo computazionale sempre più grande.

Si ritiene pertanto che la scelta di una pannellatura 30×8 produca dei risultati affidabili ad un costo computazionale non eccessivo. Rispetto al caso più accurato, cioè con una pannellatura 45×20 si commette un errore di 0.75% sulla velocità e di 1.2% sulla frequenza.

6.6.2 Convergenza mesh strutturale

Si esegue ora lo studio sulla mesh strutturale in modo analogo a quanto fatto nel Paragrafo precedente 6.6.1, mantenendo una mesh aerodinamica fissa a 30×8 pannelli. I risultati sono riportati nella tabella della pagina seguente.

Mesh	$V_F \text{ [m/s]}$	$f_F \text{ [Hz]}$
20×4	66.063	38.657
30×5	66.756	38.825
35×7	67.197	39.047
40×10	67.393	39.184
45×15	67.478	39.232
50×20	67.512	39.235
60×30	67.512	39.232

Tabella 6.15: Risultati dell'analisi

I risultati riportati nella Tabella 6.15 sono riportati graficamente nella Figura 6.38.

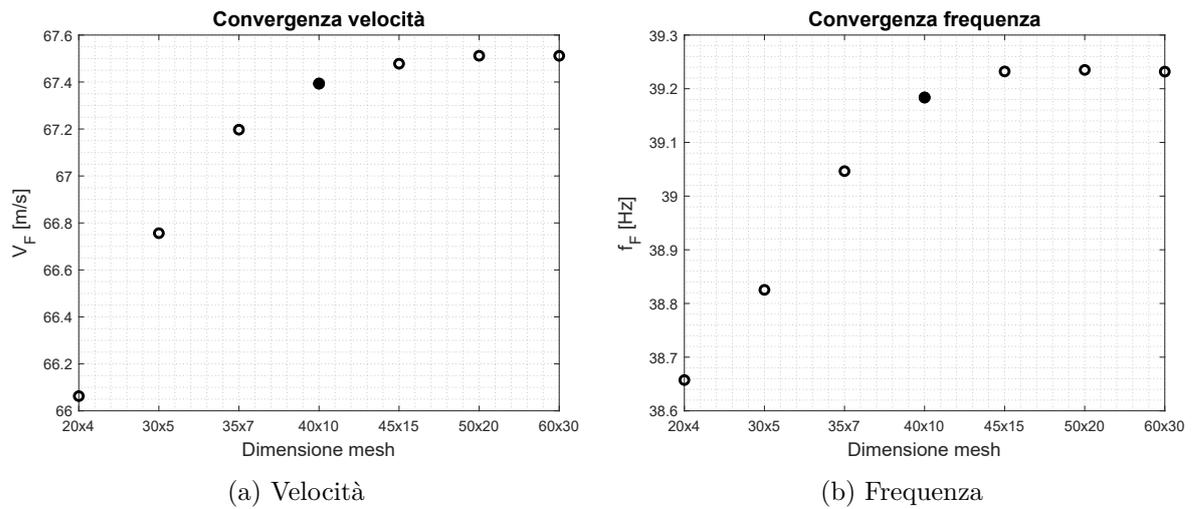


Figura 6.38: Il cerchio pieno indica la mesh utilizzata per le analisi aeroelastiche sopra riportate, ossia 40×10 .

Il numero di elementi, in questo caso va da 80 a 1800. Di nuovo, si ottiene una convergenza ad un asintoto orizzontale per mesh più fitte. Mesh con pochi elementi portano ad una qualità peggiore dei risultati.

La dimensione della mesh di partenza, 40×10 , si avvicina molto all'asintoto orizzontale che sembra essere raggiunto, osservando i grafici sia della velocità che della frequenza, a partire dalla dimensione della mesh successiva, con 675 elementi. Si considera dunque valida la mesh utilizzata poiché consente di avere una buona rappresentazione dei risultati ad un costo computazionale contenuto.

L'errore commesso rispetto al caso di mesh più fitta è di 0.18% per la velocità, e 0.12% per la frequenza.

Capitolo 7

Analisi sul velivolo finale

Il velivolo considerato è quello già studiato in [31] e in seguito in [30] e raffigurato di seguito.

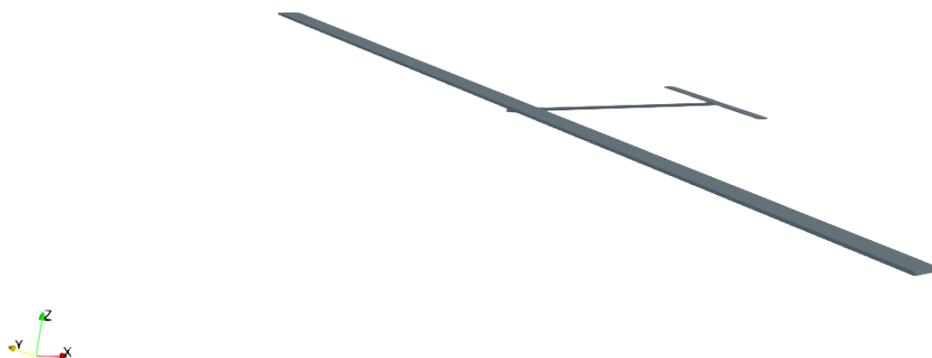


Figura 7.1: Schema del velivolo in 3D

Le dimensioni sono riportate in Tabella 7.1.

Apertura alare	a_{ala} [m]	32
Corda alare	c [m]	1
Lunghezza fusoliera	L [m]	10
Apertura alare coda	a_{coda} [m]	5
Corda alare coda	c_{coda} [m]	0.5

Tabella 7.1: Dimensioni del velivolo considerato.

mentre le caratteristiche dei materiali sono invece riassunte in Tabella 7.2.

Modulo di Young	E [GPa]	180
Modulo di Poisson	ν [-]	0.30
Densità	ρ [kg/m ³]	1800

Tabella 7.2: Caratteristiche del materiale utilizzato.

La superficie portante del velivolo è la somma della superficie della coda e di quella dell'ala; quest'ultima contribuisce al 93% circa alla superficie portante totale. Per questo motivo è necessario studiarla prima separatamente dal resto dell'aereo. In seguito, verranno aggiunte all'ala anche la fusoliera, la coda e in ultimo i motori. Tuttavia, ancor prima di eseguire i test sull'ala del velivolo si vuole eseguire un test sull'ala di Goland per comprendere il comportamento aeroelastico che ci si deve aspettare spostando i motori nelle due direzioni considerate.

7.1 Ala di Goland

Per ala di Goland si intende un test aeroelastico spesso utilizzato come validazione introdotto nel 1945 da M. Goland [32]. Il test consiste nel determinare le caratteristiche aeroelastiche di una trave incastrata di sezione rettangolare con delle caratteristiche geometriche e meccaniche particolari ed elencate nella tabella seguente:

Parametro	Valore	Valore convertito
Apertura alare	20 ft	6.096 m
Corda	6 ft	1.8288 m
Rigidezza a flessione	23.6E+06 <i>lbf · ft²</i>	9.75E+06 <i>N · m²</i>
Rigidezza a torsione	2.39E+06 <i>lbf · ft²</i>	0.99E+05 <i>N · m²</i>
Massa per unità di lunghezza	0.746 <i>slugs/ft</i>	35.719 <i>kg/m</i>
Momento di inerzia per unità di lunghezza	1.943 <i>slugs · ft</i>	8.643 <i>kg · m</i>
Densità dell'aria	0.00237 <i>slugs/ft³</i>	1.221 <i>kg/m³</i>

Tabella 7.3: Dati relativi all'ala di Goland

Si vuole validare lo studio condotto in [33] dove l'ala di Goland è costituita da un materiale viscoelastico e in cui viene considerata anche l'influenza della spinta dei motori.

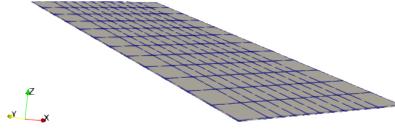


Figura 7.2: Modello FEM dell'ala di Goland.

Per trovare conferma nello studio bibliografico citato si realizza la trave con 12 elementi B3 e una sezione rettangolare formata da 11 elementi $L9$. Con questa discretizzazione e con i dati della Tabella 7.3 si sono condotte le analisi aeroelastiche spostando i motori. Questi ultimi, sono modellati come delle masse concentrate in corrispondenza di uno degli elementi $L9$ della sezione. I motori vengono in questo modo spostati in 9 posizioni lungo la corda in modo che il loro baricentro sia alla stessa quota dell'asse elastico dell'ala, cioè nella mezzeria della sezione, come eseguito dallo studio bibliografico.

In assenza di dati sufficienti per replicare esattamente lo studio, non è stato possibile ottenere le stesse rigidezze utilizzate in [33]. Osservando altri studi trovati in letteratura, [34] e [35], ci si rende conto che l'effetto della rigidezza, della geometria (rettangolare) della sezione, e dunque della rigidezza della struttura non influisce sul comportamento della velocità di flutter per le varie posizioni dei motori lungo la corda e lungo l'apertura alare. L'effetto principale che si riscontra è una variazione della velocità di flutter più o meno debole.

Con queste considerazioni si vuole eseguire il test sull'ala su un'ala con le stesse dimensioni geometriche riportate in Tabella 7.3, ma considerando come materiale alluminio, tale per cui:

$$E = 73.8 \text{ GPa}$$

$$G = 28.4 \text{ GPa}$$

I risultati ottenuti sono mostrati di seguito e vengono confrontati i risultati relativi ad un'analisi aerodinamica stazionaria della bibliografia. Si precisa che la denominazione del sistema di riferimento presentata in questo elaborato differisce da quella dello studio considerato.

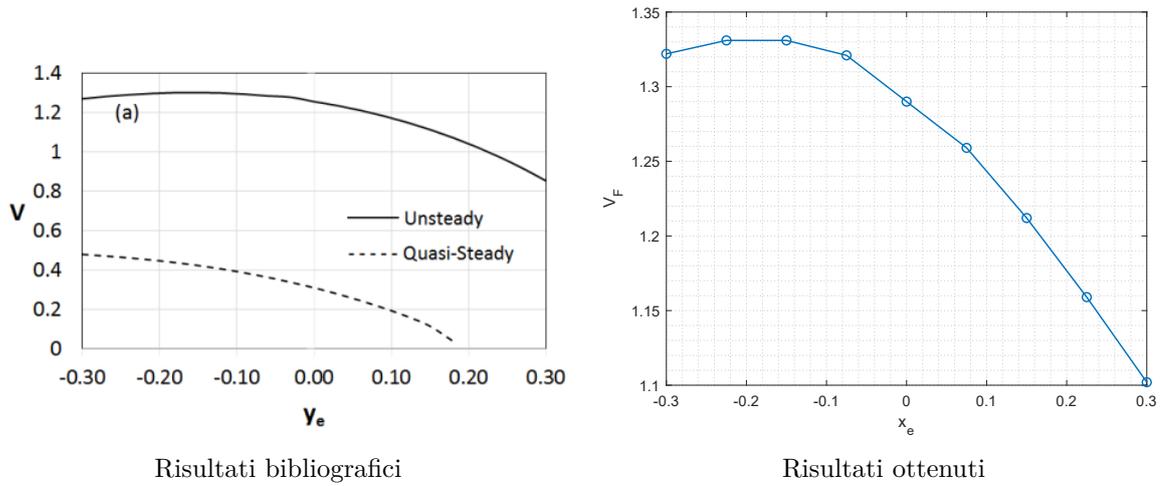


Figura 7.3: Andamento della velocità di flutter per varie posizioni dei motori lungo la corda.

La velocità di flutter viene normalizzata con la prima frequenza torsionale e la lunghezza della corda:

$$V_F = \frac{V_{FEM}}{b\omega_\theta}$$

x_e esprime invece lo scostamento del baricentro del motore dall'asse elastico dell'ala rapportato alla lunghezza della corda. La posizione $x_e = 0$ rappresenta la condizione in cui baricentro del motore e asse elastico dell'ala coincidono; valori negativi si riferiscono a motori avanzati verso il bordo di attacco, mentre valori positivi si riferiscono a motori arretrati verso il bordo di fuga.

Il massimo della velocità di flutter si trova quando il motore è distante circa il 10% della corda verso il bordo di attacco. Per valori superiori la velocità decresce. Questo è in accordo con lo studio citato.

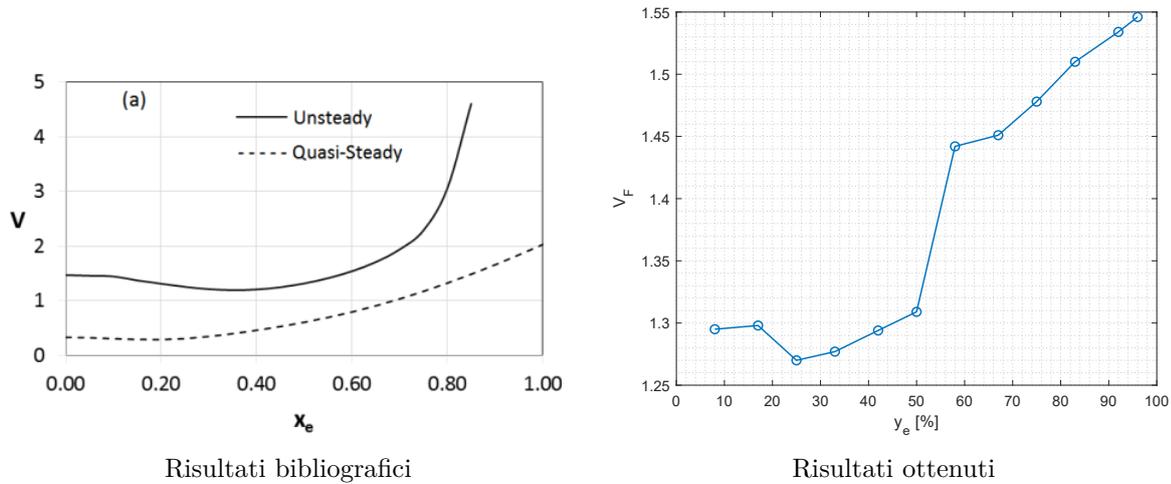


Figura 7.4: Andamento della velocità di flutter per varie posizioni dei motori lungo l'apertura alare.

y_e esprime la distanza che separa il baricentro del motore dalla radice della trave, cioè l'incastro, rapportato alla lunghezza complessiva della trave.

La velocità minima di flutter si ha quando i motori sono posizionati circa al 40% dell'apertura alare; oltre tale punto il punto di instabilità aumenta vertiginosamente. Anche questo risultato è in accordo con gli i risultati bibliografici considerati.

7.2 Analisi dell'ala

Per eseguire delle analisi più fedeli ad un caso reale si considera un'ala in parete sottile e rinforzata con delle centine poste una all'incastro con la fusoliera, e una seconda posta circa al 42% della semiapertura alare.

Per discretizzare l'ala si sono considerate due semiale simmetriche rispetto all'asse x e vincolate con un incastro in corrispondenza del piano $y = 0$. Ogni semiala è costruita con 7 elementi B3 e, per la sezione, 4 elementi L9 (uno per ogni lato della sezione) e un quinto elemento centrale per modellare la centina dove presente.

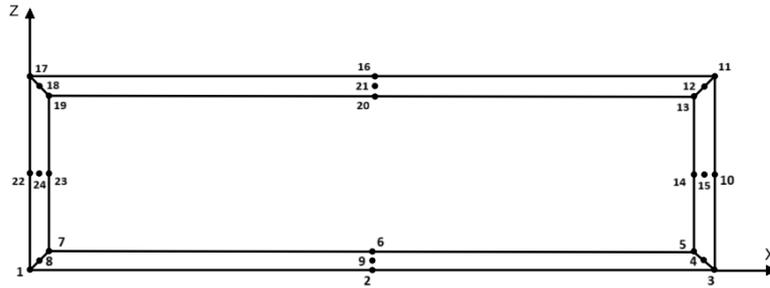


Figura 7.5: (a) Ala in parete sottile

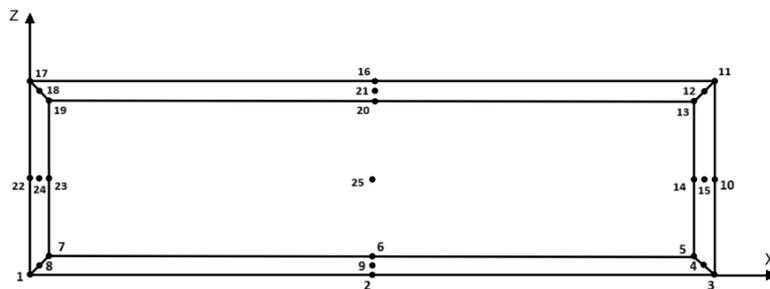


Figura 7.6: (b) Ala in parete sottile rinforzata

La lunghezza degli elementi B3 è costante lungo tutta l'apertura ad eccezione dell'elemento all'incastro con la fusoliera, il quale è reso di una lunghezza tale da far combaciare i nodi di questa e quelli dell'ala. In questo modo lungo ciascuna semiala si ha un primo elemento molto corto alla radice e a seguire altri 6 elementi, più lunghi, tutti di uguale lunghezza e di cui uno è rinforzato con una centina.

Sulla struttura così discretizzata si è effettuata prima un'analisi dinamica per valutare i modi e le frequenze vibrazionali. In seguito si è condotta anche un'analisi aeroelastica dinamica per ottenere il punto di flutter. I risultati delle due analisi sono riportati a seguire.

Modo	f [Hz]	Tipo di modo
1	0.611	I flessionale attorno a x
2	3.858	II flessionale attorno a x
3	3.943	I flessionale attorno a z
4	12.098	III flessionale attorno a x
5	12.969	I torsionale
6	24.583	II flessionale attorno a z

Tabella 7.4: Frequenze e modi di vibrazione dei primi 6 modi dell'ala incastrata.



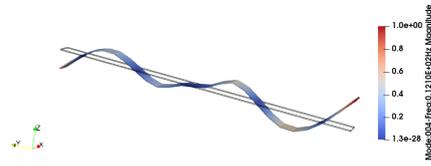
(a) Modo 1



(b) Modo 2



(c) Modo 3



(d) Modo 4



(e) Modo 5



(f) Modo 6

Per eseguire le analisi di flutter è necessario specificare i punti per le spline nel file `POSTPROCESSING.dat`; in questo caso se ne sono scelti 100 equispaziati su tutta l'ala. Oltre a ciò si definiscono alcuni parametri nel file `NASTRAN_FLUTTER.dat`, in particolare si è tolta la simmetria nel piano xz , i punti P1 e P2, necessari per la costruzione della mesh aerodinamica (60×8 pannelli), hanno coordinate $(0, 0, 0)$ e $(0, -16, 0)$. Infine si considera un range di velocità tra 2 m/s e 150 m/s con 300 intervalli all'interno.

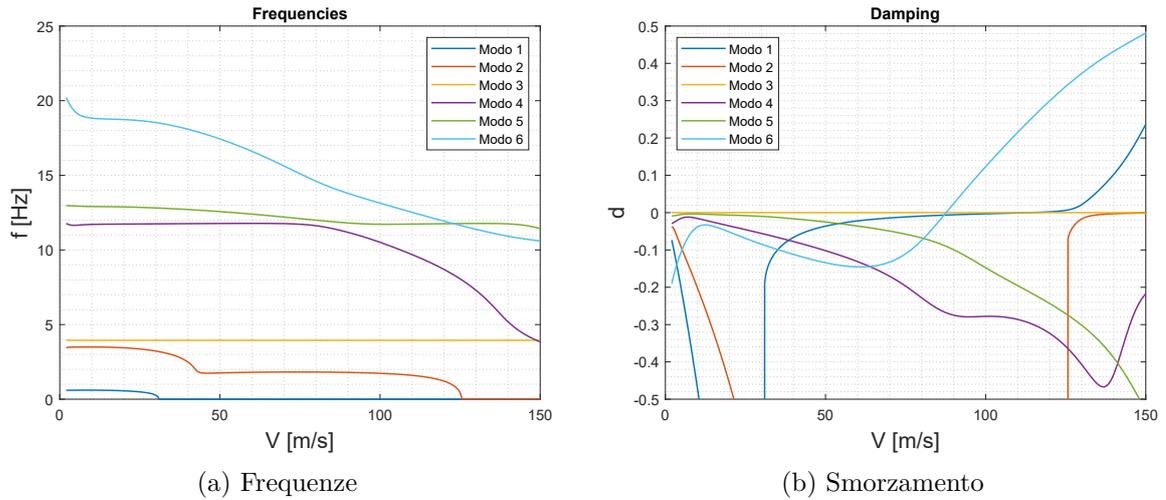


Figura 7.10: Frequenze e smorzamenti dei primi 6 modi dell'ala incastrata

Dalla Figura 7.10 si trova che il primo modo ad andare in instabilità è il sesto, cioè il secondo modo flessionale attorno a z . Risulta pertanto:

$$V_F = 89.75 \text{ m/s}$$

$$f_F = 13.294 \text{ Hz}$$

Per avere un comportamento dinamico più fedele alla realtà si può eseguire un'analisi dinamica senza il vincolo di incastro; il modello strutturale è lo stesso ma è necessario togliere il vincolo dal file BC.dat.

Modo	f [Hz]	Tipo di modo
7	1.538	I flessionale attorno a x
8	3.202	II flessionale attorno a x
9	6.535	III flessionale attorno a x
10	9.950	I flessionale attorno a z
11	10.063	IV flessionale attorno a x
12	12.654	I torsionale
13	15.933	V flessionale attorno a x
14	20.739	II flessionale attorno a z
15	41.630	III flessionale attorno a z

Tabella 7.5: Frequenze e modi di vibrazione dell'ala libera

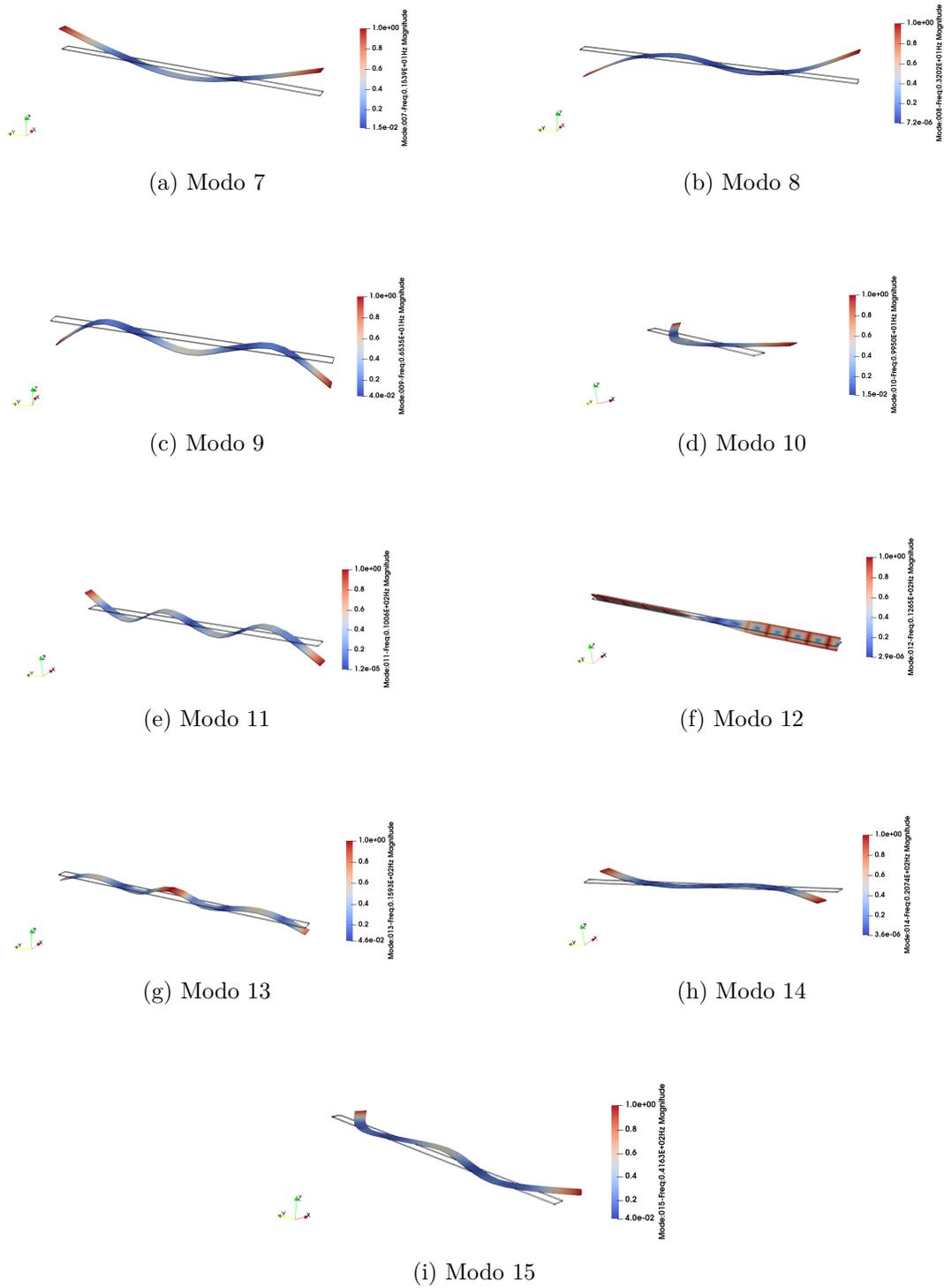


Figura 7.15: Modi vibrazionali dell'ala non incastrata.

Nella Tabella 7.5 e nella Figura ?? sono rappresentati solo i modi dal 7 al 15 perché i primi 6 sono di corpo rigido, quindi non competono ai modi vibrazionali di un'analisi dinamica, mentre i modi dal 16 in poi riguardano i modi locali degli elementi beam.

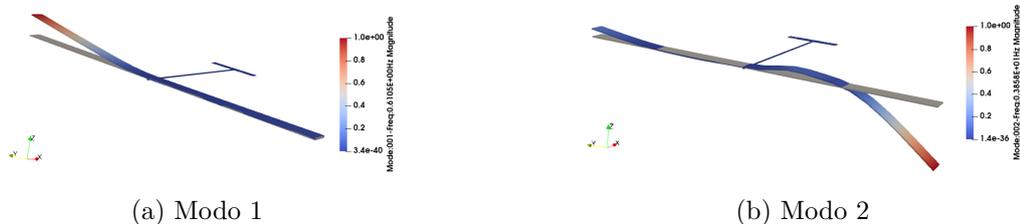
Si aggiungono ora la fusoliera e la coda. La prima è discretizzata con 11 elementi beam B3 per tutta la sua lunghezza di 10 m lungo l'asse x , e la cui sezione è rettangolare con una larghezza, lungo y , di 0.05 m e un'altezza di 0.10 m in z . Per la coda invece si realizza la stessa struttura in parete sottile dell'ala senza essere rinforzata con le centine. Ha un'apertura complessiva di 5 m lungo y e una sezione rettangolare con una corda di 0.5 m in x , un'altezza di 0.04 m in z . Questa viene discretizzata con 7 elementi beam B3 per ogni semiapertura, quindi un totale di 14 lungo l'asse y .

7.3 Analisi dinamica e di flutter del velivolo

Costruito il modello del velivolo si conduce prima un'analisi dinamica per studiare frequenze naturali e modi di vibrazione. Si pone innanzitutto un vincolo d'incastro in $y = 0$, cioè nel piano di simmetria. Le semiali dell'ala e della coda risultano in questo modo incastrate alla fusoliera. Si ottengono i seguenti risultati:

Modo	Ala		Velivolo	
	f [H_z]	Tipo di modo	f [H_z]	Tipo di modo
f_1	0.6106	I flessione x	0.6105	I flessione x
f_2	3.858	II flessione x	3.858	II flessione x
f_3	3.943	I flessione z	3.944	I flessione z
f_4	12.098	III flessione x	12.099	III flessione x
f_5	12.968	I torsione	12.970	I torsione
f_6	24.583	II flessione z	17.228	IV flessione x
f_7	22.459	I instabilità pannelli	24.585	II flessione z

Tabella 7.6: Contronto delle frequenze e dei modi di vibrare tra il velivolo incastrato con quelli dell'ala incastrata della Tabella 7.4.



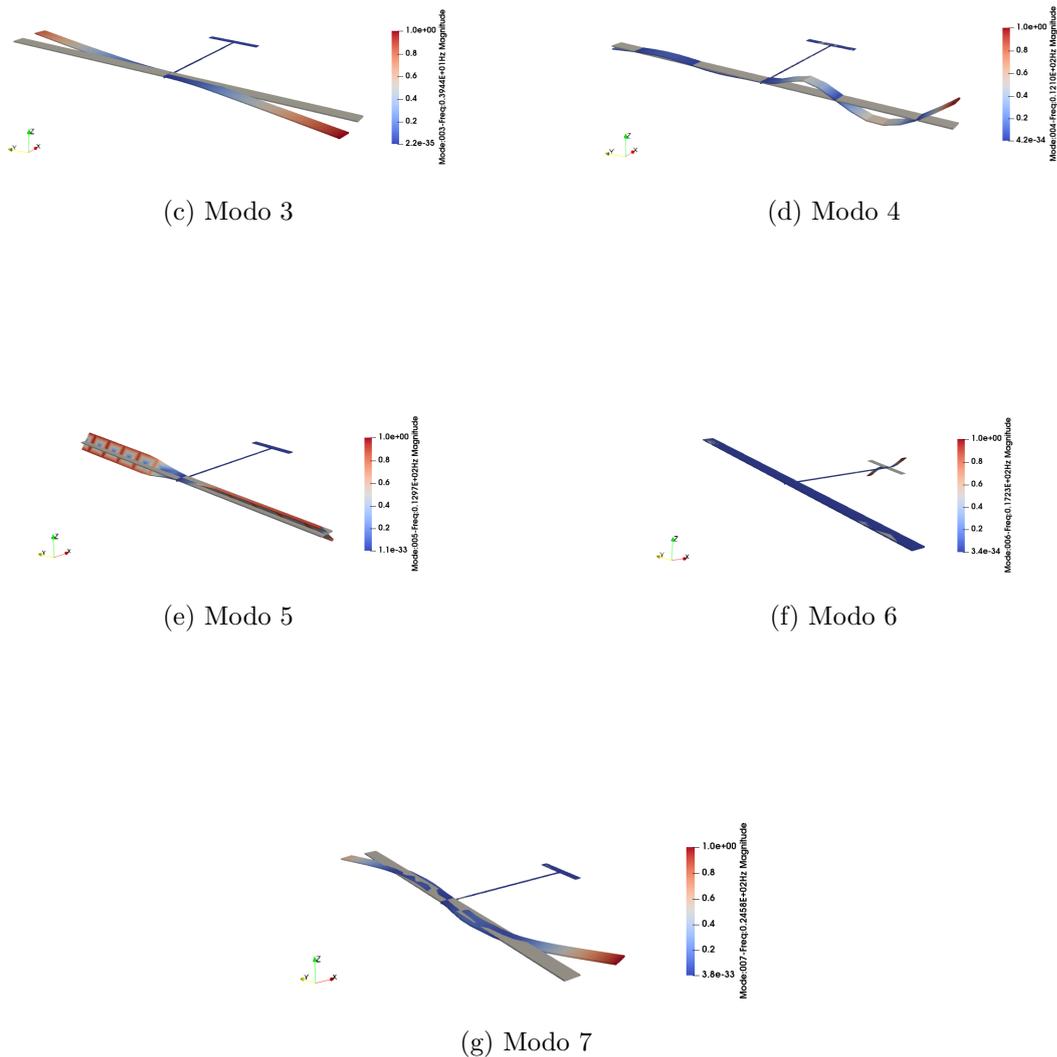


Figura 7.19: Modi vibrazionali del velivolo incastrato.

Dalla Tabella 7.6 si può verificare come le frequenze siano circa le stesse per i due casi, così come i modi di vibrare ad eccezione del settimo dove per l'ala è presente il modo di instabilità dei pannelli.

Per l'analisi di flutter bisogna considerare anche la coda come superficie portante; dunque ad essa bisogna conferire dei punti per le spline e una pannellatura. Si è deciso di assegnare per la coda 50 punti equispaziati per le spline e di creare una mesh aerodinamica di 30×4 . Per l'ala invece si mantengono 100 punti per le spline e una mesh di 60×8 pannelli. Inoltre, l'intervallo di velocità considerato per le analisi è tra 2 m/s e 150 m/s con 500 intervalli all'interno.

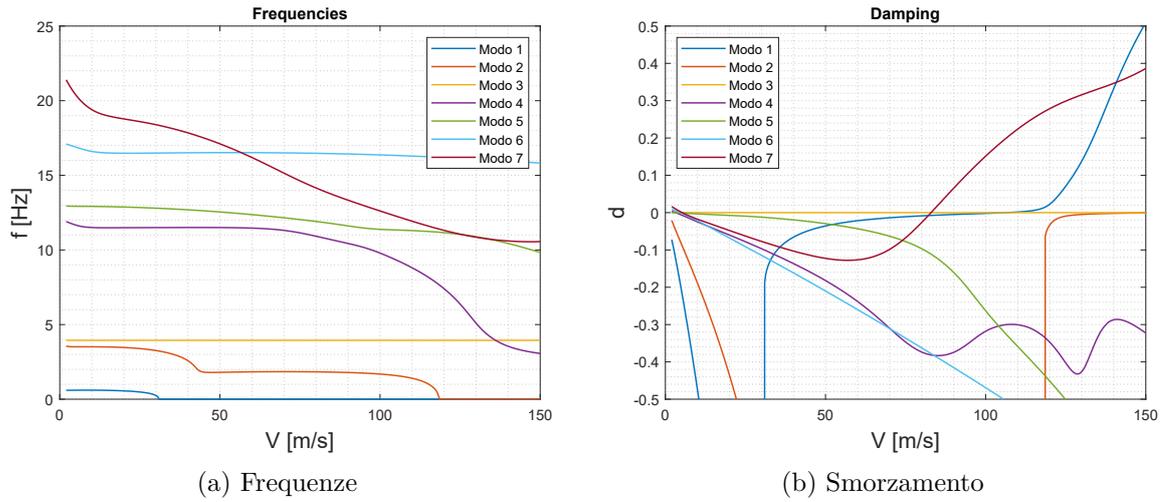


Figura 7.20: Frequenze e smorzamenti dei primi 7 modi del velivolo incastrato.

Il modo ad andare in instabilità è il settimo:

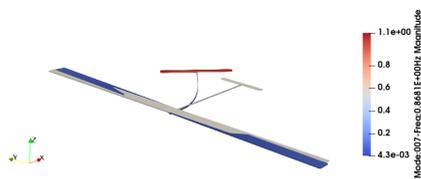
	V_F [m/s]	f_F [Hz]
Ala	89.75	13.294
Velivolo	82.95	13.89

Tabella 7.7: Confronto della condizione di flutter tra il velivolo e l'ala incastrati.

Il velivolo ora è lasciato libero, cioè si è rimosso il vincolo di incastro in $y = 0$ nel file `BC.dat`, come già eseguito per l'ala. Vengono ora riportati i risultati ottenuti per le stesse analisi.

Modo	f [Hz]
7	0.8681
8	1.4477
9	3.1928
10	5.0898
11	5.2509
12	5.4024
13	6.3194
14	8.3604
15	10.1029
16	11.9482
17	12.7069
18	12.9364
19	14.9274
20	20.1069

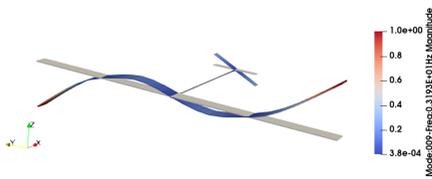
Tabella 7.8: Frequenze e modi vibrazionali dell'ala libera.



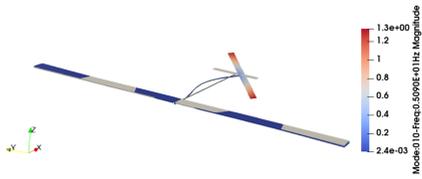
(a) Modo 7



(b) Modo 8



(c) Modo 9



(d) Modo 10

7.3. Analisi dinamica e di flutter del velivolo

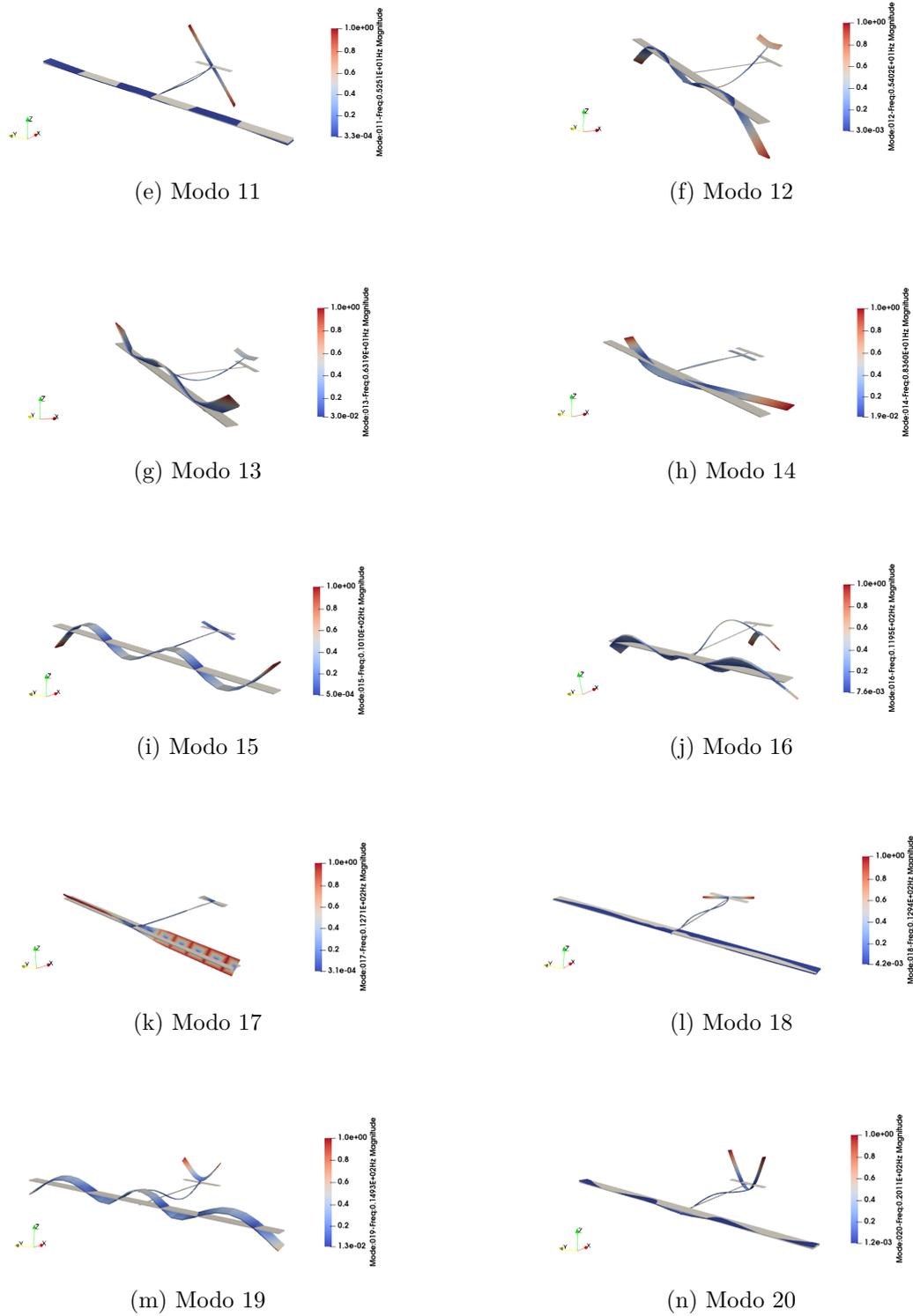


Figura 7.27: Modi vibrazionali del velivolo libero

L'analisi di flutter viene condotta inserendo gli stessi parametri aeroelastici dell'analisi precedente. Quindi si scelgono 100 punti per la spline sull'ala e altri 50 per la coda. Le mesh per ala e coda sono rispettivamente 60×8 e 30×4 pannelli. L'intervallo di velocità è compreso tra 2 m/s e 150 m/s con 500 intervalli.

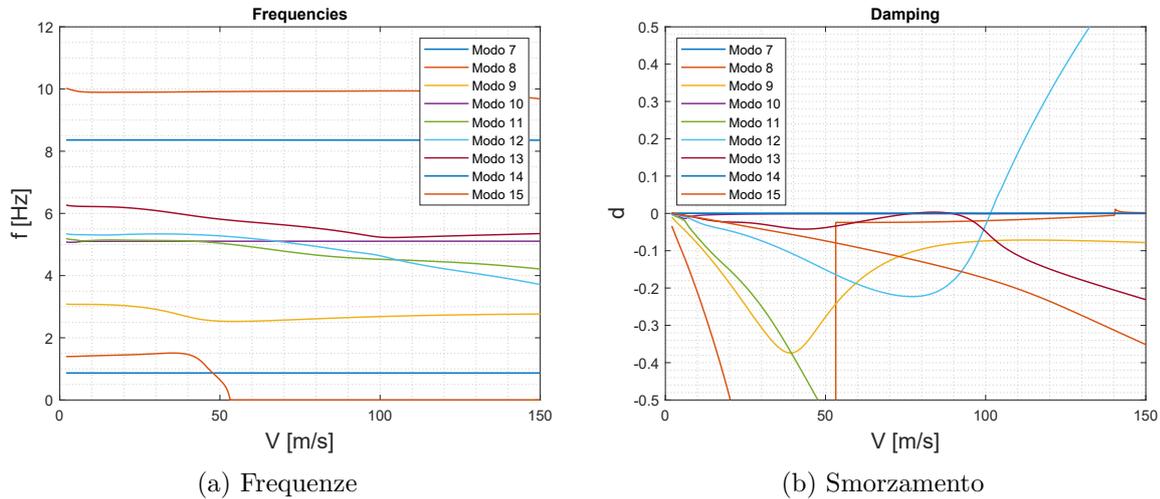


Figura 7.28: Frequenze e smorzamenti di alcuni modi del velivolo libero.

In questo caso il modo ad andare in instabilità è il dodicesimo:

	V_F [m/s]	f_F [Hz]
Ala	102.15	12.802
Velivolo	108.15	4.657

Tabella 7.9: Confronto della condizione di flutter tra il velivolo e l'ala libera.

7.4 Velivolo con i motori

Prima di aggiungere i motori si è deciso di infittire la mesh sull'ala in modo tale da poter spostare i motori lungo la corda e lungo l'apertura alare in un numero maggiore di punti.

Introducendo ora i motori sulle semiali si ottiene la versione definitiva del velivolo che sarà utilizzata per le analisi con le reti neurali. È dunque importante modellarli in maniera corretta.

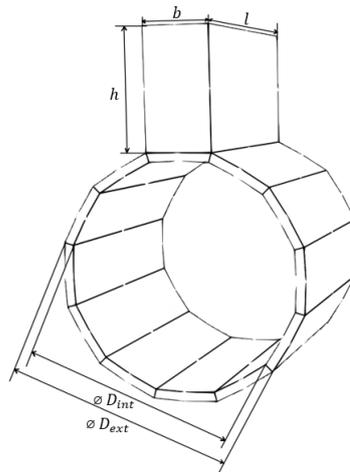


Figura 7.29: Modello del motore e del pilone associato.

Il motore viene discretizzato con un dodecagono inscritto in una circonferenza di diametro 1 m . Lo spessore totale è di 4 cm , metà verso l'interno e metà verso l'esterno della circonferenza. Sulla parte superiore è posizionato il pilone che ha il compito di collegarlo alla semiala dell'aereo. Il resto delle dimensioni geometriche è fornito nella tabella seguente.

Diametro interno	$\varnothing D_{int}$ [m]	0.96
Diametro esterno	$\varnothing D_{ext}$ [m]	1.04
Lunghezza motore/pilone	l [m]	1
Altezza pilone	h [m]	0.50
Larghezza pilone	b [m]	0.27277

Tabella 7.10: Dimensioni del motore e del pilone.

Per fornire le adeguate proprietà del materiale si ha fatto riferimento alla Figura 7.30, [36].

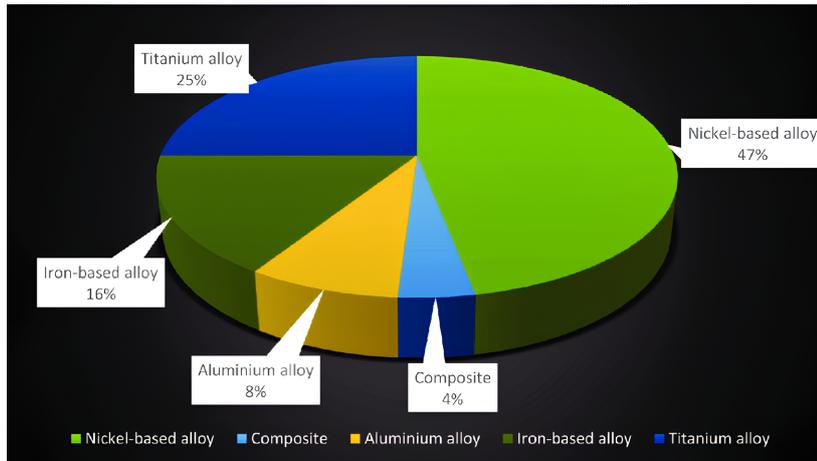


Figura 7.30: Scomposizione dei materiali utilizzati nel GE CF6.

Per ciascun materiale utilizzato nel motore GE CF6 si sono cercate le proprietà medie di modulo di Young, modulo di Poisson e densità e se ne è calcolata la media ponderata con le percentuali della Figura 7.30, [36].

Materiale	Frazione in massa	ρ [kg/m^3]	E [GPa]	ν
Leghe Ni	47%	8290	206	0.3
Leghe Ti	25%	4530	114	0.322
Leghe Fe	16%	7170	156	0.283
Leghe Al	8%	2700	70	0.33
Compositi	4%	2700	200	0.20
Materiale motore	-	6500	164	0.3012

Tabella 7.11: Proprietà medie dei materiali utilizzati nel motore considerato.

Il materiale utilizzato per i motori ha le proprietà che compaiono nella Tabella 7.11, i piloni hanno invece lo stesso materiale del resto dell'aereo.

Definita la geometria del velivolo, dei motori, e i loro materiali si ottiene la configurazione finale riportata nell'immagine seguente.



Figura 7.31: Modello definitivo del velivolo.

Verranno ora condotte le analisi dinamiche e di flutter su questo modello e verranno confrontate con la configurazione del velivolo senza motori.

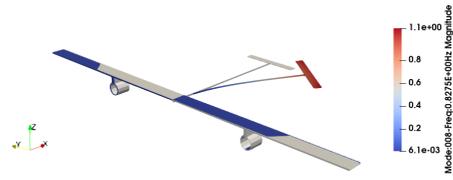
Per prima cosa si conduce un'analisi dinamica per ottenere frequenze e modi di vibrazione.

Modo	f [Hz]
7	3.0067
8	3.2448
9	3.3047
10	5.0767
11	5.0964
12	5.2469
13	5.6867
14	8.2987
15	9.0215
16	9.7697
17	10.642
18	12.927
19	14.450
20	19.962
21	20.224

Tabella 7.12: Frequenze e modi vibrazionali del velivolo libero con i motori.



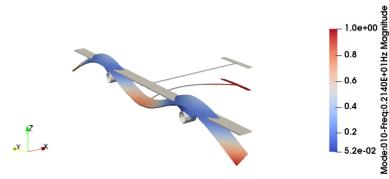
(a) Modo 7



(b) Modo 8



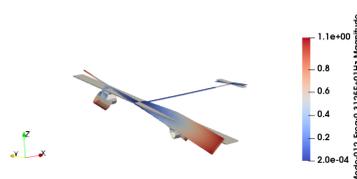
(c) Modo 9



(d) Modo 10



(e) Modo 11



(f) Modo 12



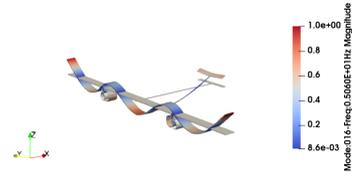
(g) Modo 13



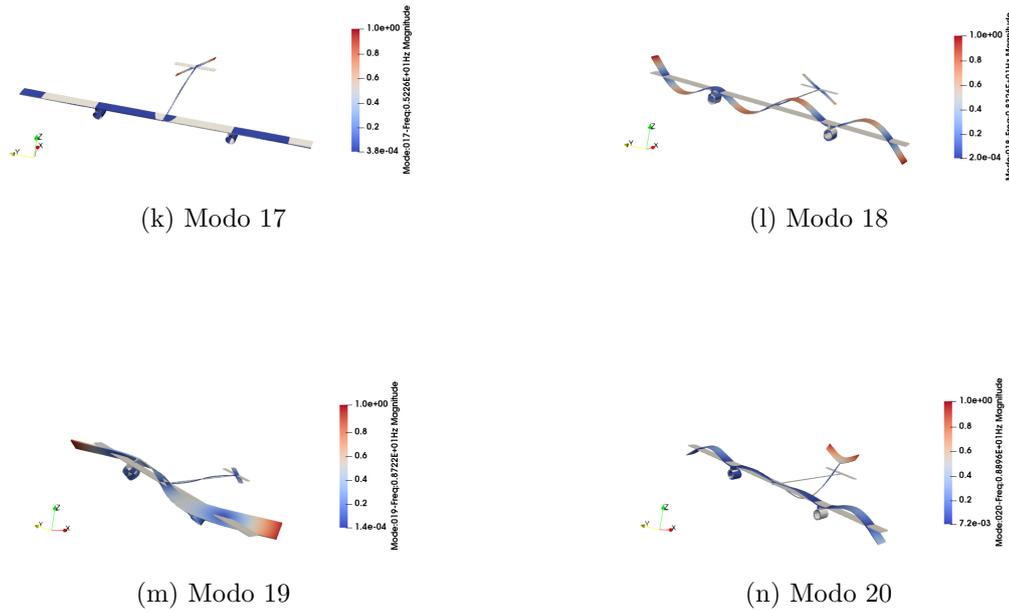
(h) Modo 14



(i) Modo 15



(j) Modo 16



Per l'analisi di flutter si inseriscono gli stessi parametri del velivolo senza motori: stessa mesh, stessi punti per la spline e stesso intervallo di velocità.

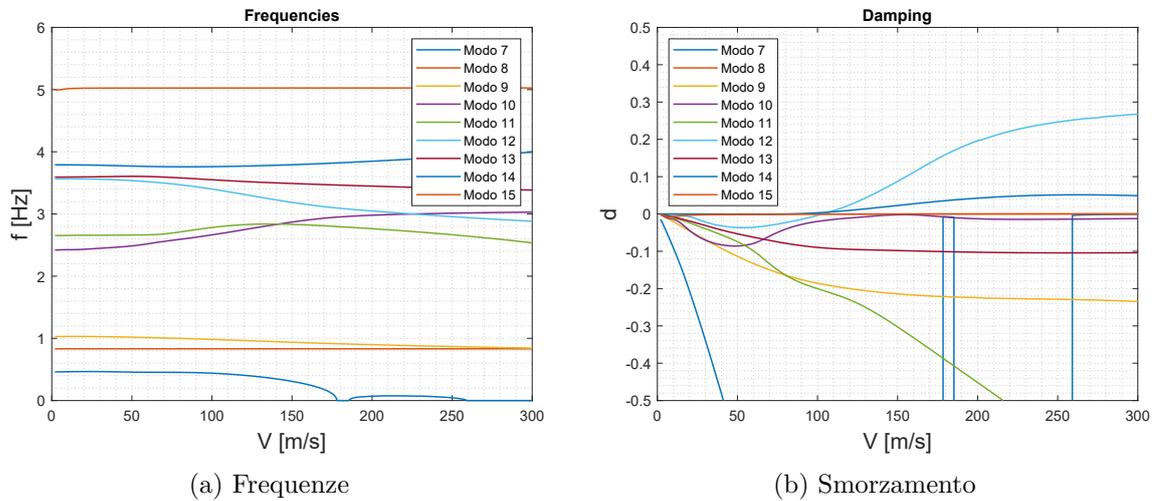


Figura 7.39: Frequenze e smorzamenti di alcuni modi del velivolo libero con i motori.

Il primo modo ad andare in instabilità è il 12 ad una velocità di 103.50 m/s e una frequenza 3.386 Hz .

	V_F [m/s]	f_F [Hz]
Velivolo senza motori	108.15	4.657
Velivolo con motori	103.50	3.386

Tabella 7.13: Confronto della condizione di flutter tra il velivolo libero con i motori e senza di essi.

7.5 Effetto della posizione dei motori sulla condizione di flutter

In questa sezione verranno spostati i motori lungo l'apertura alare e lungo la corda in modo da studiare l'effetto che ha la posizione di questi sulla velocità e sulla frequenza di flutter.

7.5.1 Effetto della posizione del motore lungo la corda

Il motore viene spostato in 9 stazioni; a partire dalla posizione iniziale x_0 , in cui il baricentro dell'insieme pilone+motore è allineato con l'asse elastico dell'ala, si sposta il motore in 4 posizioni verso il trailing edge del profilo alare (indicate con il pedice negativo), e altre 4 posizioni verso il leading edge (indicate con il pedice positivo).

Considerando la posizione iniziale dei motori lungo l'apertura alare, cioè a $y = 4$ m circa dalla fusoliera, il risultato ottenuto è rappresentato nella seguente tabella, in cui le posizioni x_i sono misurate a partire dal centro elastico della sezione alare, ossia x_0 :

	x_{-4}	x_{-3}	x_{-2}	x_{-1}	x_0	x_1	x_2	x_3	x_4
x_i [m]	-0.6667	-0.5000	-0.3333	-0.1667	0.0000	0.1667	0.3333	0.5000	0.6667
V_F [m/s]	98.70	106.00	106.20	107.30	103.50	96.40	98.70	99.10	101.70

Tabella 7.14: Risultati relativi alla velocità di flutter per varie posizioni dei motori lungo la corda.

7.5. Effetto della posizione dei motori sulla condizione di flutter

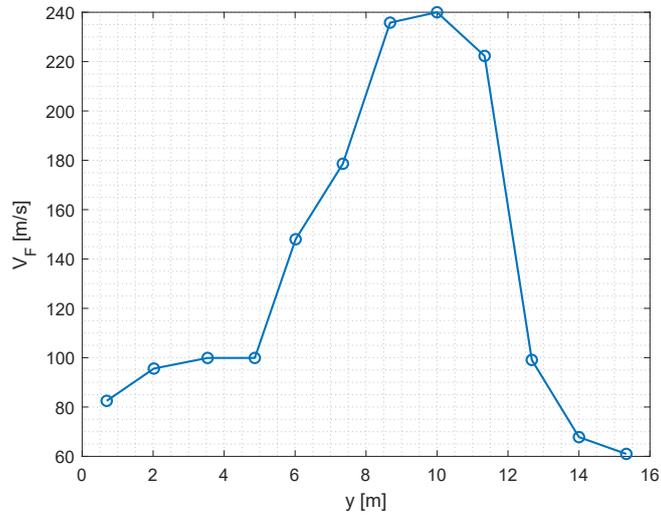


Figura 7.40: Andamento della velocità di flutter per varie posizioni dei motori lungo la corda.

Ciò che si ottiene è che a partire dalla posizione più avanzata dei motori la velocità di flutter aumenta fino ad un massimo in x_0 ; arretrando il motore ulteriormente le velocità di flutter sono inferiori.

7.5.2 Effetto della posizione del motore lungo l'apertura alare

Il motore viene ora spostato in 12 stazioni equamente spaziate partendo dalla radice fino all'apertura alare. La posizione di partenza è indicata come y_4 , le posizioni più vicine alla radice y_1 a y_3 , mentre quelle più vicine al tip alare vanno da y_5 a y_{12} . Nella tabella seguente sono riportate le velocità di flutter in varie posizioni dei motori lungo l'apertura alare con il baricentro dei motori in corrispondenza dell'asse elastico dell'ala.

	y_i [m/s]	V_F [m/s]
y_1 [m]	0.691	82.50
y_2	2.021	95.60
y_3	3.353	99.90
y_4	4.684	103.50
y_5	6.015	148.00
y_6	7.346	178.60
y_7	8.678	235.80
y_8	10.00	240.00
y_9	11.34	222.30
y_{10}	12.671	99.10
y_{11}	14.003	67.80
y_{12}	15.334	61.00

Tabella 7.15: Risultati relativi alla velocità di flutter per varie posizioni dei motori lungo l'apertura alare.

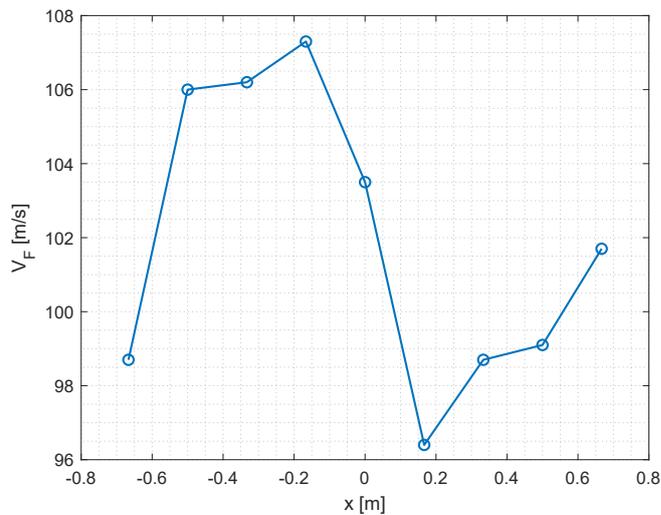


Figura 7.41: Andamento della velocità di flutter per varie posizioni dei motori lungo l'apertura alare.

La velocità di flutter viene massimizzata con i motori in posizione intorno al 50% dell'apertura alare, mentre al di fuori di quella regione il flutter si manifesta a velocità via via più basse.

7.5. Effetto della posizione dei motori sulla condizione di flutter

Le differenze negli andamenti della velocità al variare delle due posizioni considerate tra l'ala di Goland e il velivolo considerato si ritengono imputabili principalmente alla diversa geometria dell'ala. Nel caso del velivolo infatti, l'ala è realizzata come una struttura in parete sottile dotata di centine, cioè rinforzi, concentrate in un numero discreto di posizioni; l'ala di Goland invece è una struttura *piena* senza parti in parete sottile. Un'altra causa è imputabile alla diversità del materiale, dove per lo studio bibliografico osservato è utilizzato un materiale viscoelastico modellato con funzioni matematiche.

Poiché non è stato possibile trovare un confronto bibliografico che confermasse i risultati ottenuti per il velivolo, le analisi di machine learning verranno effettuate sull'ala di Goland con l'obiettivo di apprendere le sue caratteristiche di instabilità aeroelastiche e, grazie a queste, avere una base teorica sul suo comportamento. Si rende così possibile un futuro studio che veda come punto di partenza i risultati trovati in questo elaborato e vada a determinare le caratteristiche di un'ala di diversa geometria, ad esempio in parete sottile e dotata di centine di rinforzo.

Capitolo 8

Analisi di machine learning

In questo capitolo verrà esposta la procedura che si è seguita per le analisi condotte con le reti neurali e verranno commentati i risultati ottenuti. Prima di effettuare ciò però viene definito il dataset ottenuto a partire dall'ala di Goland.

8.1 Creazione del dataset

Le analisi condotte sull'ala di Goland comprendono 9 posizioni dei motori lungo la corda e 12 posizioni lungo l'apertura alare, quindi un totale di 108 posizioni diverse dei motori, a cui corrispondono altrettante velocità di flutter dell'ala. Per ciascuna posizione sarà pertanto necessario effettuare un'analisi aeroelastica così da estrarre la velocità. Il dataset è pertanto formato da due input, x_e , y_e e un output, V_F .

Per eseguire un'analisi con le reti neurali è necessario suddividere l'intero dataset in tre set più piccoli: il set di *train* finalizzato ad addestrare la rete, il set di *validation* che valuta la bontà dell'addestramento, durante l'addestramento stesso, agendo sul gradiente della funzione di perdita, e il set di *test* per verificare che il modello realizzato riesca a prevedere dei risultati che non ha mai incontrato durante la fase di training. È proprio sui risultati di test che viene stabilito quanto il modello di rete neurale è capace di svolgere il suo compito.

Poiché non esistono degli algoritmi o metodi che specifichino esattamente come costruire l'architettura delle rete, ci si deve affidare a regole generali per una prima approssimazione, per poi valutare come impatta ogni parametro singolarmente sui risultati finali.

Una regola generale per dividere il dataset è destinare il 20% dei dati al set di test, l'80% per il training, e il 10% dei dati di training per il set di validation. Applicando questa regola si ottiene un dataset suddiviso come descritto dalla tabella di seguito.

Set	Quantità di dati	Frazione %
Dataset iniziale	108	100%
Test set	22	20%
Training	86	80%
→ Validation	→ 9	→ 10%
→ Train	→ 77	→ 90%

Tabella 8.1: Suddivisione del dataset nei tre set utili.

Il modo in cui il singolo dato viene assegnato ad uno dei tre set può essere controllato direttamente dal codice agendo sul random state. Questo verrà discusso nella prossima sezione.

8.2 Costruzione del modello

Fin'ora si sa che il modello deve avere due input, le due posizioni dei motori, e un output, la velocità di flutter. Come già accennato nella sezione precedente, non esiste un metodo univoco per determinare l'architettura migliore; si procede allora variando alcuni parametri interni e valutando gli errori di volta in volta.

8.2.1 Numero di layers della DNN

Sono state eseguite analisi variando il numero di neuroni: 20, 32, 64, 128 ma l'errore minimo si è riscontrato con 32. Si sceglie pertanto come singolo layer uno strato di 32 neuroni, ognuno dotato di una **ReLU** come funzione di attivazione, tipica nelle reti profonde. Vengono eseguite diverse analisi variando il numero di layers da 2 a 7 e monitorando gli errori di train, di validation e di test calcolati con la metrica **MSE** già incontrata in (5.2).

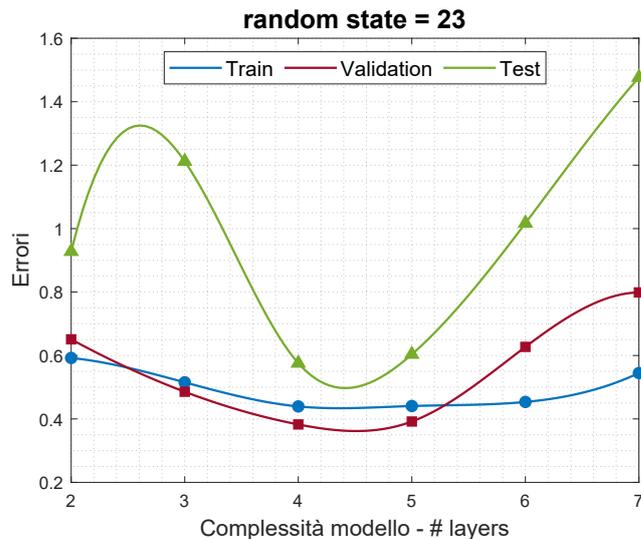


Figura 8.1: Errori di train, validation e di test per diverse complessità del modello, con un random state pari a 23

Nella Figura 8.1 viene mostrato il comportamento dei tre errori al variare del numero di layers interni della rete. Come detto in precedenza, la bontà del modello viene valutata osservando l'errore di test, ma è importante controllare che anche l'errore di validation non sia eccessivo. I modelli con 4 e 5 layers sembrano essere i migliori, ma poiché le differenze negli errori sono minime si definisce come modello migliore quello che ne ha 4 (poiché ha meno parametri da allenare, riduce il rischio di overfitting e ha un costo computazionale inferiore).

Questo risultato è stato ottenuto impostando un random state pari a 23. L'estrazione dei dati dal dataset iniziale, per suddividerli nei tre set, è randomica e non esiste alcun criterio che faccia preferire l' i -esimo dato al set di train, di validation o di test. Fissare il random state ad un certo valore garantisce che ogni volta che viene eseguito il training del modello i tre set siano sempre formati dagli stessi dati. Cambiare il random state, e fissarlo pari ad un altro numero, permette di ottenere dei set diversi. È utile fare diverse analisi con diversi valori di random states per poter monitorare gli errori con dei set diversi di dati.

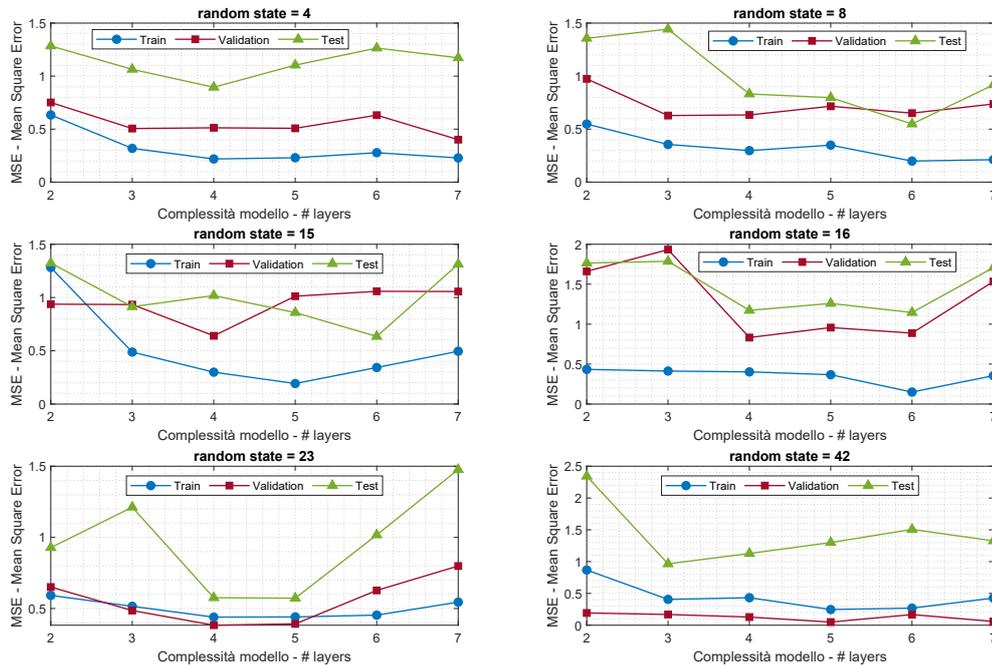


Figura 8.2: Errori di train, validation e di test per diverse complessità del modello, e per diversi random states.

Si sono scelti 6 diversi valori di random state:

$$random\ state = \{4\ 8\ 15\ 16\ 23\ 42\}$$

come si nota dai grafici in Figura 8.2, la soluzione migliore è sempre intorno a 4 layers interni. Questo suggerisce che l'architettura migliore sia proprio quella.

Da precisare che il numero di epoche con cui sono state allenate le varie configurazioni è diverso poiché è stato introdotto nel codice un `Checkpoint` per salvare il modello che minimizza l'errore di validazione. In questo modo si previene il rischio di overfitting.

Si fissa pertanto l'architettura della rete a 4 layers interni da 32 neuroni ciascuno, questa viene mostrata nella figura seguente.

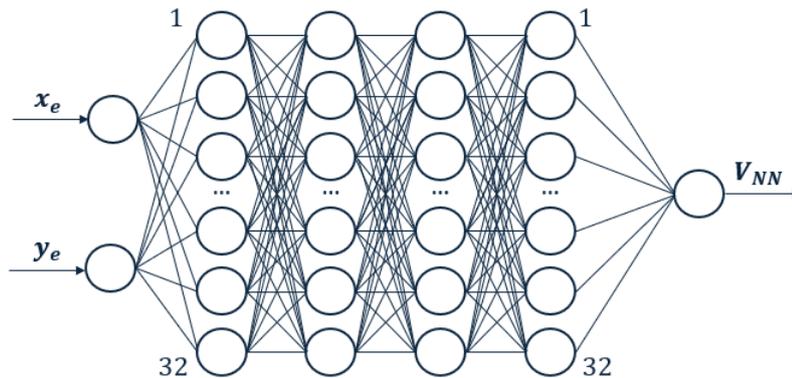


Figura 8.3: Architettura finale della rete neurale.

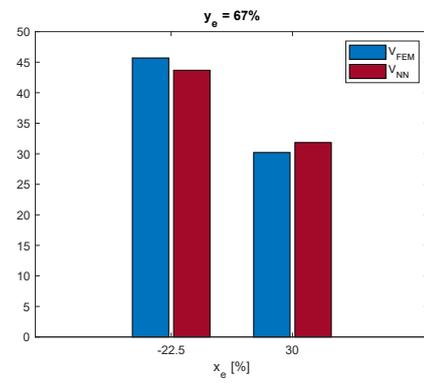
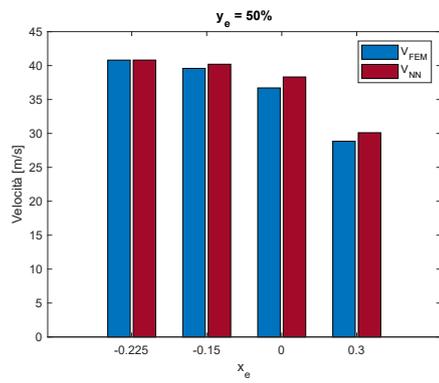
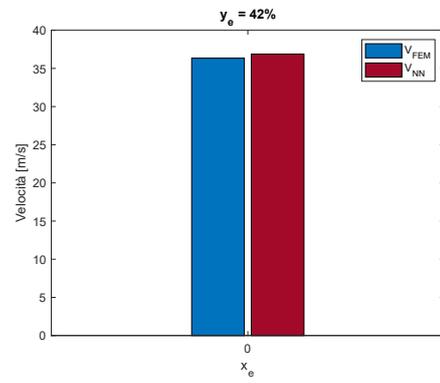
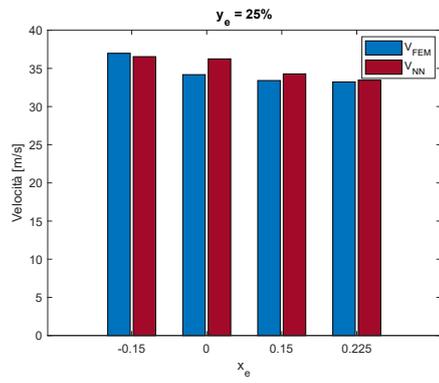
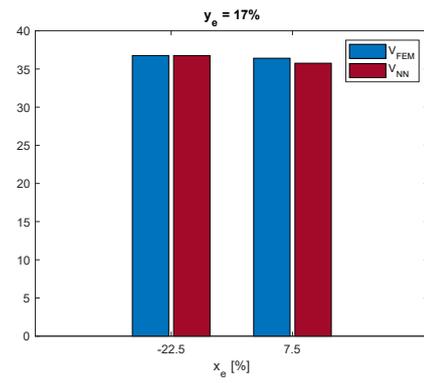
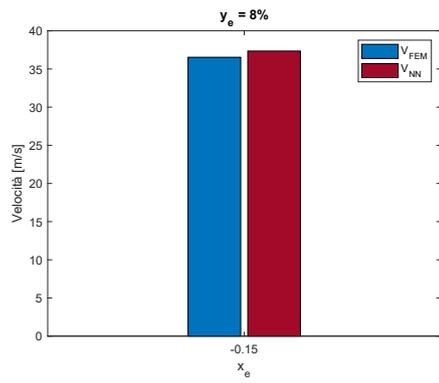
Per fornire una stima del numero di parametri che la rete allena per riprodurre le velocità note, è utile ricordare che ogni neurone vede in input un bias b e n pesi w provenienti dagli n neuroni del layer precedente (vedi Figura 5.6). Con ciò a mente si può costruire la tabella di seguito.

Layer	Neuroni	Parametri
1	32	96
2	32	1056
3	32	1056
4	32	1056
<i>output</i>	1	33
Totale		3297

Tabella 8.2: Numero di parametri che la rete allena durante il training.

La rete così costruita è in grado di prevedere le velocità di flutter fornite dalle analisi FEM, commettendo un certo errore che dipende dall'inizializzazione randomica dei pesi iniziali alla prima epoca, dal random state ma anche dalla dimensione dei tre set. Con un random state di 16 si ottengono i seguenti risultati.

8.2. Costruzione del modello



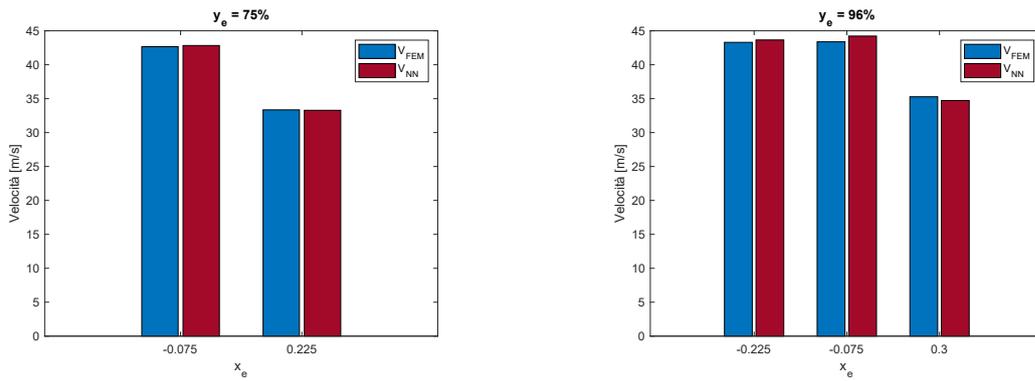
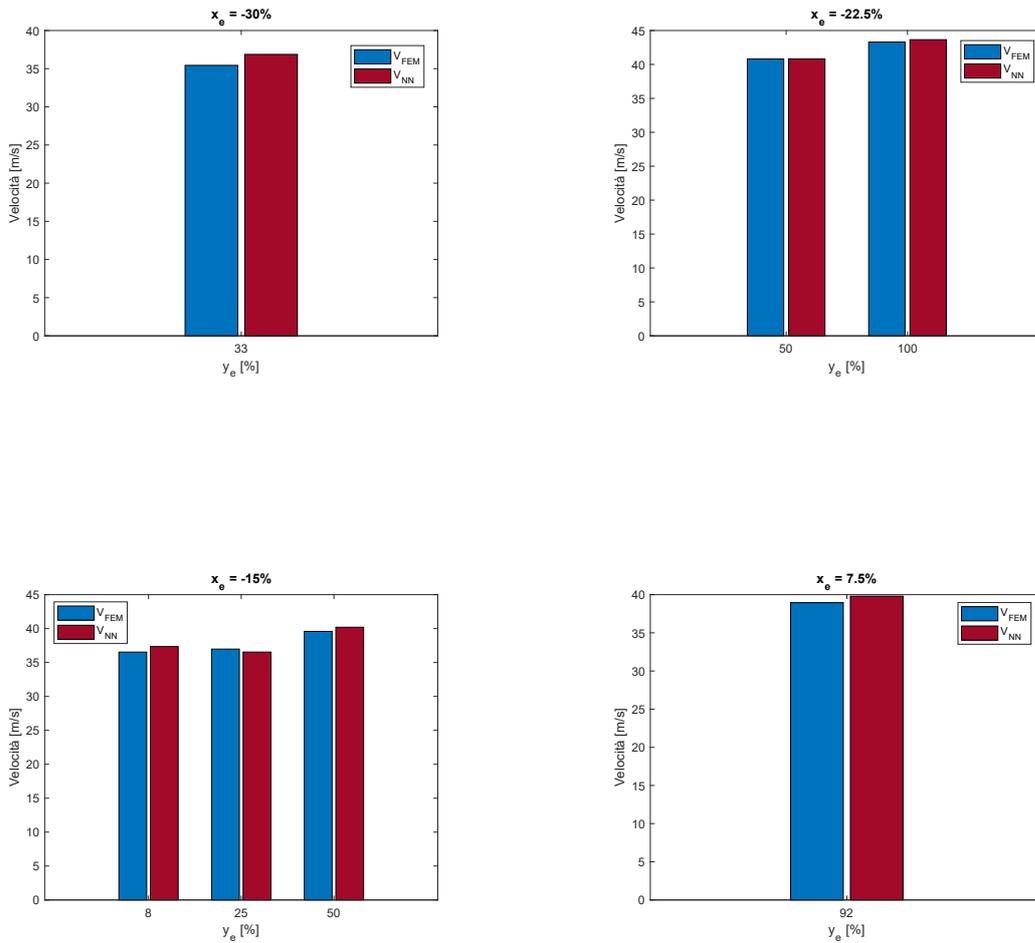


Figura 8.7: Confronto tra le due velocità al variare dell'input x_e e per diverse posizioni y_e .



8.3. Ottimizzazione delle dimensioni dei set e del numero di epoche di training

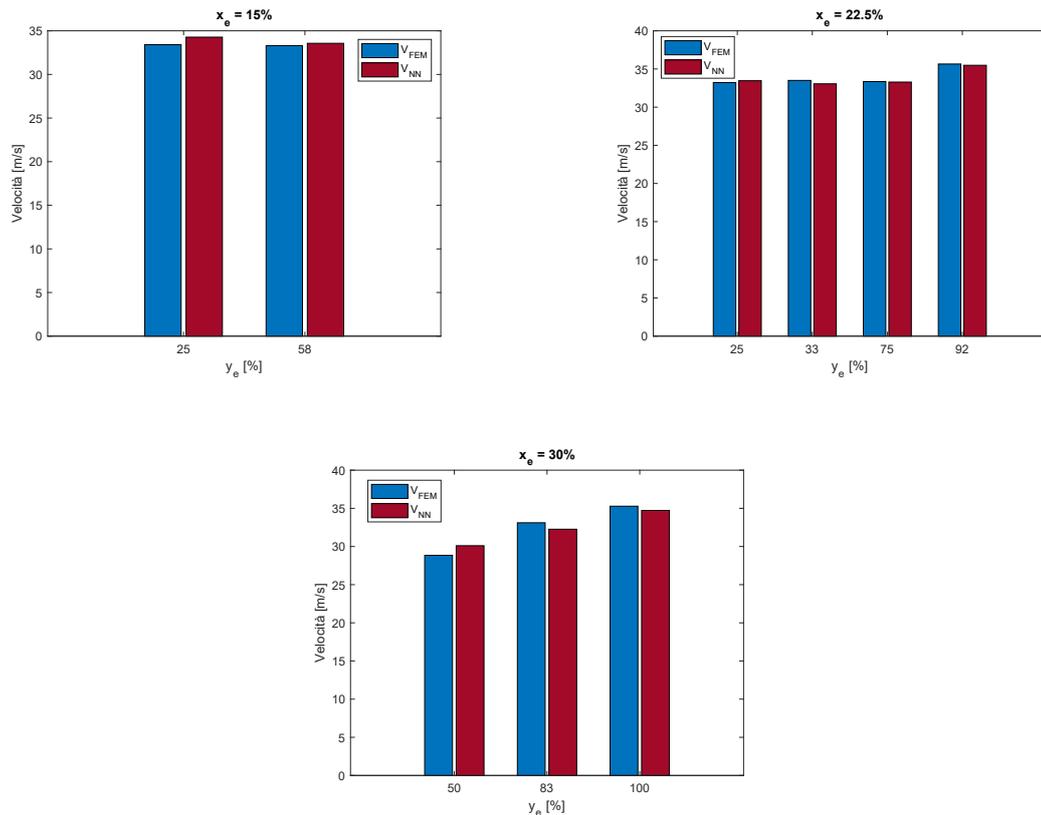


Figura 8.11: Confronto tra le due velocità al variare dell'input y_e e per diverse posizioni x_e .

Gli ultimi grafici mostrano come le velocità predette dalla rete neurale si avvicinino molto a quelle trovate con le analisi FEM nelle diverse posizioni considerate

8.3 Ottimizzazione delle dimensioni dei set e del numero di epoche di training

In questa sezione si vuole dimostrare come sia possibile ridurre la dimensione del set di training ottenendo delle performance comunque accettabili da parte del modello di machine learning. Ridurre il train set significa fare un allenamento con meno input, cioè dare meno informazioni alla rete richiedendo prestazioni comunque accettabili.

Il vantaggio principale che si ottiene da queste considerazioni è che avere meno input da fornire alla rete neurale equivale a un risparmio computazionale nella fase di creazione del dataset, poiché ogni dato corrisponde ad una simulazione numerica con codice FEM diversa.

8.3.1 Scelta *manuale* degli input

Riducendo progressivamente la dimensione del set di training è possibile che, a seconda del random state, la rete neurale non venga allenata lungo una delle due direzioni in input, mentre

viene fatta variare l'altra. Ad esempio, le posizioni dei motori lungo la corda variano tra le seguenti coordinate:

$$x_e = \{-0.30 \quad -0.225 \quad -0.15 \quad -0.075 \quad 0.00 \quad 0.075 \quad 0.15 \quad 0.225 \quad 0.30\}$$

per ciascuna di esse ci sono 12 posizioni y_e lungo l'apertura alare. Riducendo il numero di input da fornire al training può capitare che tutte e 12 le posizioni in corrispondenza di un certo x_i passano al set di test o di validation, e non a quello di train. La rete perciò rischia di non essere addestrata correttamente. Per evitare questo problema si è deciso di selezionare una percentuale variabile di input da ogni posizione x_i . Così facendo si assicura che almeno l'input x_e sia sempre presente durante il training.

Un'altra considerazione da fare è riguardo il validation set. All'aumentare del test size viene ridotto il set di training, e di conseguenza anche quello di validation. Poiché già in partenza, con un test size fisso al 20%, il set di validation era esiguo (9 dati), si è deciso che per le successive analisi questo sia fisso allo 0% giacché un set così scarso di dati non è in grado di quantificare con un'accuratezza sufficiente la bontà del processo di training.

Si vogliono effettuare 4 casi in cui la dimensione del set di train è sempre più piccola. I dati che vengono rimossi da questo set vengono aggiunti a quello di test così da avere più dati per validare il modello. Dunque verranno considerati 4 casi in cui il set di test è pari al 33%, 42%, 50% e 58% del totale dei dati. Di seguito vengono riportati i risultati ottenuti in cui si cerca di replicare le curve bibliografiche trovate nel capitolo precedente.

8.3. Ottimizzazione delle dimensioni dei set e del numero di epoche di training

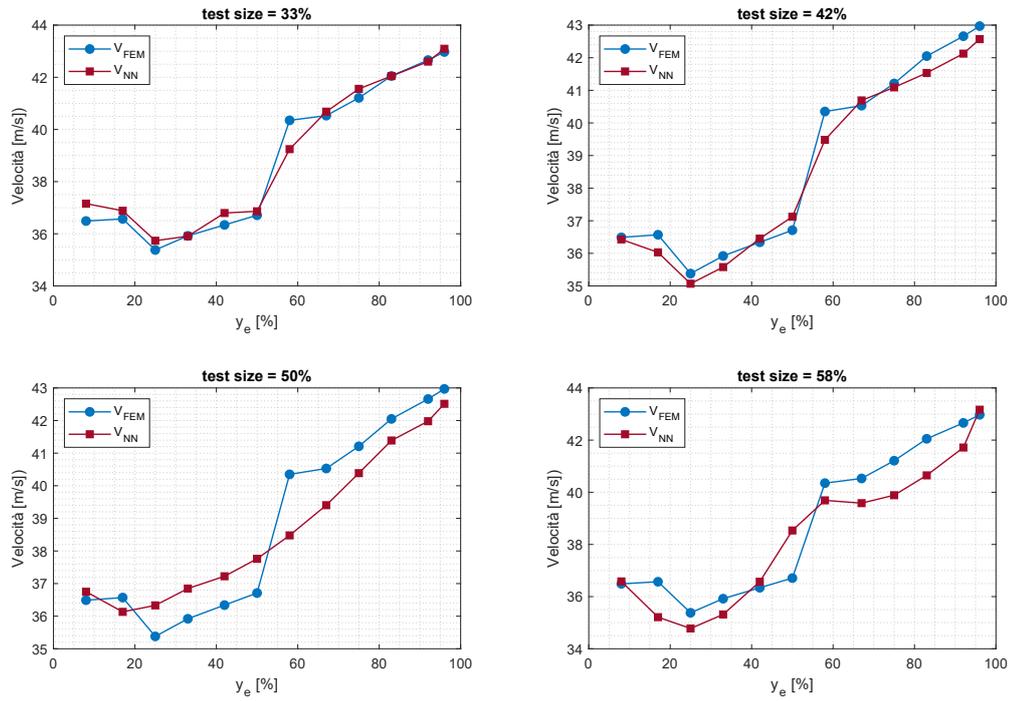


Figura 8.12: Velocità note e predette a confronto dei vari test size per $x_e = 0$ e lungo l'apertura alare

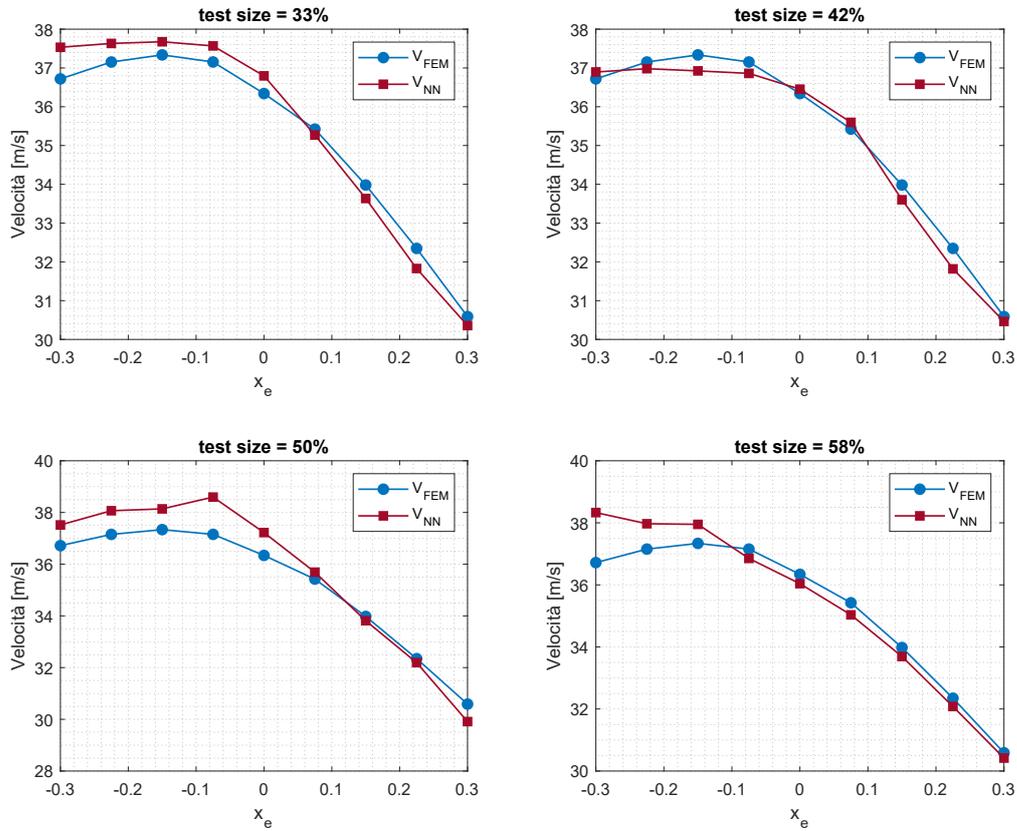


Figura 8.13: Velocità note e predette a confronto dei vari test size per $y_e = 42\%$ e lungo la corda

8.3.2 Valutazione degli errori e scelta del dataset migliore

Per ogni test size analizzato, le velocità predette dalla rete neurale si discostano, in qualche misura, da quelle note dalle analisi strutturali. Per valutare il caso migliore si mettono a confronto gli errori di ogni test size valutati come

$$Err = |V_{FEM} - V_{NN}|. \quad (8.1)$$

Dai grafici ottenuti risulta chiaro come riducendo il set di train il modello è sempre meno in grado di replicare i risultati commettendo errori sempre maggiori. La valutazione del modello migliore però non è semplice poiché è necessario tenere in conto di diversi parametri. Nella sezione successiva verrà illustrato il processo logico seguito per individuare il set che più minimizza il costo computazionale delle simulazioni numeriche, garantendo errori contenuti.

8.3. Ottimizzazione delle dimensioni dei set e del numero di epoche di training

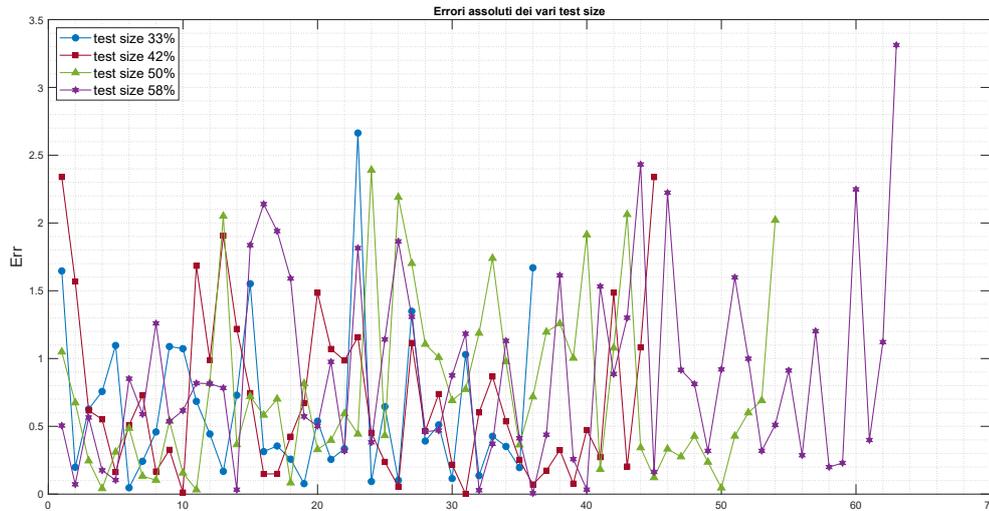


Figura 8.14: Confronto degli errori su ogni dato nei 4 test size considerati.

Dal momento che dalla Figura 8.14 è difficile interpretare quale sia il test size che produca un errore più basso sull'output si è deciso di procedere nel modo seguente: ogni errore viene trattato come una funzione continua su \mathbb{R}^1 e limitata. Ciò consente di definire, per ciascuna funzione, il valore massimo, mediano, medio e RMS. Per una maggior precisione nella scelta si quantifica anche il numero di volte in cui l'errore, valutato come la differenza tra le due velocità con (8.1), supera una certa soglia oltre il quale la previsione fornita dalla rete è considerata sbagliata. Questo valore di soglia viene scelto pari a 1.50 m/s .

Con questa procedura si ottiene la tabella seguente.

test size	33%	42%	50%	58%
Massimo [m/s]	2.664	2.539	2.673	3.312
Medio [m/s]	0.629	0.704	0.749	0.893
Mediano [m/s]	0.436	0.537	0.688	0.813
RMS [m/s]	0.848	0.925	0.980	1.132
$Err > 1.50 \text{ m/s}$	4	6	8	13
$Err \% > 1.50 \text{ m/s}$	11%	13%	15%	21%

Tabella 8.3: Parametri per monitorare gli errori dei vari test size a confronto.

Con la Tabella 8.3 si ha un'idea più chiara sul comportamento dell'errore nei vari casi: l'errore massimo commesso aumenta progressivamente all'aumentare della dimensione del set di test, così come il valore mediano, medio e RMS.

Data la presenza di outliers in ogni negli errori dei vari test size si preferisce prendere in considerazione il valore mediano piuttosto che il valor medio: il 50% degli errori di ogni test size è inferiore al valore mediano. Il restante 50% è invece al di sopra. Grazie al valore di tolleranza introdotto è possibile stabilire anche il comportamento dell'errore al di sopra del valore mediano. La soluzione migliore è quella con un test size del 33% poiché garantisce un numero di previsione dei dati oltre la soglia di tolleranza basso. Tuttavia, si ritiene accettabile anche il test size al 42% poiché nonostante l'errore massimo aumenti, così come il numero dei dati previsti non correttamente, osservando il valore mediano ci si rende conto che la popolazione degli errori di quel test size è così divisa: il 50% delle previsioni hanno uno scostamento dal valore reale (quello FEM) inferiore a 0.537 m/s , il 37% è contenuto tra 0.537 m/s e 1.50 m/s e solo il 13% dei dati di test viene prevista scorrettamente. Con queste conclusioni si vuole mostrare di seguito il comportamento del dataset al 42% di dati nel set di test, nella previsione delle velocità nelle varie posizioni dei motori.

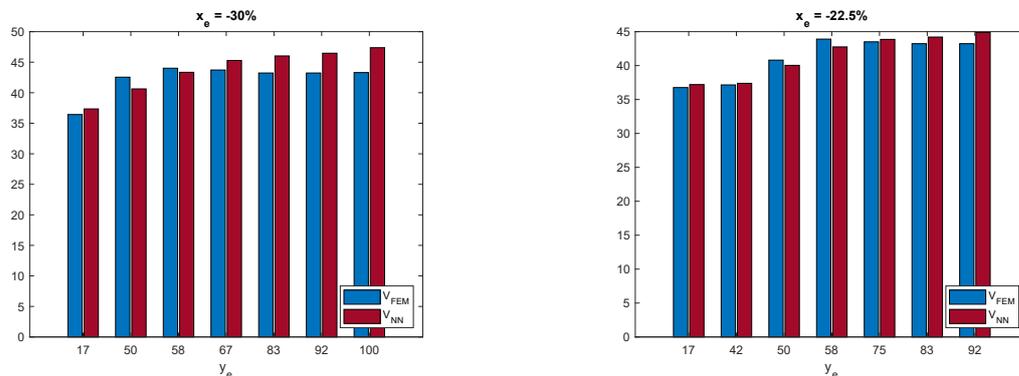
8.4 Analisi sul modello definitivo

Con le considerazioni fatte fin'ora, il modello di machine learning capace di prevedere il legame tra la velocità di flutter e la posizione dei motori lungo l'apertura alare e lungo la corda, per l'ala di Goland, è così formato:

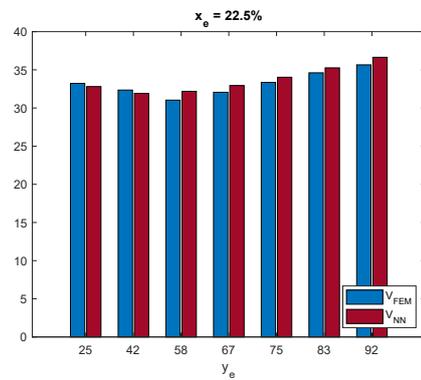
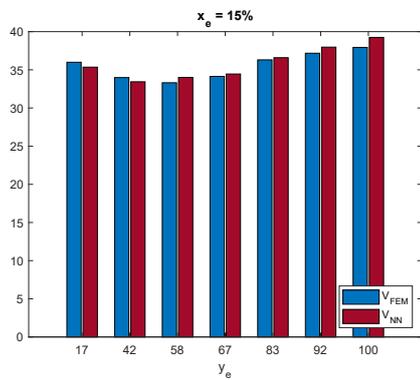
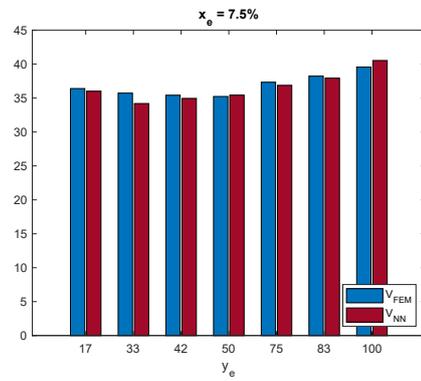
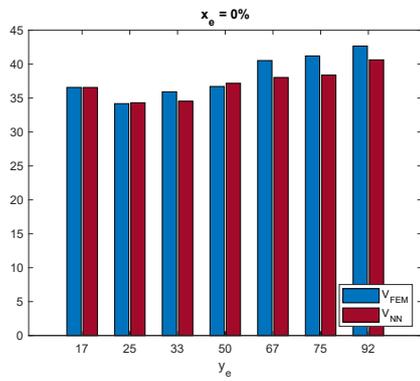
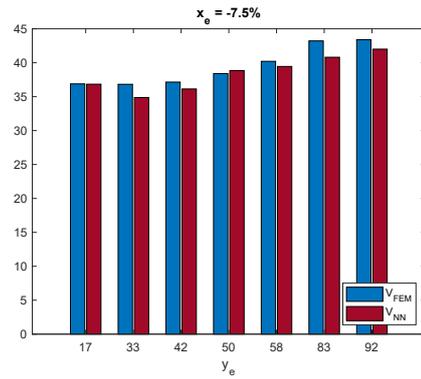
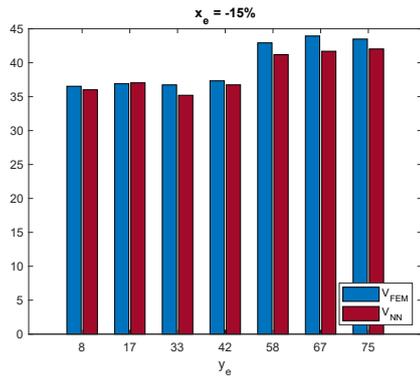
# layers × 64 neuroni	4
# parametri totali	3297
Test size	42%
Train size	58%
Validation size	0%

Tabella 8.4: Parametri definitivi della rete neurale.

Si è costruita la rete neurale con i parametri elencati nella Tabella 8.4, e con essi si sono condotte le analisi di machine learning sul dataset formato dai risultati riguardanti l'ala di Goland.



8.4. Analisi sul modello definitivo



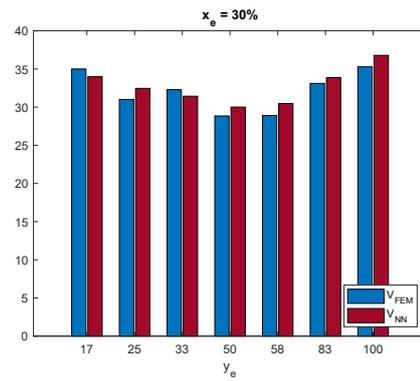
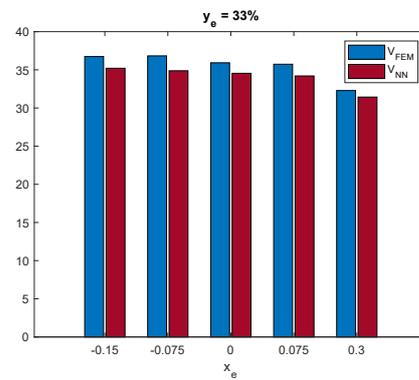
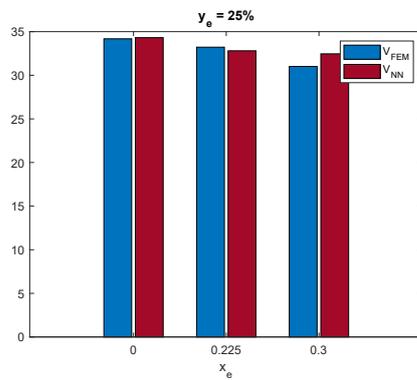
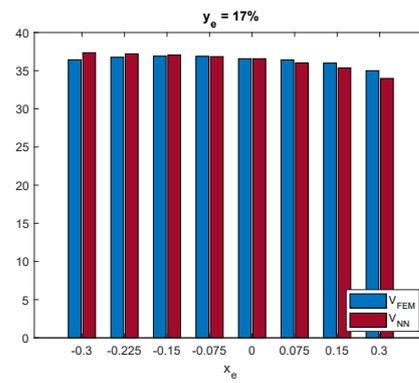
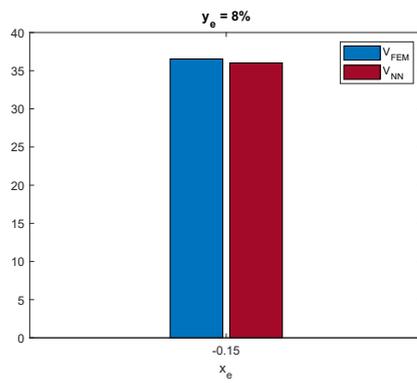


Figura 8.19: Confronto tra le due velocità al variare dell'input y_e e per diverse posizioni x_e



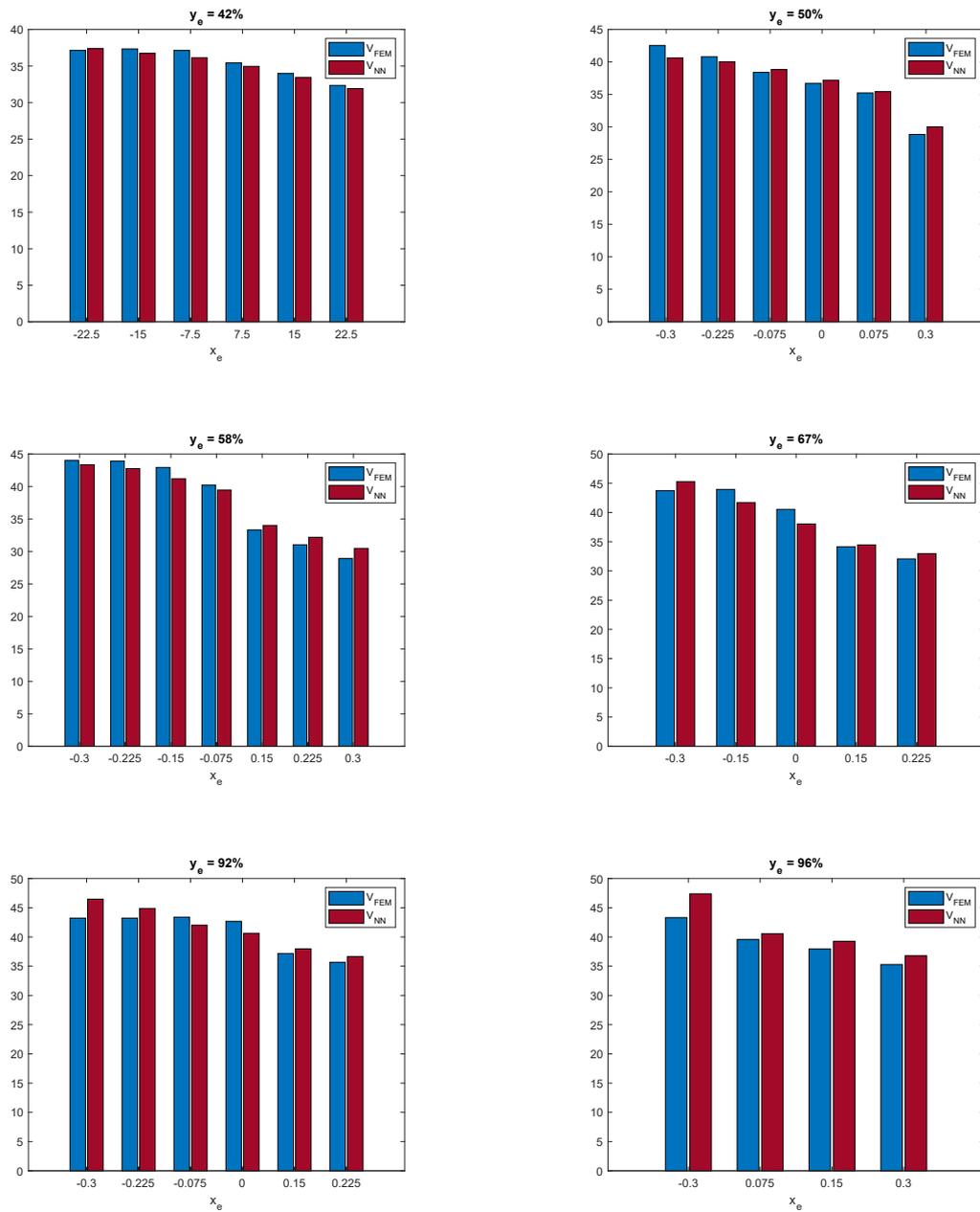


Figura 8.24: Confronto tra le due velocità al variare dell'input x_e e per diversi valori di y_e

Il modello di machine learning che utilizza il 58% dei dati per eseguire il training è in grado di approssimare discretamente il legame tra la velocità di flutter e la posizione dei motori su tutta l'ala. Per eseguire ciò utilizza un set di train popolato con 63 dati consentendo un risparmio sul numero di simulazioni necessarie a descrivere il fenomeno pari al 42%.

Capitolo 9

Conclusioni

Il lavoro su questa tesi si è concentrato sullo studio della risposta al flutter di un velivolo, e di un'ala incastrata, per diverse posizioni dei motori sulla superficie alare. L'obiettivo principale era quello di dimostrare come sia possibile utilizzare modelli di machine learning, quali sono le reti neurali artificiali, per ridurre il costo computazionale delle simulazioni numeriche da eseguire.

Stabilita la risposta aeroelastica del velivolo considerato e già esaminato in precedenza da altri studi, si sono modellati i motori per renderli i più verosimili possibili ad un caso reale. Il comportamento aeroelastico del velivolo compreso di motori è risultato essere difficilmente confrontabile con altri casi studio a causa della specificità del problema. Per questo motivo si è preferito proseguire con le analisi successive utilizzando l'ala di Goland.

Le analisi condotte sull'ala di Goland hanno mostrato come variano le caratteristiche aeroelastiche su un'ala incastrata. In questo caso, è stato possibile verificare la fedeltà dei risultati ottenuti con diversi studi bibliografici e, pertanto, si è ritenuto opportuno tentare di riprodurre con le reti neurali i risultati su questo caso studio.

Le reti neurali artificiali hanno subito mostrato come sia possibile riprodurre i risultati di tutte le simulazioni numeriche condotte, risparmiando il costo computazionale per ottenere alcuni dati. Con la prima suddivisione dei set (80% training + 20% test), eseguita in accordo con le tecniche più comunemente utilizzate in questi ambiti, la rete neurale ottiene in output, con i dati di test, che non aveva mai incontrato durante l'addestramento, velocità prossime a quelle trovate con le simulazioni numeriche. Ciò suggerisce come sia possibile ottenere il 20% dei risultati senza eseguire simulazioni con i codici di calcolo.

Riconosciuto ciò, si è determinato fino a che punto è possibile ridurre il set di training e quindi il numero di simulazioni FEM da condurre. Si sono studiati 4 casi in cui progressivamente il set di training veniva ridotto, arricchendo via via quello di test per avere più dati con cui testare il modello. Come prevedibile le prestazioni del modello diminuivano con il diminuire dei dati utilizzati per allenare la rete, ma con il 58% dei dati di simulazioni si riesce ancora ad avere una buona approssimazione dei risultati. Per set di training più piccoli non si riesce a ricostruire in modo abbastanza fedele il legame tra le due posizioni x_e , y_e e la velocità di flutter.

Si sostiene che l'impiego delle reti neurali artificiali può essere uno strumento molto potente in fase di progettazione per risparmiare sul costo computazionale.

Un possibile prossimo sviluppo a riguardo dello studio svolto può essere l'introduzione di ulteriori parametri di input come la terza direzione non considerata z_e fuori dalla superficie alare per la posizione dei motori, il numero di questi ma anche la geometria della struttura alare o il suo materiale, con l'obiettivo di determinare il legame tra loro e la condizione di instabilità al flutter. A tal scopo sarebbe possibile impiegare l'utilizzo delle reti convoluzionali che consentono di avere architetture con meno parametri di aggiornare ad ogni epoca di training.

Bibliografia

- [1] I. E. Garrick and W. H. Reed III. Historical Development of Aircraft Flutter. *Journal of Aircraft*, 18(11):897–912, 1981.
- [2] A. G. Von Baumhauer and C. Koning. On the stability of oscillations of an airplane wing. Technical report, 1923.
- [3] T. Theodorsen. General theory of aerodynamic instability and the mechanism of flutter. Technical report, 1979.
- [4] H. Ashley and G. Zartarian. Piston Theory-A New Aerodynamic Tool for the Aeroelastician. *Journal of the aeronautical sciences*, 23(12):1109–1118, 1956.
- [5] J. Durkin. History and applications. *Expert Systems*, pages 1–22, 2002.
- [6] A. L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229, 1959.
- [7] C. E. Shannon. A chess-playing machine. In *Computer Games I*. Springer, 1988.
- [8] E. Carrera, M. Cinefra, M. Petrolo, and E. Zappino. *Finite Element Analysis of Structures Through Unified Formulation*. John Wiley & Sons, 2014.
- [9] E. Carrera, G. Giunta, and M. Petrolo. *Beam Structures: Classical and Advanced Theories*. John Wiley & Sons, 2011.
- [10] L. Prandtl. Applications of modern hydrodynamics to aeronautics. 1923.
- [11] J. J. Bertin and R. M. Cummings. *Aerodynamics for engineers*. Cambridge University Press, 2021.
- [12] T.P. Kalman, W. P. Rodden, and J.P. Giesing. Application of the Doublet-Lattice Method to Nonplanar Configurations in Subsonic Flow. *Journal of Aircraft*, 8(6):406–413, 1971.
- [13] E. Albano and W.P. Rodden. A Doublet-Lattice Method for Calculating Lift Distributions on Oscillating Surfaces in Subsonic Flows. *AIAA journal*, 7(2):279–285, 1969.
- [14] M.T. Landahl. Kernel Function for Nonplanar Oscillating Surfaces in a Subsonic Flow. *AIAA Journal*, 5(5):1045–1046, 1967.

-
- [15] W. P. Rodden, P. F. Taylor, and S. C. McIntosh Jr. Further Refinement of the Subsonic Doublet-Lattice Method. *Journal of aircraft*, 35(5):720–727, 1998.
- [16] R. L. Harder and R. N. Desmarais. Interpolation using surface splines. *Journal of aircraft*, 9(2):189–191, 1972.
- [17] H. J. Hassig. An approximate true damping solution of the flutter equation by determinant iteration. *Journal of Aircraft*, 8(11):885–889, 1971.
- [18] K. Suzuki. *Artificial neural networks: Architectures and applications*. BoD–Books on Demand, 2013.
- [19] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola. Dive into deep learning. *arXiv preprint arXiv:2106.11342*, 2021.
- [20] P. Ramachandran, B. Zoph, and Q. V. Le. Swish: a self-gated activation function. *arXiv preprint arXiv:1710.05941*, 7(1):5, 2017.
- [21] K. Hornik, M. Stinchcombe, and H. White. Universal approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks. *Neural networks*, 3(5):551–560, 1990.
- [22] G. Benyamin and C. Mark. The Theory Behind Overfitting, Cross Validation, Regularization, Bagging, and Boosting: Tutorial. 2023.
- [23] D. H. Hubel. Single unit activity in striate cortex of unrestrained cats. *The Journal of physiology*, 147(2):226, 1959.
- [24] A. Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. ” O’Reilly Media, Inc.”, 2022.
- [25] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi. Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9:611–629, 2018.
- [26] S. Cai, Z. Mao, Z. Wang, M. Yin, and G. E. Karniadakis. Physics-Informed Neural Networks (PINNs) for fluid mechanics: a review. *Acta Mechanica Sinica*, 37(12):1727–1738, 2021.
- [27] S. Markidis. The Old and the New: Can Physics-Informed Deep-Learning Replace Traditional Linear Solvers? *Frontiers in big Data*, 2021.
- [28] L. Viscardi. Analisi aeroelastica e controllo di strutture portanti mediante modelli FEM e a parametri discreti, 2021.
- [29] F. Gruttmann and W. Wagner. Shear correction factors in Timoshenko’s beam theory for arbitrary shaped cross-sections. *Computational mechanics*, 27(3):199–207, 2001.
- [30] M. Santori. Analisi aeroelastiche di strutture di velivoli utilizzando teorie strutturali avanzate e DLM, 2023.

Bibliografia

- [31] M. J. Patil, D. H. Hodges, and C. E.S. Cesnik. Nonlinear aeroelasticity and flight dynamics of high-altitude long-endurance aircraft. *Journal of Aircraft*, 38(1):88–94, 2001.
- [32] M. Goland. The flutter of a uniform cantilever wing. 1945.
- [33] YS Matter, TT Darabseh, and A-H I Mourad. Effect of engine location on flutter speed and frequency of a tapered viscoelastic wing. 370(1), 2018.
- [34] SA. Fazelzadeh, A. Mazidi, and H. Kalantari. Bending torsional flutter of wings with an attached mass subjected to a follower force. *Journal of Sound and Vibration*, 323(1-2):148–162, 2009.
- [35] MR. Amoozgar, S. Irani, and GA. Vio. Aeroelastic instability of a composite wing with a powered-engine. *Journal of Fluids and Structures*, 36:70–82, 2013.
- [36] T. Yanko and O. Dmytrenko. Prospects for the implementation of new materials and technologies in the aerospace industry. *Transactions on Aerospace Research*, 2021(4):1–10, 2021.